

Dynamisierung elektronischer Märkte durch Adaption

INAUGURAL-DISSERTATION

zur Erlangung des akademischen Grades eines
Doktors der Wirtschaftswissenschaften
an der Wirtschaftswissenschaftlichen Fakultät
der Bayerischen Julius-Maximilians-Universität Würzburg

Vorgelegt von

Diplom-Kaufmann

Wolf Knüpffer

aus München

Würzburg 2002

Erstgutachter
Prof. Dr. R. Thome

Inhaltsverzeichnis

Inhaltsverzeichnis		I
Abbildungsverzeichnis		VI
Tabellenverzeichnis		X
Abkürzungsverzeichnis		XI
1	Motivation	1
1.1	Zielsetzung und Einordnung.....	2
1.2	Aufbau und Vorgehensweise.....	4
2	Ökonomische Grundlagen	6
2.1	Elektronische Märkte aus transaktionskostentheoretischer Sicht.....	6
2.2	Entstehung neuer Formen der Kooperation.....	12
2.3	Elektronische Märkte im Electronic Commerce	14
2.4	Ökonomische Effekte	16
2.4.1	Kommunikationseffekt.....	17
2.4.2	Integrationseffekt	19
2.4.3	Handelsmittlereffekte.....	21
2.4.4	Netzeffekte	25
2.5	Formen des Electronic Commerce und ökonomische Effekte	28
2.5.1	Online-Shops.....	29
2.5.2	Online-Malls	30
2.5.3	Online-Auktionen	30

2.5.4	Virtuelle Gemeinschaften	31
2.6	Umsetzung ökonomischer Potentiale	33
3	Methoden, Werkzeuge und Vorgehensmodelle für die Entwicklung betriebswirtschaftlicher Anwendungssysteme	37
3.1	Begriffsabgrenzung	38
3.2	Paradigmen und Methoden der Softwareentwicklung	39
3.2.1	Individualsoftwareentwicklung	41
3.2.2	Komponentenbasierte Softwareentwicklung	42
3.2.2.1	Wichtige Eigenschaften von Softwarekomponenten	42
3.2.2.2	Formen der Schnittstellenimplementierung	46
3.2.3	Anpassung von betriebswirtschaftlicher Standardsoftware	47
3.3	Werkzeuge	48
3.3.1	Werkzeugklassifikation	48
3.3.2	Bewertung der Werkzeugklassen	50
3.3.2.1	Individualentwicklungsumgebungen	51
3.3.2.2	Funktionsorientierte Standardsoftware	53
3.3.2.3	Betriebswirtschaftliche Standardanwendungssoftwarelösungen	55
3.3.3	Folgerungen für die Werkzeugauswahl	59
3.4	Vorgehensmodelle	60
3.4.1	Sequentielle Vorgehensmodelle	63
3.4.2	Inkrementell-iterative Vorgehensmodelle	66
3.5	Zusammenfassung	69

4	Systementwicklung durch wissensbasierte Adaption	71
4.1	Continuous Systems Engineering.....	71
4.2	Adaption von Standardanwendungssoftware nach dem ODYSSEUS-Konzept.....	75
4.2.1	Adaptionsarten	76
4.2.2	Adaptionsprinzipien.....	77
4.2.2.1	Heuristische Analysestrategie	78
4.2.2.2	Frühzeitige Implementierung lösbarer Aufgaben.....	80
4.3	ODYSSEUS als Leitfaden zur Adaption elektronischer Märkte.....	81
4.3.1	Bezugsrahmen.....	81
4.3.2	Freiheitsgrade bei der Anpassung.....	83
4.3.2.1	Modellierungsunterstützung bei der Anpassung von Prozessen.....	85
4.3.2.2	Gestaltungsfreiheit beim Design der Benutzerschnittstelle	86
4.3.2.3	Erweiterbarkeit von Lösungen	87
5	Entwurf der Standardanwendungssoftwarebibliothek.....	89
5.1	Formulierung der Anforderungen.....	89
5.2	Systemarchitektur der Softwarebibliothek	91
5.3	Datenbankbasierte Anwendungsrepräsentation.....	97
5.3.1	Hierarchische Parameterstruktur.....	97
5.3.2	Abbildung von Komponentenexemplaren	101
5.4	Erweiterung der Softwarebibliothek.....	104

6	Konzeption des Anforderungsnavigators	107
6.1	Wissensbasierte Abbildung der Analyse.....	109
6.1.1	Auswahl einer Problemlösungsmethode	110
6.1.2	Aufbau der Wissensbasis.....	112
6.1.2.1	Formulierung des Fragenkatalogs.....	114
6.1.2.1.1	Einordnung des Unternehmens	117
6.1.2.1.2	Bewertung der Marktposition	118
6.1.2.1.3	Sortiments- und Produktanalyse	120
6.1.2.1.4	Prozessanalyse.....	124
6.1.2.1.5	Ressourcenanalyse	125
6.1.2.2	Strukturierung der Lösungsmenge	126
6.1.2.3	Anwendung der Inferenzstrategien	129
6.1.3	Generierung der Ergebnisdokumentation.....	131
6.2	Konzeption des Prozesseditors.....	133
6.3	Verfahren zum Design der Benutzerschnittstelle.....	137
6.4	Unterstützung der Systemeinführung.....	139
6.5	Integration der Erfolgskontrolle.....	140
7	Realisierung des IntelliShop-Systems.....	145
7.1	Systemarchitektur.....	145
7.2	Aufbau und Integration des Expertensystems.....	148
7.3	Anwendungsentwicklung mit dem CommerceNavigator	149
7.3.1	EC-Analyse.....	150
7.3.2	Anwendungsgestaltung.....	153
7.3.2.1	Anpassung der Anwendungsarchitektur	154
7.3.2.2	Anpassende Modellierung im Prozesseditor.....	155
7.3.2.3	Detailanpassung des Anwendungsdesigns.....	157
7.3.2.4	Anpassung einzelner Komponenten.....	158

7.3.2.5	Verlinken von Komponenten	160
7.3.3	Inbetriebnahme	162
7.4	Anwendungsbetrieb und Erfolgskontrolle.....	162
8	Erfahrungsbericht und Ausblick.....	165
8.1	Kritische Würdigung der Ziele	165
8.2	Einordnung der Ergebnisse.....	167
8.3	Bewertung aktueller Entwicklungen	168
Anhang	169
A1	Grafische Darstellung des Fragenkatalogs	169
A1.1	Frageklassenhierarchie.....	169
A1.2	Fragehierarchie	170
A2	Beispiel für einen wissensbasierten Dialog.....	171
A3	Entscheidungsbäume zur Ableitung von Diagnosen nach Frageklassen	177
Literaturverzeichnis	185

Abbildungsverzeichnis

Abbildung 1-1: Aufbau der Arbeit	4
Abbildung 2-1: Interaktionen zwischen Marktteilnehmern in den einzelnen Phasen einer Markttransaktion [in Anlehnung an ZERB99, S. 148].....	8
Abbildung 2-2: Entstehungsmuster elektronischer Märkte und hybrider Koordinationsformen [in Anlehnung an PICO98, S. 230]	12
Abbildung 2-3: Beispiele für Formen der Kooperation [in Anlehnung an MÜLL03, S. 20]	13
Abbildung 2-4: Bereiche des Electronic Commerce (in Anlehnung an [HERM99, S. 23]).....	15
Abbildung 2-5: Effekt höherer Informationstransparenz auf die Preisbildung [ZERB99, S. 152]	18
Abbildung 2-6: Wertschöpfungspotentiale durch digitalen Vertrieb über elektronische Märkte (in Anlehnung an [THOM00, S. 9])	20
Abbildung 2-7: Transaktionen zwischen Anbietern, Nachfragern und Handelsmittlern	21
Abbildung 2-8: Auswirkungen elektronischer Transaktionsabwicklung auf Handelsmittler (in Anlehnung an [SARK95]).....	23
Abbildung 2-9: Wirkung positiver Feedback-Effekte [ZERB99, S. 158].....	26
Abbildung 2-10: Erfolgsfaktoren für Virtuelle Gemeinschaften (in Anlehnung an [HAGE97, S. 75]).....	32
Abbildung 2-11: Ursachen für den Misserfolg in Projekten (in Anlehnung an [STRA02, S. 21]).....	33
Abbildung 2-12: Verfügbarkeit einer Strategie (in Prozent von 465 Unternehmen) [STRA02, S. 28]	34

Abbildung 2-13: Inhalte eines strategischen Konzepts	35
Abbildung 3-1: Trends bei Softwarelösungen [BALZ00, S. 33]	40
Abbildung 3-2: Vorgehensmodell zur Einführung von Electronic Commerce- Lösungen in KMU [LOHS02, S. 9].....	65
Abbildung 3-3: Aufwandsverteilung im Rational Unified Process (in Anlehnung an [KRUC00, S. 23])	67
Abbildung 4-1: Integration von Informationsverarbeitung und Organisation zu einer stetig verbesserten Lösung durch CSE [THOM96, S. 80]....	73
Abbildung 4-2: Phasenweise Realisierung des ODYSSEUS-Konzepts für die R3-Einführung (in Anlehnung an [VOGE98, S. 61])	79
Abbildung 5-1: Architektur des Komponentenframeworks	92
Abbildung 5-2: Aufruf einer Komponente	93
Abbildung 5-3: Elektronischer Vertrieb nicht-digitaler Güter	95
Abbildung 5-4: Elektronischer Vertrieb digitaler Güter	96
Abbildung 5-5: Entity-Relationship-Modell der Parameterverwaltung.....	98
Abbildung 5-6: Ablauf beim Einlesen der Komponentenparameter.....	104
Abbildung 6-1: Gesamtdarstellung des Adaptionleitfadens	108
Abbildung 6-2: Klassenstruktur des Fragenkatalogs.....	116
Abbildung 6-3: Heterarchisierung von Lösungselementen am Beispiel des elektronischen Vertriebs	128
Abbildung 6-4: Abbildung der wissensbasierten Analyse als spezielles Klassifikationsproblem	130
Abbildung 6-5: Generierung des Ergebnisberichts	132
Abbildung 6-6: Integration und Anwendung der Werkzeuge zur Prozessmodellierung.....	136
Abbildung 6-7: Werkzeugbasierte Unterstützung des Anwendungsdesigns.....	138
Abbildung 6-8: Ablauf der Systemeinführung	140

Abbildung 6-9: Durchführung der Erfolgskontrolle	141
Abbildung 7-1: Systemarchitektur des IntelliShop-Systems	146
Abbildung 7-2: Entscheidungsbaum zur Analyse der Kundenstruktur.....	148
Abbildung 7-3: Aufbau des CommerceNavigators.....	150
Abbildung 7-4: Analyse der Kundenstruktur in der EC-Analyse	151
Abbildung 7-5: Ergebnisbericht der EC-Analyse	152
Abbildung 7-6: Auswahl eines Strukturtemplates	154
Abbildung 7-7: Prozesseditor.....	155
Abbildung 7-8: Definition des Corporative Designs der Anwendung.....	157
Abbildung 7-9: Parametrisierung der Komponentenstrukturelation am Beispiel der Produktsuche	159
Abbildung 7-10: Parametrisierung der Elementerelation am Beispiel der Produktsuche	160
Abbildung 7-11: Linkmanager zur Verwaltung von Hyperlinks zwischen Komponenten.....	161
Abbildung 7-12: Produktverwaltung im IntelliShop.....	163
Abbildung A-1: Frageklassenhierarchie	169
Abbildung A-2: Frageklassenhierarchie	170
Abbildung A-3: Frageklasse „Allgemeine Angaben zu Ihrem Unternehmen“ im D3-System	171
Abbildung A-4: Folgefragen zur Unternehmenseinordnung	172
Abbildung A-5: Begründungsgraph nach der Unternehmenseinordnung.....	173
Abbildung A-6: Template zur Analyse der Kundenstruktur, Stufe I.....	173
Abbildung A-7: Begründungsgraph nach der Analyse der Kundenstruktur, Stufe	174
Abbildung A-8: Abschließender Begründungsgraph.....	174
Abbildung A-9: Ergebnis des Inferenzprozesses	175

Abbildung A-10: Entscheidungsbaum zur Einordnung des Unternehmens.....	178
Abbildung A-11: Entscheidungsbaum zur Klassifizierung der Kunden	179
Abbildung A-12: Entscheidungsbaum zur demographischen Analyse	180
Abbildung A-13: Entscheidungsbaum zur Wettbewerbsanalyse	181
Abbildung A-14: Entscheidungsbaum zur Produktanalyse, Teil I.....	182
Abbildung A-15: Entscheidungsbaum zur Produktanalyse, Teil II	183
Abbildung A-16: Entscheidungsbaum zur Prozessanalyse	183
Abbildung A-17: Entscheidungsbaum zur Ressourcenanalyse.....	184

Tabellenverzeichnis

Tabelle 2-1:	Handelsmittlerfunktionen und Handelsmittlersysteme	22
Tabelle 2-2:	Übersicht der Möglichkeiten zur Nutzung ökonomischer Effekte in Abhängigkeit der Ausprägungsformen elektronischer Märkte .	28
Tabelle 3-1:	Kriterien zur Bewertung der Produktklassen	51
Tabelle 3-2:	Übersicht marktführender Application Server	53
Tabelle 3-3:	Auswahl von Standardanwendungssoftwarelösungen für KMU...	57
Tabelle 3-4:	Bewertung der Produktklassen	58
Tabelle 5-1:	Strukturrelation zur Abbildung von Suchmasken	102
Tabelle 5-2:	Elementerelation von Suchmasken.....	102
Tabelle 6-1:	Anwendungsgebiete von Expertensystemen nach Problemklassen.....	111
Tabelle 6-2:	Auswahl relevanter Produkteigenschaften für den elektronischen Handel.....	123
Tabelle 6-3:	Zusammenfassung der Auswertungsmöglichkeiten von Datenquellen und Bewertung der Ergebnisse für die Erfolgskontrolle	143

Abkürzungsverzeichnis

al.	aliter
BPR	Business Process Reengineering
CORBA	COmmon Request Broker Architecture
CRM	Customer Relationship Management
CSE	Continuous Systems Engineering
DBMS	DatenBankManagementSystem
DCOM	Distributed Component Object Model
engl.	englisch
ER-Modell	Entity-Relationship-Modell
ERP	Enterprise Resource Planing
GIF	Graphics Interchange Format
HTML	HyperText Markup Language
HTTP	HyperText Transmission Protocol
IDEF	Integration DEFINITION for Information Modeling
IDL	Interface Definition Language
KI	Künstliche Intelligenz
KMU	Kleine und Mittlere Unternehmen
MECK	Mainfränkisches Electronic Commerce Kompetenzzentrum
o. A.	ohne Autor
ODBC	Open Database Connectivity
ODYSSEUS	Organisatorisch-Dynamische Spezifikation von Systemmodulen Entsprechend der UnternehmensStruktur

OMG	Object Management Group
OOP	ObjektOrientierte Programmierung
PENELOPE	Prozess-EbenenAnalyse für Ergänzungsentwicklung, Lückenidentifikation und Organisatorische Problemlösung
RDBMS	Relationales DatenbankManagementSystem
RGB	Red Green Blue
RMI	Remote Method Invocation
SGML	Standardized Generalized Markup Language
SMS	Short MeSsage
SQL	Structured Query Language
URL	Universal Resource Locator
UML	Unified Modeling Language
WAP	Wireless Application Protocol
WWW	World Wide Web
WML	Wireless Markup Language
XML	EXtensible Markup Language

1 Motivation

In der Einleitung der Richtlinie des Europäischen Parlaments und des Europäischen Rates über den elektronischen Geschäftsverkehr wird darauf hingewiesen, dass die elektronische Geschäftsabwicklung insbesondere in kleinen und mittleren Unternehmen (KMU) erhebliche Beschäftigungspotentiale und die Chance zur Stärkung der Wettbewerbsfähigkeit bietet [EU00]. Gerade diesen Unternehmen schafft das Internet einen Zugang zu neuen, globalen Märkten und damit u. a. völlig neue Chancen am Absatzmarkt. Bisher werden diese allerdings nur zögernd wahrgenommen [LOHS02, S. 11f.].

KMU haben andererseits gegenüber Großunternehmen erhebliche Nachteile bei der Erschließung elektronischer Märkte, denn sie verfügen in der Regel nur über geringes informationstechnisches Know-how und wenig qualifiziertes Personal für die Entwicklung eigener IT-Systeme. Schwerer als fehlendes technisches Know-how und geringe Kapazitäten wiegen mangelnde Erfahrungen bei der Entwicklung erfolgreicher Strategien für elektronische Märkte (siehe 2.6). So wird die Neuausrichtung des eigenen Unternehmens und der Umbau gewachsener Abläufe und Organisationsstrukturen zum schwer kalkulierbaren Risiko. Auch immer komfortabler und komplexer werdende Werkzeuge lösen dieses Problem nicht. Vielmehr ist die Vielfalt der Werkzeuge kaum mehr überschaubar. KMU sind oftmals schon mit der Auswahl geeigneter Produkte und Dienstleister überfordert, denn um diese Entscheidungen treffen zu können, sind fundierte technische Kenntnisse und genaue Vorstellungen über mittel- und langfristige technische und betriebswirtschaftliche Aufgabenstellungen im jeweiligen Projekt nötig. Vor allem die exakte Formulierung ihrer speziellen Anforderungen stellt KMU regelmäßig vor Probleme, da diese oft nicht über die notwendigen betriebswirtschaftlichen Erfahrungen beim Einsatz der neuen Medien und die erforderlichen technischen Spezialkenntnisse verfügen.

So bleibt der Einstieg in die digitale Geschäftsabwicklung über elektronische Märkte bisher überwiegend Unternehmen vorbehalten, die die Kompetenz zur Entwicklung aufwändiger Kommunikationssysteme besitzen und das Risiko einer Fehlentwicklung tragen können oder bewusst eingehen.

KMU benötigen dagegen einfache, kostengünstige und umfassende Unterstützung nicht nur bei der technischen Realisierung und dem Betrieb, sondern vor allem auch beim Entwurf einer geeigneten Lösung.

1.1 Zielsetzung und Einordnung

Ziel dieser Arbeit ist die Konzeption eines Werkzeugs, das den gesamten Projektverlauf der Entwicklung elektronischer Märkte von der Beschreibung einer betriebswirtschaftlich sinnvollen Gesamtkonzeption bis zur Implementierung und dem Betrieb einer Lösung unterstützt. Damit sollen auch Unternehmen mit geringem eigenem Know-how auf dem Gebiet der elektronischen Geschäftsabwicklung in die Lage versetzt werden, speziell auf ihre Anforderungen zugeschnittene Lösungen aufzubauen und langfristig erfolgreich zu betreiben. Entscheidend ist dabei, dass neben der Realisierung einer ersten Lösung auch die laufend notwendige Anpassung an neue Anforderungen unterstützt wird. Nur wenn auch diese Aufgabe von Beginn an im Projekt berücksichtigt wird, kann im dynamischen Umfeld des Electronic Commerce der wirtschaftliche Betrieb und die Attraktivität eines elektronischen Marktes langfristig sichergestellt werden.

Mehrjährige Erfahrungen bei der Konzeption, Realisierung und Anwendung von Werkzeugen zur Einführung betriebswirtschaftlicher Standardanwendungssoftware am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg haben gezeigt, dass durch Verwendung wissensbasierter Werkzeuge die individuellen Anforderungen eines Unternehmens analysiert und mit Hilfe einer Standardanwendungssoftwarebibliothek weitgehend automatisiert in ein hochwertiges, funktionsfähiges System überführt werden können. Bei konsequentem Einsatz dieser Werkzeuge wird ein völlig neuer Ansatz der Einführung und Anpassung betriebswirtschaftlicher Informationssysteme möglich.

Ausgangspunkt der Entwicklung dieses Ansatzes sind die Arbeiten von HUFgard und Thome. In [HUFg94] werden die wesentlichen konzeptionellen Grundlagen zur Adaption betriebswirtschaftlicher Softwarebibliotheken entwickelt und beschrieben. Der Begriff **Adaption** ist der Beschreibung biologischer Systeme entliehen. In der Biologie bedeutet Adaption die ständige Anpassung von Lebewesen an ihre Umwelt [HUFg94, S. 12]. Analog bezeichnet die Adaption im Zusammenhang mit der

Einführung von Softwarebibliotheken ihre kurzfristige Anpassung an Umweltveränderungen mit Hilfe eines Adaptionswerkzeuges. Auf Basis der Ergebnisse von HUFGARD wird in [THOM96] das Konzept des Continuous Systems Engineering dargestellt, das eine wesentliche Grundlage für die Ableitung des hier vorgestellten Ansatzes liefert. Ergänzend wird die Arbeit von VOGELSANG zur Entwicklung von Werkzeugen für die Adaption von Prozessen herangezogen [VOGE98].

Allerdings lassen sich die bisher vorwiegend für die Adaption innerbetrieblicher Informationssysteme verwendeten Methoden und Werkzeuge nicht ohne weiteres auch zum Aufbau elektronischer Märkte verwenden. Jede Online-Präsenz erfordert stets die direkte Auseinandersetzung mit Konkurrenten. Projekte zum Aufbau elektronischer Märkte unterliegen daher z. T. ganz eigenen Rahmenbedingungen. Mit der schnellen Verbreitung des World Wide Web entsteht unübersehbar eine Vielzahl neuer Werkzeuge für die unterschiedlichsten Anforderungen. Einige der verfügbaren Produkte werden als Standardsoftware bezeichnet. Die nähere Untersuchung des Marktangebots in Kapitel 3 zeigt, dass diese Produkte die schnelle Einführung und flexible Anpassung in sehr unterschiedlichem Maß unterstützen. Dies ist z. T. auf die mangelnde Adaptionfähigkeit oder auf unzureichende Adaptionswerkzeuge zurückzuführen. Eine Softwarebibliothek mit den erforderlichen Eigenschaften für die Adaption und ein entsprechendes Adaptionswerkzeug existieren bisher nicht.

Auf der Grundlage der Ergebnisse dieser Arbeit wurde am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik eine vollständige Lösung entwickelt, die sich seit 1998 unter dem Namen IntelliShop in der Version 2.0 im Einsatz befindet. Die Darstellung der praktischen Umsetzung der erarbeiteten Konzepte im IntelliShop-System erfolgt in Kapitel 7.

1.2 Aufbau und Vorgehensweise

Die Arbeit ist in acht Kapitel gegliedert. Nach der hier beschriebenen Vorgehensweise wird das oben formulierte Ziel schrittweise erarbeitet.

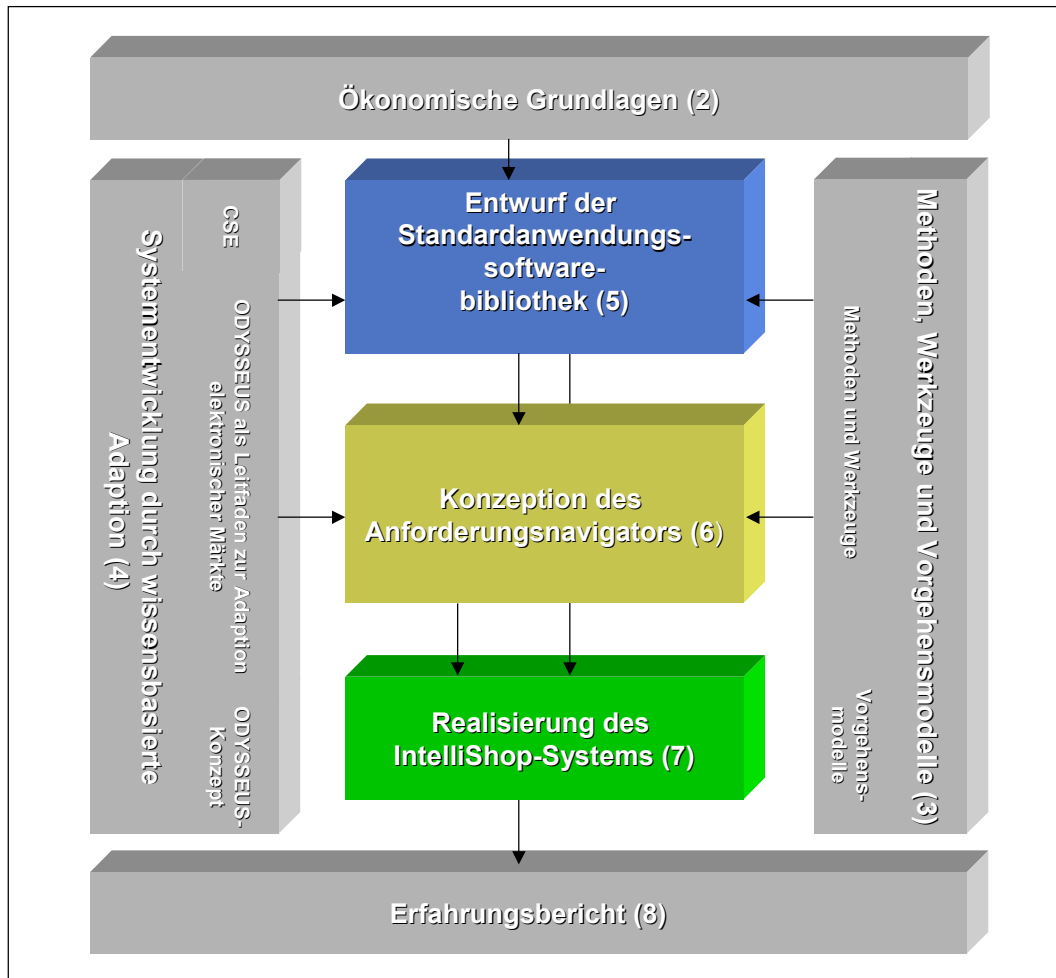


Abbildung 1-1: Aufbau der Arbeit

In Kapitel 2 werden zunächst die Potentiale der Nutzung elektronischer Märkte aufgezeigt. Damit werden die ökonomischen Grundlagen der Konzeption geeigneter Lösungen erarbeitet. Die Diskussion der Chancen und Risiken auf elektronischen Märkten orientiert sich an allgemein gültigen ökonomischen Überlegungen, sodass die Zusammenhänge deutlich werden, die von einzelnen Beispielen oder aktuellen Entwicklungen unabhängig gelten. Diese werden später zur Ableitung von Regeln für die automatische Erzeugung eines Lösungsvorschlags für ein individuelles Marktsystem durch ein wissensbasiertes Werkzeug herangezogen.

Das Kapitel 3 dokumentiert den Stand der Entwicklung von Methoden, Werkzeugen und Vorgehensmodellen zum Aufbau elektronischer Märkte und setzt sich mit den Problemen auseinander, die sich aktuell aus der Sicht von KMU bei der Realisierung eigener Lösungen ergeben. In 3.3 wird das umfangreiche Angebot an Werkzeugen zum Aufbau elektronischer Märkte nach unterschiedlichen Ansätzen der Entwicklungsunterstützung klassifiziert. Die systematische Betrachtung lässt erkennen, dass die verfügbaren Werkzeuge keine ausreichende Unterstützung für KMU bieten, zeigt aber auch, welche softwaretechnischen Konzepte für die Entwicklung einer neuen Lösung übernommen werden können. In 3.4 werden exemplarisch Vertreter der im Einsatz befindlichen Arten von Vorgehensmodellen für die Projektdurchführung vorgestellt und bewertet. Dabei wird auch deutlich, dass diese gerade in Bezug auf die Anforderungen kleiner und mittlerer Unternehmen erhebliche Nachteile aufweisen. Dieses Kapitel liefert damit wesentliche Erkenntnisse für die Entwicklung einer neuen Vorgehensweise.

In den Kapiteln 4 bis 7 wird ein neuer Ansatz zur Adaption elektronischer Märkte entwickelt. Dazu wird in Kapitel 4 zunächst das ODYSSEUS-Konzept als neue Vorgehensweise bei der Systementwicklung eingeführt und auf seine Anwendbarkeit für den Aufbau elektronischer Märkte überprüft. Auf dieser Grundlage wird in Kapitel 5 eine Standardanwendungssoftwarebibliothek konzipiert, die durch werkzeuggestützte Adaption an die individuellen Anforderungen eines Unternehmens angepasst werden kann. Anschließend wird in Kapitel 6 nach dem erweiterten ODYSSEUS-Konzept ein werkzeuggestützter Leitfaden (Anforderungsnavigator) zur Adaption dieser Standardanwendungssoftware entwickelt. In Kapitel 7 wird als Beispiel für die Umsetzung und vollständige Integration der entwickelten Konzepte die Realisierung des IntelliShop-Systems beschrieben.

Kapitel 8 geht abschließend auf die bisherigen Erfahrungen aus dem Einsatz des IntelliShop-Systems in der Praxis ein und fasst die Ergebnisse zusammen.

2 Ökonomische Grundlagen

Um später Regeln für die Ableitung sinnvoller Konzepte und die Realisierung erfolgversprechender elektronischer Märkte ableiten zu können, müssen zunächst die relevanten ökonomischen Zusammenhänge erfasst werden. Im ersten Schritt wird dazu eine Abgrenzung elektronischer Märkte und ihre Einordnung in das Umfeld des Electronic Commerce vorgenommen. Darauf aufbauend lassen sich elementare ökonomische Wirkungszusammenhänge erkennen (2.4). Diese können zur Bewertung der verschiedenen Ausprägungsformen elektronischer Märkte für den Einsatz in der Praxis aus der Sicht kleiner und mittlerer Unternehmen (2.5) und zur Darstellung der Anforderungen bei der Konzeption individueller Lösungen (2.6) herangezogen werden.

2.1 Elektronische Märkte aus transaktionskostentheoretischer Sicht

Den zentralen Ansatz zur Einordnung und Abgrenzung elektronischer Märkte liefert die Transaktionskostentheorie. Aus transaktionskostentheoretischer Sicht sind Märkte neben Hierarchien die zweite grundlegende Koordinationsform zum Austausch wirtschaftlicher Aktivitäten. In Hierarchien werden die Mechanismen zur Leistungskoordination durch Strukturen innerhalb einer Organisation definiert [COAS37]. Im Unterschied dazu erfolgt der Gütertausch auf Märkten durch indirekte wechselseitige Abstimmung zwischen grundsätzlich gleichberechtigten Akteuren [HIMB94]. Die Koordination erfolgt durch Interaktionsprozesse im Rahmen von Markttransaktionen.

Jede Markttransaktion lässt sich grob in drei Phasen unterteilen [LANG94, S. 19-21]:

Informationsphase

In der ersten Phase einer Markttransaktion steht die Suche nach Marktpartnern im Vordergrund. Dazu werden möglichst genaue und vergleichbare Informationen über angebotene Produkte und Dienstleistungen sowie wirtschaftliche oder auch technische Hintergrundinformationen über potentielle Handelspartner benötigt, die zur Entscheidung über die Abgabe eines Angebots ausgewertet werden können. Je nach Qualität, Umfang und Aggregationsniveau der vorhandenen Informationen fallen dabei Kosten in sehr unterschiedlicher Höhe an. Sie werden in der Transaktionskostentheorie als Anbahnungskosten bezeichnet. Die Informationsphase endet mit der Abgabe eines Angebots.

Vereinbarungsphase

In der Vereinbarungphase nehmen die Marktpartner Kontakt auf und handeln auf der Grundlage des Angebots der ersten Phase die genauen Bedingungen für die Abwicklung der Markttransaktion aus. Das schließt neben Verhandlungen über die Zahlungsart und Konditionen, die den Erwerb eines Gutes betreffen, auch Vereinbarungen über Sekundärtransaktionen, wie Versicherung, logistische Leistungen oder Finanztransaktionen, ein. Die Vereinbarungphase wird mit dem Abschluss einer für beide Seiten verbindlichen Verpflichtung beendet.

Abwicklungsphase

In der Abwicklungsphase findet die Erfüllung der vereinbarten Leistungen und Gegenleistungen der Handelspartner statt. Sie schließt neben der Erbringung der eigentlichen Leistung auch sekundäre Leistungen und die Kontrolle der Leistungserbringung der Gegenseite ein.

Ein elektronischer Markt entsteht, wenn Kommunikationsbeziehungen zur Abwicklung von Markttransaktionen auf elektronischen Medien abgebildet werden. PICOT bezeichnet dies als Mediatisierung von Markttransaktionen. Ein vollständig mediatisierter Markt liegt dann vor, wenn die Interaktion der Marktpartner in allen Phasen einer Transaktion über ein integriertes Informationssystem erfolgt, das auch die Preisbildung zwischen Angebot und Nachfrage unterstützt [PICO01, S. 338]. Ein vollstän-

dig mediatisierter Markt ist nur realisierbar, wenn die gehandelten Güter auf elektronischem Weg übertragbar sind.

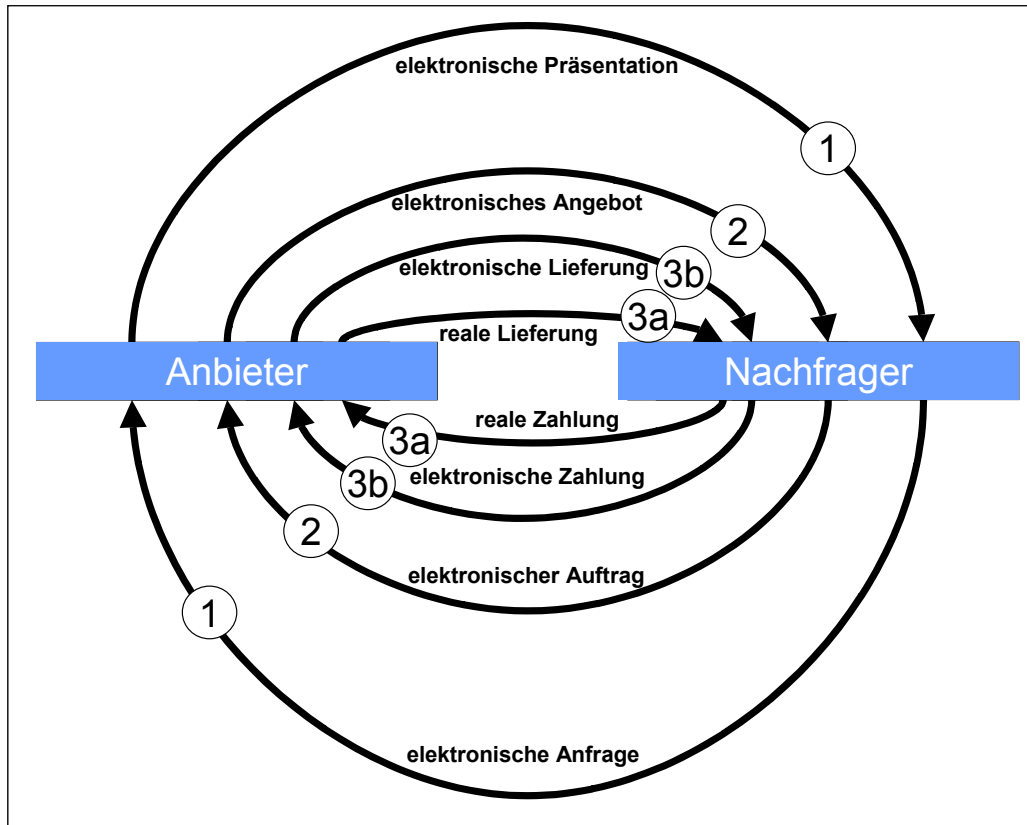


Abbildung 2-1: Interaktionen zwischen Marktteilnehmern in den einzelnen Phasen einer Markttransaktion [in Anlehnung an ZERB99, S. 148]

In diesem Fall können Informationsphase (1), Vereinbarungsphase (2) und Abwicklungsphase (3b) einer Transaktion elektronisch abgebildet werden. Dies ist nur bei vollständig digitalisierbaren oder immateriellen Gütern (z.B. Verfügungsrechten) möglich.

Beim Handel mit nicht-digitalisierbaren Produkten muss in der Abwicklungsphase zumindest eine physische Lieferung erfolgen, der häufig eine Zahlung auf klassischem Weg gegenüber steht. In der Praxis finden sich auf den elektronischen Märkten im Internet unterschiedlichste Mischformen der oben dargestellten Interaktionen zur Durchführung von Transaktionen. Nur in wenigen Fällen werden die Potentiale der elektronischen Integration vollständig ausgeschöpft. Die Gründe für die häufigen

Defizite sind meist weniger technischer als vielmehr organisatorischer und wirtschaftlicher Natur.

Der unterschiedliche Grad, in dem verschiedene Informations- und Kommunikationssysteme Transaktionen unterstützen, veranlasst SCHMID zu einer zweistufigen Definition elektronischer Märkte: „*Elektronische Märkte im engeren Sinne* sind mit Hilfe der Telematik realisierte Marktplätze...“, „die alle Phasen der Transaktion (Informationsphase, Vereinbarungsphase, Abwicklungsphase) unterstützen“ [SCHM93, S. 468]. Märkte, die dieser Definition gerecht werden, sind in der Praxis kaum anzutreffen.

„*Elektronische Märkte im weiteren Sinne* sind informationstechnische Systeme zur Unterstützung aller oder einzelner Phasen und Funktionen der marktmäßig organisierten Leistungskoordination“ [SCHM93, S. 468]. Da es sich demnach bei einem elektronischen Markt um ein auf elektronischen Medien basierendes Informationssystem handelt, werden nachfolgend elektronische Märkte auch als elektronische Marktsysteme bezeichnet [LANG94, S. 22].

Die Definition von SCHMID sagt nichts aus über Anforderungen bezüglich der Integration der einzelnen Phasen. Die Integration der einzelnen Teilkomponenten einer Lösung ist eine zentrale Forderung der Wirtschaftsinformatik. Immerhin wird in [SCHM99] als weiteres wesentliches Kriterium für einen elektronischen Markt die vollständige Abwicklung der unmittelbaren Kommunikation zwischen Nachfrager und Anbieter über elektronische Medien zum Austausch konkreter Tauschabsichten angegeben, denn erst dadurch wird ein elektronisches Medium zum Treffpunkt von Angebot und Nachfrage [SCHM99, S. 40]. Heute wird von Kunden wie Betreibern elektronischer Märkte zunehmend eine hohe Integration jedes elektronischen Marktsystems mit vorhandenen Systemen, wie etwa der Bestellabwicklung oder Fertigungsunterstützung, gefordert, denn erst durch die Integration komplexer Abläufe entstehen die wesentlichen Vorteile der Geschäftsabwicklung über elektronische Medien.

Elektronische Märkte sind durch folgende Merkmale gekennzeichnet:

Unterstützung von Koordinationsmechanismen

Im Unterschied zu Hierarchien und Unternehmensnetzwerken, wo die Koordination des Leistungsaustausches durch ein Regelwerk von Vereinbarungen, Macht- oder Weisungsbefugnissen bzw. Kooperationsvereinbarungen geregelt ist, erfolgt diese Koordination auf Märkten über die Preisbildung [ZBOR96, S. 63]. Mechanismen werden zur Zeit auf elektronischen Märkten in sehr unterschiedlichem Ausmaß unterstützt. Angebots- oder Nachfragefixierung sind Kennzeichen elektronischer Märkte, die ein Anbieter bzw. Nachfrager dominiert. Sie werden auch als proprietäre Märkte bezeichnet [HANK90, S. 60-72]. Typische Beispiele für proprietäre Märkte sind Online-Shops. Online-Auktionssysteme bieten dagegen einen halbautomatischen Mechanismus durch Ermittlung des jeweils höchsten Gebots.

Interaktivität

Aus der Anforderung an elektronische Märkte, Interaktionen zwischen mehreren Marktpartnern zu unterstützen, die in ihren Entscheidungen frei und gleichberechtigt sind, folgt die besondere Bedeutung interaktiver Benutzerschnittstellen elektronischer Marktsysteme. So hat der Internet-Dienst World Wide Web mit seinen technischen Möglichkeiten zur flexiblen Gestaltung interaktiver Benutzungsoberflächen entscheidend zur Entwicklung und Verbreitung elektronischer Märkte beigetragen.

Standortunabhängigkeit

Mit digitalen Informations- und Kommunikationssystemen lassen sich beliebige Entfernungen überbrücken. Es spielt daher aus technischer Sicht für die Marktteilnehmer keine Rolle, an welchem geographischen Ort ihre Markttransaktionen durchgeführt werden. Ihr Zugang zum Markt wird allein durch technische Restriktionen bestimmt. Für KMU entsteht dadurch die Chance, ihre Absatzmöglichkeiten geographisch nahezu unbegrenzt auszudehnen. Beim Aufbau der Infrastruktur eines internationalen elektronischen Marktes sollten jedoch länderspezifische steuer- und handelsrechtliche Vorschriften sowie unterschiedlich hohe Telekommunikationsgebühren in Betracht gezogen werden.

Offenheit

Häufig wird als weiteres Merkmal eines elektronischen Marktes die Offenheit des Systems angeführt. Darunter versteht man die freie Zugänglichkeit für alle, die sich den geltenden Regeln des Marktes unterwerfen [SCHM93, PICO98, S. 319]. Offenheit ist aber kein notwendiges Merkmal, denn auch innerhalb geschlossener Nutzergruppen können elektronische Märkte funktionieren, sofern die Zahl der Marktteilnehmer die kritische Anzahl für das Funktionieren der Marktmechanismen übersteigt. Wird der Zugang zum Markt bewusst beschränkt, können kurzfristig für Anbieter und Nachfrager erhebliche Wettbewerbsvorteile entstehen. 1976 erlangte United Airlines mit dem Aufbau eines geschlossenen ersten Flugreservierungssystems zunächst einen deutlichen Vorsprung vor der Konkurrenz. Der Erfolg wurde aber sehr schnell durch das System der Gesellschaft American Airlines übertroffen, das auch andere Anbieter zuließ, aber Flüge aus dem eigenen Angebot stets zuerst ausführte. Solche und ähnliche Beispiele zeigen, dass Konkurrenzdruck und die Präferenzen der Nachfrager langfristig meist zu einer Öffnung geschlossener Märkte führen [MALO87; MALO89]. Dank der fortschreitenden Vernetzung von Computersystemen und der hohen Verfügbarkeit standardisierter Kommunikationstechnologien ist es heute viel leichter möglich, konkurrierende Systeme zu geschlossenen elektronischen Märkten aufzubauen.

Damit verliert Offenheit als entscheidendes Merkmal für einen elektronischen Markt an Bedeutung. Wichtig für den wirtschaftlichen Erfolg ist die Teilnahme einer ausreichenden Zahl von Marktteilnehmern, die als gleichberechtigte Partner handeln. Erst wenn diese kritische Masse an Akteuren erreicht ist, können die Mechanismen des Marktes funktionieren. Die Kosten für Aufbau und Betrieb werden erst ab einer bestimmten Anzahl von Transaktionen gedeckt. Die Akzeptanz eines Marktes ist maßgeblich abhängig von der Attraktivität des Produktangebots. Auf elektronischen Märkten spielen vor allem effiziente und komfortable Funktionen zur Transaktionsabwicklung und die Qualität der Produktdarstellung eine wichtige Rolle.

2.2 Entstehung neuer Formen der Kooperation

Gründe für die Entstehung eines neuen elektronischen Marktes können nicht nur die rein elektronische Abbildung eines bestehenden, bisher nicht-elektronischen Marktes, sondern vor allem auch die Weiterentwicklung bzw. Öffnung eines Informationssystems in Richtung eines elektronischen Marktes sein. Durch die Entwicklung und Verbreitung der Informations- und Kommunikationstechnologie wird die elektronische Abwicklung von Transaktionen nicht nur innerhalb eines Unternehmens, sondern auch über Unternehmensgrenzen hinweg zunehmend effizienter möglich. Neue Technologien, die eine vielseitige, multimediale Beschreibung von Produkten und Dienstleistungen und eine verstärkte Anbindung unternehmensinterner Produktionssysteme zur Herstellung hochindividueller Erzeugnisse unterstützen, sind weitere Faktoren, die eine Verschiebung der Leistungskoordination von Hierarchien zu Märkten nach sich ziehen (Move-to-the-Market) [MALO87]. Dies führt auch zur Entwicklung von Zwischenformen der Leistungskoordination zwischen elektronischen Hierarchien und elektronischen Märkten.

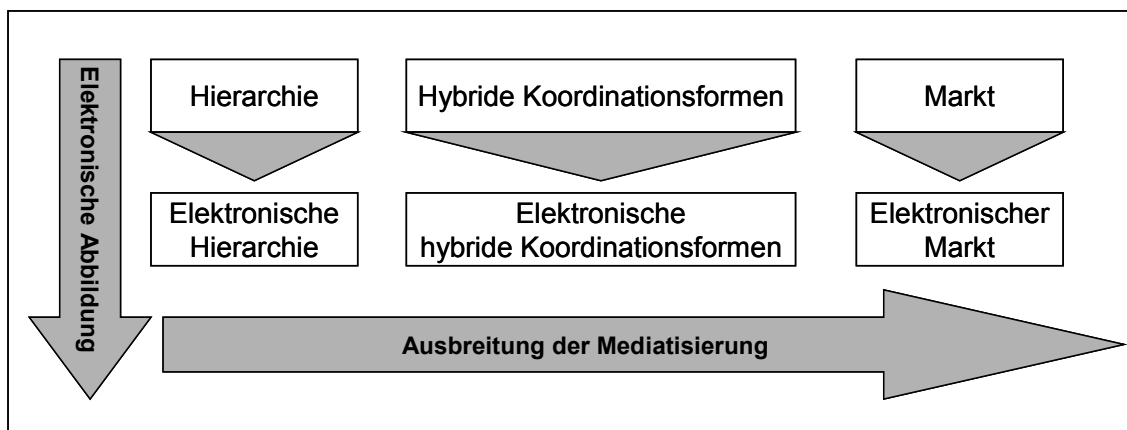


Abbildung 2-2: Entstehungsmuster elektronischer Märkte und hybrider Koordinationsformen [in Anlehnung an PICO98, S. 230]

Hybride Koordinationsformen zwischen Markt und Hierarchie werden durch zwei oder mehr rechtlich und wirtschaftlich selbstständige Unternehmen gebildet, die Verbindungen zur Nutzung von Wettbewerbsvorteilen eingehen, die innerhalb eines Unternehmens oder über den Markt nicht zu realisieren sind [BOGA95, S. 162-163].

Im Vordergrund steht beim Eingehen dieser Verbindungen nicht der Wettbewerbs-, sondern der Kooperationsgedanke. Basis dieser Verbindungen sind in der Regel langfristige enge Verbindungen und gegenseitiges Vertrauen. Meist erstrecken sie sich nur auf Teilbereiche der betroffenen Unternehmen [MÜLL03, S.20].

Im Rahmen einer Kooperation ist die Frage zu beantworten, inwieweit eine vertikale Integration zwischen den beteiligten Unternehmen sinnvoll ist. Dies hängt von verschiedenen Einflussgrößen auf die Höhe der Transaktion ab [PICO01, S. 50-54]. Danach können Kooperationsformen nach dem vertikalen Integrationsgrad geordnet und in vertikale Kooperations- und Beherrschungsformen unterteilt werden (siehe Abbildung 2-3, S. 13).

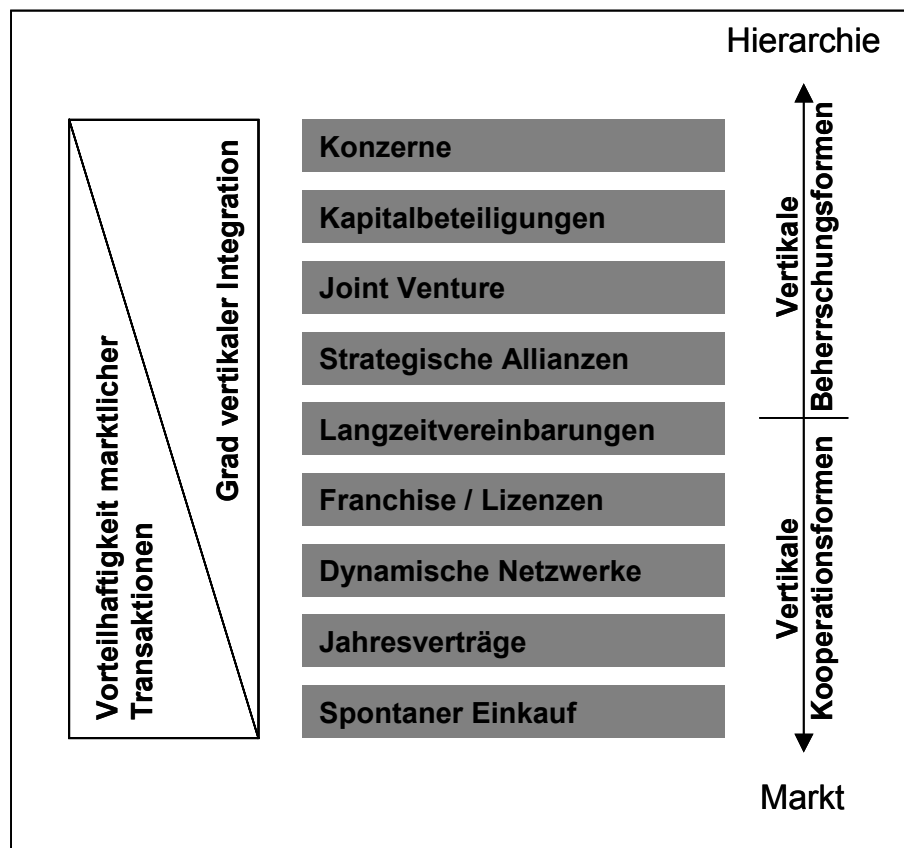


Abbildung 2-3: Beispiele für Formen der Kooperation [in Anlehnung an MÜLL03, S. 20]

Nachdem die technischen Möglichkeiten zum Aufbau vielfältiger Kooperationen über elektronische Medien gegeben sind, kann die Entscheidung für eine elektronische Form der Kooperation zunehmend aus ökonomischen Überlegungen heraus getroffen

werden. Dies führt vielfach zu einem Wandel festgefügtter Unternehmensstrukturen und Geschäftsbeziehungen. Durch die Zusammenarbeit in neuen Kooperationsformen bietet sich für Unternehmen die Chance, ihre Ressourcen schnell auszuweiten und nicht nur regionale, sondern auch internationale Unterschiede, wie z. B. bestehende Lohndifferenzen, Know-how-Gefälle und Vorteile verschiedener nationaler Rechtsordnungen flexibel nutzen zu können [PICO01, S. 327].

2.3 Elektronische Märkte im Electronic Commerce

Der Begriff „Electronic Commerce“ wird nicht selten mit jeder Abwicklung von Transaktionen über einen elektronischen Markt unmittelbar gleichgesetzt. Dies führt zwangsläufig dazu, dass die Potentiale und Probleme bei Aufbau und Nutzung elektronischer Märkte nicht genau erfasst werden. Eine Übersicht der unterschiedlichen Auffassungen von Electronic Commerce findet sich bei [DEUT98, S. 6-9]. Nach THOME und SCHINZER umfasst **Electronic Commerce** „alle Formen der digitalen Abwicklung der Geschäftsprozesse zwischen Unternehmen und zu deren Kunden über globale öffentliche und private Netze“ [THOM00, S. 1]. Diese Definition verdeutlicht die vielfältigen Einsatzgebiete von Electronic Commerce. In der Literatur werden sie häufig in die Teilbereiche Business-to-Business und Business-to-Consumer differenziert, da sich die Geschäftsabwicklung zwischen Unternehmen unter anderen Rahmenbedingungen vollzieht als Geschäfte mit Endkunden oder Konsumenten. Neben diesen dominierenden Bereichen sind weitere Anwendungsgebiete des Electronic Commerce zu unterscheiden (siehe Abbildung 2-4). Elektronische Märkte spielen in allen Bereichen als Basis unterschiedlicher Geschäftsmodelle eine zunehmend wichtigere Rolle. Für KMU besteht der Vorteil elektronischer Märkte vor allem darin, über dieses Medium eine Vielzahl von Endkunden mit neuen Produkten und Dienstleistungen erreichen zu können, die auf klassischem Weg nur zu erheblich höheren Kosten zu gewinnen wären (siehe 2.4.1).

		Nachfrager der Leistung		
		Consumer	Business	Administration
Anbieter der Leistung	Consumer	Consumer-to-Consumer z. B. Internet-Kleinanzeigenmarkt	Consumer-to-Business z. B. Jobbörsen mit Anzeigen von Arbeitssuchenden	Consumer-to-Administration z. B. Steuerabwicklung von Privatpersonen (Einkommenssteuer etc.)
	Business	Business-to-Consumer z. B. Bestellung eines Kunden in einer Internet-Shopping Mall	Business-to-Business z. B. Bestellung eines Unternehmens bei einem Zulieferer	Business-to-Administration z. B. Steuerabwicklung von Unternehmen (Umsatzsteuer, Körperschaftssteuer etc.)
	Administration	Administration-to-Consumer z. B. Internet-Kleinanzeigenmarkt	Administration-to-Business z. B. Beschaffungsmaßnahmen öffentlicher Institutionen im Internet	Administration-to-Administration z. B. Transaktionen zwischen öffentlichen Institutionen im In- und Ausland

Abbildung 2-4: Bereiche des Electronic Commerce (in Anlehnung an [HERM99, S. 23])

Für Großunternehmen bieten elektronische Märkte zusätzlich zahlreiche Möglichkeiten, auch den zwischenbetrieblichen Handel effizienter abzuwickeln. Die verschiedenen Initiativen einiger Großunternehmen zum Aufbau elektronischer Handelsplattformen zeigten, dass hier erhebliche Einsparungspotentiale erwartet werden. Ein noch relativ neuer Anwendungsbereich des Electronic Commerce ist die Abwicklung der Vergabe öffentlicher Aufträge über internetbasierte Kommunikationssysteme. Aktuelle Gesetzgebungsbestrebungen der Europäischen Union lassen erkennen, dass dieser Bereich zukünftig erheblich an Bedeutung gewinnen wird [EU00a].

Auch die zunehmende Geschäftsabwicklung über mobile Endgeräte wird von obiger Definition erfasst, da digitale Geschäftsabwicklung über öffentliche Netze auch digitale Funknetze einschließt. Die Probleme bei der Integration mobiler Endgeräte in bereits digital abwickelbare Geschäftsprozesse sind vorwiegend durch technische Ergänzungen und Anpassungen der Benutzerschnittstelle zu lösen.

In [PICO01] wird Electronic Commerce nach Koordinationsmechanismen zur Abwicklung „jeder Art wirtschaftlicher Tätigkeit auf Basis elektronischer Verbindungen“ differenziert. Elektronische Märkte sind demnach neben elektronischen Hierarchien, elektronischen Unternehmensnetzwerken und Unternehmenskooperationen

„eine institutionelle und technische Plattform für Electronic Commerce“ [PICO01, S. 317]. Die Vor- und Nachteile elektronischer Märkte gegenüber elektronischen Hierarchien werden in [MALO87] anhand transaktionskostentheoretischer Überlegungen analysiert. Andere Koordinationsmechanismen werden dort als Mischformen dieser beiden Grundtypen der Leistungskoordination betrachtet. Der Vorteil von Märkten gegenüber Hierarchien liegt danach vor allem darin, dass durch intensiven Vergleich vieler Angebote auf einem Markt ein Gut mit den gewünschten Eigenschaften zu einem nahezu minimalen Preis erworben werden kann. Gerade KMU mit schlanken Organisationsstrukturen und oft spezialisierten Produktangeboten bieten sich damit neue Chancen. Ebenso schafft die Etablierung neuer Kooperationsformen Möglichkeiten zum Aufbrechen festgefügtter Wertschöpfungsketten zwischen Unternehmen. Dies kann zur Umbildung von Wertschöpfungsketten in Wertschöpfungsnetze mit elektronischen Märkten als Knotenpunkte führen, in denen KMU verstärkt um Aufträge konkurrieren.

2.4 Ökonomische Effekte

Jedes Engagement im Electronic Commerce muss langfristig an den wirtschaftlichen Vorteilen gemessen werden, die für ein Unternehmen realisiert werden. Aus dieser Perspektive macht es kaum Sinn, zum Aufbau eines elektronischen Marktes aus Kostengründen und aufgrund fehlender eigener Erfahrung Technologien und Geschäftsmodelle anderer einfach zu übernehmen und dabei möglicherweise eigene Chancen zu vergeben. Entscheidend ist, die individuellen Vorteile elektronischer Märkte zu erkennen und umzusetzen.

Aus wirtschaftstheoretischer Sicht lassen sich grundlegende ökonomische Effekte identifizieren, die auf elektronischen Märkten auftreten. Anhand einer theoretisch fundierten Betrachtung dieser Wirkungszusammenhänge lassen sich Erkenntnisse ableiten, die unabhängig von der Entwicklung neuer Technologien, Produkte und Geschäftsmodelle Gültigkeit besitzen. Sie können zur Bewertung verschiedener Ausprägungsformen elektronischer Märkte herangezogen werden (siehe 2.5). Die Beschreibung elementarer ökonomischer Zusammenhänge ist schließlich die Grundlage für die Abbildung allgemein gültiger Regeln in einem wissensbasierten System, das die

Konzeption eines elektronischen Marktes nach den individuellen Anforderungen eines Unternehmens wissensbasiert unterstützt (siehe 6.1.2).

Die ersten elektronischen Märkte wurden vorrangig zum Austausch von Informationen zwischen einer meist eng begrenzten Zahl von Marktteilnehmern eingerichtet. Motivation für den Aufbau war die Einsparung von Transaktionskosten. Bereits 1987 beschreiben MALONE et al. neben dieser als Kommunikationseffekt bezeichneten Auswirkung der Mediatisierung von Markttransaktionen zwei weitere Effekte, die zu nachhaltigen Veränderungen der wirtschaftlichen Austauschprozesse [MALO87, S. 497] führen. Die dort noch vage umschriebenen Zusammenhänge sind heute an vielen Beispielen aus der Praxis zu beobachten. Mit der Verbreitung elektronischer Märkte treten zudem immer deutlicher Netzeffekte auf, die zur Entwicklung völlig neuer Marktmechanismen führen.

Die nachfolgende Darstellung beschränkt sich nicht nur auf eine theoretische Einordnung und Begründung. Die beobachtbaren Auswirkungen der ökonomischen Effekte auf elektronischen Märkten werden darüber hinaus an Beispielen aufgezeigt.

2.4.1 Kommunikationseffekt

Der Kommunikationseffekt beschreibt zwei Auswirkungen der Mediatisierung auf den Austausch von Informationen [MALO87, S. 488]:

- (1) Informationen können in kürzerer Zeit bzw. in der gleichen Zeit mehr Informationen verarbeitet werden.
- (2) Die Kosten für den Informationsaustausch können stark gesenkt werden.

Für KMU eröffnet der Kommunikationseffekt die Chance, mit vergleichsweise geringem Investitionsrisiko neben den Global Players auf weltweiten Märkten präsent zu sein. Die Anforderungen, die mit der Erschließung neuer Märkte und Zielgruppen an viele Bereiche des Unternehmens (z. B. Vertrieb, Logistik) gestellt werden, lassen sich mit Unterstützung technischer Systeme alleine aber kaum bewältigen, sondern erfordern umfangreiche Anpassungen bestehender Organisationsstrukturen und Abläufe. Hinzu kommt, dass der Kommunikationseffekt auch Veränderungen auf den Märkten bewirkt, auf denen Unternehmen bereits etabliert sind. Die höhere Geschwindigkeit des Informationsaustausches und die Senkung der Transaktionskosten

führen allgemein zu einer erhöhten Informationstransparenz. Auf Märkten mit geringer Informationstransparenz können Anbieter ihren Informationsvorsprung bezüglich des Angebots nutzen, um bei geringen Absatzmengen einen Marktpreis zu realisieren, der über dem minimalen Angebotspreis liegt. Auf elektronischen Märkten schwindet dieser Vorteil, denn dort können Nachfrager in relativ kurzer Zeit umfangreiche Angebots- und Preisvergleiche durchführen. Dies führt zu einer Senkung des Marktpreises in Richtung des minimalen Angebotspreises (siehe Abbildung 2-5).

In der Vergangenheit wurde dies vor allem beim elektronischen Handel mit standardisierten, gut vergleichbaren Gütern deutlich. Beispiele sind der elektronische Wertpapier- und Terminhandel. Mit wachsender Verfügbarkeit der Kommunikationstechnik und spezieller Lösungen für Angebotsvergleiche im Internet wirkt sich der Kommunikationseffekt aber auch auf Märkte für weniger standardisierte Güter aus.

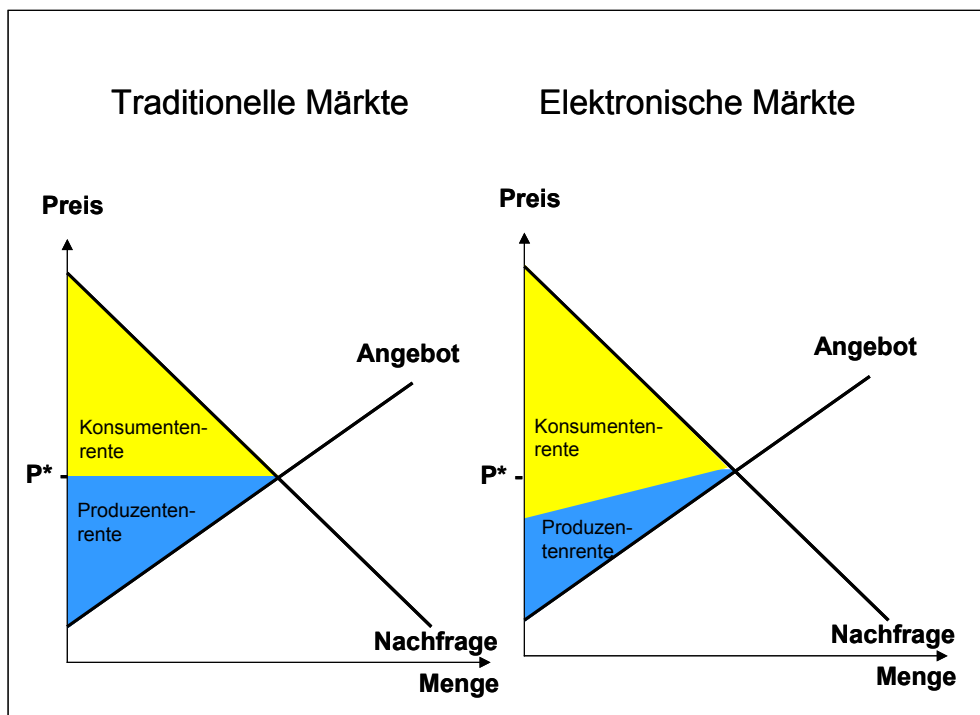


Abbildung 2-5: Effekt höherer Informationstransparenz auf die Preisbildung [ZERB99, S. 152]

Eine Befragung, die Gemini Consult in Zusammenarbeit mit der Universität Freiburg 1998 unter Fach- und Führungskräften deutscher Unternehmen durchführte, kam zu dem Ergebnis, dass das Internet bereits in großem Umfang zu Preis- und Ange-

botsvergleichen genutzt wird. Die befragten Personen erwarteten deshalb eine Verschärfung des Wettbewerbs und eine Abnahme der Kundenloyalität [SCHO98]. An der zunehmend wichtigeren Rolle, die Preisagenturen und Auktionsplattformen im Internet bei der Angebotsauswahl spielen, ist dies bereits nachvollziehbar.

Die Angst vor Angebotsvergleichen hält bisher einige Unternehmen davon ab, Preisinformationen im Internet zu veröffentlichen. Eine vollständig elektronische Abwicklung wird damit unmöglich. Langfristig muss diese Strategie zu Nachteilen gegenüber Anbietern führen, die alle Potentiale elektronischer Märkte effektiv nutzen. Auch die Strategie, durch undurchsichtige Preisstrukturen Angebotsvergleiche zu erschweren, ist wenig erfolgversprechend. Einige Handelsmittler haben sich bereits auf die gezielte Analyse derartiger Angebote spezialisiert (z. B. www.billiger-telefonieren.de). Empfehlenswert sind Strategien, die trotz Angebotsvergleich ein Abwandern der Kunden verhindern, wie beispielsweise die Gewährung von Rabatten für Wiederholkäufer oder spezielle Zusatzleistungen für Stammkunden.

2.4.2 Integrationseffekt

Integrationseffekte entstehen, wenn bisher getrennt abgewickelte, aber aneinander grenzende Prozesse entlang einer Wertschöpfungskette durch Mediatisierung integriert werden [MALO87, S. 488]. Der Einsatz von EDIFACT- oder ähnlichen Systemen zur Automatisierung des Datenaustausches in fest definierten Geschäftsbeziehungen zwischen Großunternehmen und ihren Zulieferern ist ein bekanntes Beispiel für die Nutzung des Integrationseffektes innerhalb von hierarchischen Organisationsstrukturen.

Elektronische Märkte bieten dagegen die Chance, Integrationseffekte auch in multilateralen Lieferantenbeziehungen zu nutzen und/oder Kunden direkt in die Informationsverarbeitung und Organisation eines Unternehmens zu integrieren. Voraussetzung für die Nutzung von Integrationseffekten ist eine genaue und zumeist aufwändige Abstimmung und Integration aller betroffenen organisatorischen Abläufe und technischen Systeme. Dem Integrationsaufwand bei der Implementierung solcher Lösungen stehen langfristig hohe Kosteneinsparungspotentiale entlang der gesamten Wertschöpfungskette gegenüber. Die eingesparten Kosten können zur Erhöhung der Gewinnspanne und/oder zur Senkung von Preisen verwendet werden.

Die Integration von Kunden durch digitalen Vertrieb über elektronische Märkte hat direkten Einfluss auf Vertriebs- und Logistikkosten (siehe Abbildung 2-6), die zum Teil vollständig entfallen. Auch Prozesse im Marketing, in der Bestellabwicklung oder auch die Beschaffungsprozesse selbst können über elektronische Märkte häufig sehr viel effizienter abgewickelt werden. Verglichen mit Werbeaktionen über klassische Werbemedien sind die Kosten für ein werbewirksames Angebot im Web oft relativ gering. Die Abwicklung von Bestellungen über elektronische Märkte führt bei vollständiger Integration nicht nur zur erheblich schnelleren Abwicklung, sondern auch zur Vermeidung von Fehlern.

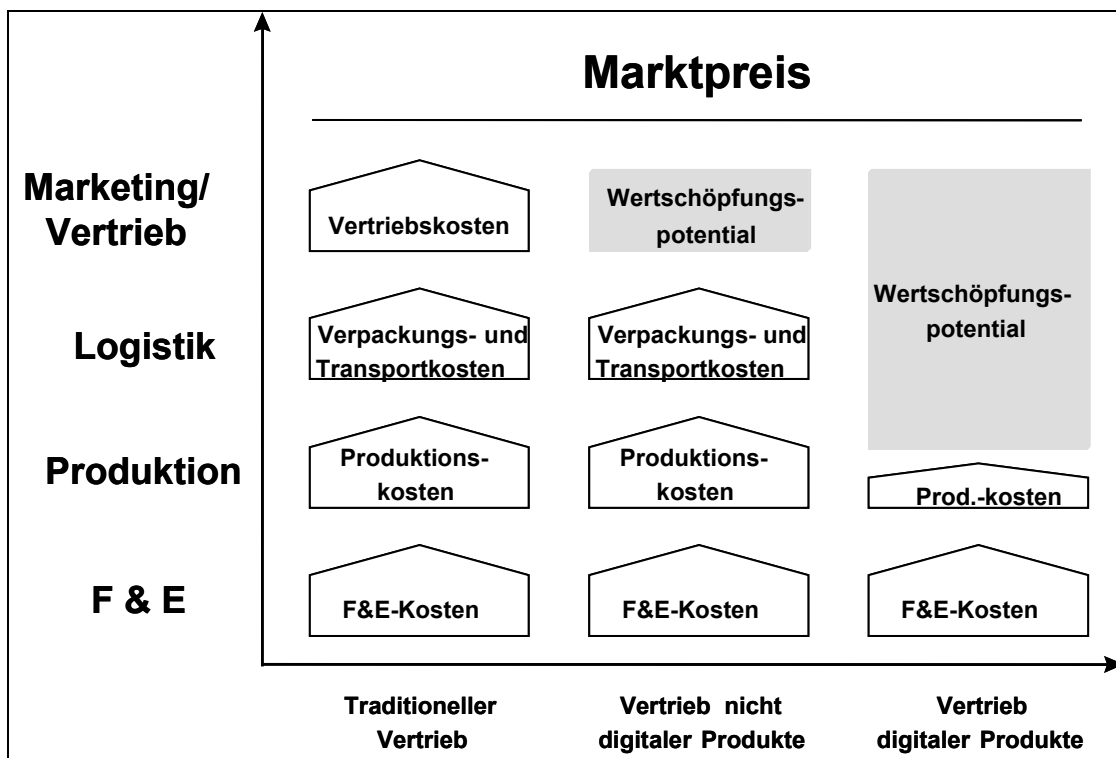


Abbildung 2-6: Wertschöpfungspotentiale durch digitalen Vertrieb über elektronische Märkte (in Anlehnung an [THOM00, S. 9])

Voraussetzung für die Nutzung von Integrationseffekten ist eine unmittelbare Anbindung des elektronischen Marktes an die Informationsverarbeitung des Anbieterunternehmens. Die bisher verfügbaren Werkzeuge unterstützen diese meist nur in begrenztem Umfang. Von Seiten der Kunden wird die Integration von Online-Angeboten allerdings in zunehmendem Maß gefordert, denn sie ist nicht nur Voraussetzung für eine schnelle und fehlerfreie Bestellabwicklung, sondern auch für die Bereitstellung

zusätzlicher Informationen über die Verfügbarkeit von Produkten. Beides stellt für Kunden einen wichtigen und heute vielfach schon als selbstverständlich erachteten Mehrwert elektronischer Märkte dar. Zudem sind Unternehmen aufgrund der höheren Preistransparenz, die durch elektronische Märkte entsteht (siehe 2.4.1), zunehmend gezwungen, Effizienzgewinne und strategische Potentiale durch Integrationseffekte zu nutzen, um im Wettbewerb zu bestehen.

2.4.3 Handelsmittlereffekte

Handelsmittler, auch als Intermediäre oder Makler bezeichnet, sind Institutionen, die den Austausch von Gütern auf Märkten organisieren [ZBOR96, S. 104]. Handelsmittlereffekte treten auf, wenn die Funktionen von Handelsmittlern von elektronischen Systemen übernommen werden. Anhand transaktionskostentheoretischer Überlegungen kann gezeigt werden, dass die verstärkte Mediatisierung von Handelsmittlerfunktionen sowohl zur Eliminierung etablierter handelsvermittelnder Institutionen (Disintermediation) als auch zur Entstehung ganz neuer Handelsmittler (Reintermediation) führt [SARK95].

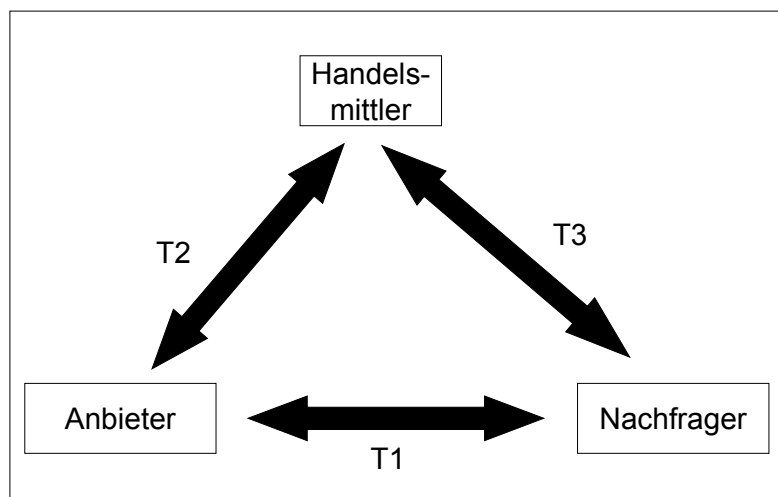


Abbildung 2-7: Transaktionen zwischen Anbietern, Nachfragern und Handelsmittlern

Aus transaktionskostentheoretischer Sicht ist die Einschaltung eines Handelsmittlers dann ökonomisch sinnvoll, wenn dadurch die Kosten für den Güteraustausch insgesamt gesenkt werden. Wenn T1, T2 und T3 in Abbildung 2-7 jeweils den Trans-

aktionskosten für den Gütertausch zwischen den angeführten Transaktionspartnern entsprechen, ist dies genau dann der Fall, wenn gilt:

$$T1 > T2 + T3$$

Intermediäre, die zur Ausübung ihrer Funktion Märkte aufbauen, treten als Marktorganisateur (MarketMaker) auf, um Transaktionsbeziehungen zu vereinfachen.

Ein Intermediär kann bei der Vermittlung zwischen Angebot und Nachfrage eine Reihe von Funktionen wahrnehmen.

Tabelle 2-1: Handelsmittlerfunktionen und Handelsmittlersysteme

Funktionen von Handelsmittlern	Systeme zur Unterstützung von Handelsmittlerfunktionen
Informationsgewinnung, Kontaktaufnahme	Suchmaschinen, Preisinformationsdienste, Lieferantenverzeichnisse
Absatzförderung, Beschaffungsunterstützung	Online-Shops, Online-Malls, Online-Auktionssysteme
Matching (Anpassung des Angebots an die Nachfrage)	Elektronische Einkaufskataloge
Verhandlungen	Agentensysteme
Physische Distribution	Online-Tracking-Systeme
Finanzierung	Online-Broking

Einige dieser Funktionen können elektronisch vollständig oder teilweise automatisiert in großer Zahl abgewickelt werden (siehe Tabelle 2-1). Die Gewinnung von Informationen, die Kontaktaufnahme sowie Marketingaktivitäten zur Absatzförderung lassen sich vollständig auf elektronischen Märkten verschiedenster Ausprägungsform abwickeln. Andere Funktionen können zumindest in wesentlichen Teilen unterstützt und damit effizienter ausgeführt werden. So versetzt der Aufbau eigener elektronischer Märkte viele Unternehmen in die Lage, diese Funktionen selbst effizienter durchzuführen als über Handelsmittler (siehe Fall 2 in Abbildung 2-8). Klassische Vertriebsstrukturen werden dadurch aufgelöst (Disintermediation). In [BENJ95, S. 67] wird am Beispiel der Textilindustrie gezeigt, dass durch die Ausschaltung von Handelsstufen erhebliche Einsparungen möglich sind. Bei DELL [DÖRF99] ist der direkte Vertrieb zum Kunden über den elektronischen Markt das Geheimnis des Er-

folges. Für Unternehmen, die mit Gütern handeln, die sich gut für den elektronischen Vertrieb eignen, kann diese Entwicklung eine akute Bedrohung bedeuten. Sie sind gezwungen, mit neuen Unternehmensstrategien zu reagieren.

Unter der Annahme, dass alle Transaktionen digital und zu minimalen Kosten abgewickelt werden könnten, würde gelten:

$$T^* = T1' = T2' = T3' \Rightarrow T2' + T3' = 2 T1' > T1$$

Hierbei bezeichnen $T1'$, $T2'$ und $T3'$ die Transaktionskosten für die digital durchgeführten Transaktionen und T^* das Kostenminimum. Wenn nach der obigen Annahme alle Transaktionskosten diesem Minimum entsprechen, würde das die vollständige Ausschaltung aller Handelsmittler bedeuten. In der Praxis fallen aber auch auf elektronischen Märkten, beispielsweise für Angebotssuche und -vergleich, vielfach Transaktionskosten an. Bei der Untersuchung, welche Auswirkungen die verstärkte Abwicklung von Transaktionen über elektronische Medien hat, lassen sich vier Fälle unterscheiden, die in Abbildung 2-8 dargestellt sind.

		Klassische Abwicklung	
		$T1 < T2 + T3$	$T1 > T2 + T3$
Elektronische Abwicklung	$T1' < T2' + T3'$	① Klassischer Direktvertrieb wird elektronisch abgewickelt	② Funktionen der Handelsmittler werden von Anbietern selbst elektronisch abgewickelt (Disintermediation)
	$T1' > T2' + T3'$	③ Neue Handelsmittler entstehen (Reintermediation)	④ Handelsmittler wickeln ihre Funktionen elektronisch ab

Abbildung 2-8: Auswirkungen elektronischer Transaktionsabwicklung auf Handelsmittler (in Anlehnung an [SARK95])

In den Fällen (1) und (4) werden die bisher schon im Unternehmen durchgeführten Aktivitäten mit Geschäftspartnern und Kunden durch eine effizientere elektronische

Abwicklung ersetzt. Der Anreiz zur verstärkten elektronischen Abwicklung liegt in den Rationalisierungs- und Wertschöpfungspotentialen durch konsequente Nutzung von Integrationseffekten. Die rasante Verbreitung elektronischer Kommunikationssysteme zwingt viele Unternehmen dazu, diese Potentiale weitestgehend auszuschöpfen, um gegenüber der Konkurrenz bestehen zu können. Eine Bedrohung der Handelsmittler durch Disintermediation liegt dann vor, wenn ihre bisher erbrachten Leistungen auf elektronischem Weg deutlich günstiger bereitgestellt werden können. Diese Unternehmen sind gezwungen, neue Vertriebsstrategien zu entwickeln. Sie können durch Zusatzleistungen des Handels, wie Sortimentsbildung oder Beratungsleistung, ihre Leistung aufwerten und durch den Einsatz elektronischer Medien so günstig wie möglich anbieten (Fall 4). Viele Handelsunternehmen haben die Flucht nach vorne angetreten und sind heute im elektronischen Handel führend. Eine Spitzenposition nehmen Versand-, Buch- und CD-Händler ein, da ihre stark standardisierten Produkte sich über elektronische Medien gut vertreiben lassen [EYGO01, S. 5]. Alternativ können diese Unternehmen ihre Erfahrungen nutzen, um völlig neue Dienstleistungen auf elektronischem Weg anzubieten (Fall 3).

Im Fall 3 entstehen neue Anbieter von Dienstleistungen, die den elektronischen Handel unterstützen. Sie können entweder selbst als Marktplatzorganisator oder als Anbieter komplementärer Dienstleistung zur elektronischen Transaktionsabwicklung auftreten. Die Leistung des Marktorganistors ist es, Informationen durch spezielle Marktveranstaltungen für eine bestimmte Zielgruppe von Marktteilnehmern zu organisieren und zu strukturieren [MOUG97, S. 140f.]. Entscheidend für den Erfolg dieser Handelsmittler ist der Mehrwert ihrer Dienstleistung für den einzelnen Nutzer. Die derzeit größten Verzeichnisdienste bieten deshalb die Option zur Personalisierung von Informationsangeboten (z. B. MyYahoo). Damit wird nicht nur die Qualität des Angebots für den einzelnen Kunden verbessert, sondern es werden auch Lock-In-Effekte gefördert, die zu einer starken Kundenbindung führen (siehe 2.4.4).

Andere Unternehmen nutzen ihre Kenntnisse über Produkteigenschaften und Präferenzen ihrer Kunden zum Aufbau von elektronischen Märkten mit speziellen Sortimenten und differenzierten Businessmodellen. Beispiele sind themenorientierte Online-Shops oder Online-Malls mit spezifischen Angeboten für eine Region wie der Virtuelle Weinladen (www.virtualwine.com) oder die Elektronik Mall Bodensee (www.emb.net).

Anbieter komplementärer Dienstleistungen konzentrieren sich auf Funktionen des Handels, die nicht oder nur unter speziellen Voraussetzungen auf elektronischem Weg abzuwickeln sind. Zu ihnen zählen Logistikdienstleister und die Anbieter von Verschlüsselungs-, Zertifizierungs- und Bezahlvorgängen. Elektronische Zertifizierungs- und Bezahlvorgänge sind nur mit speziellen und recht aufwändigen Verfahren, oft unter Einschaltung Dritter, durchzuführen. Für ein einzelnes Unternehmen ist die Bereitstellung und Koordination dieser Leistungen im Allgemeinen mit erheblich höheren Kosten verbunden als die Einschaltung eines spezialisierten Handelsmittlers. Als Komplementärprodukte auf elektronischen Märkten sind vorwiegend Produkte und Dienstleistungen interessant, die nahtlos und zu geringen Kosten in die Prozesse der digitalen Geschäftsabwicklung zu integrieren sind. Hohe Kosten für Integration und Betrieb von Fremdprodukten für zentrale Funktionen führen schnell zur Ineffizienz mehrerer Geschäftsprozesse. Die gängigen elektronischen Bezahlvorgänge sind deshalb auf allen führenden Systemplattformen verfügbar und über Standardschnittstellen in jede Web-Anwendung zu integrieren. So profitieren die Anbieter von Komplementärprodukten (siehe 2.4.4) ebenso wie die Organisatoren elektronischer Märkte durch positive Netzeffekte von steigenden Nutzerzahlen im Netz. Für viele etablierte Handelsmittler stellt diese Entwicklung eine Bedrohung dar, die sie dazu zwingt, über eine möglichst effektive Nutzung elektronischer Medien oder die Umsetzung ganz neuer Unternehmensstrategien nachzudenken. In Anbetracht der derzeitigen Wachstumsgeschwindigkeit des globalen Netzes könnte der Bedarf an neuen Intermediären sogar die Zahl der Handelsmittler übertreffen, die durch das Internet verdrängt werden [ZERD99, S. 150].

2.4.4 Netzeffekte

Netzeffekte treten in Strukturen auf, die eine große Anzahl von Wirtschaftssubjekten direkt oder indirekt miteinander in Beziehung setzen. Sie sind nicht auf Kommunikationsnetzwerke beschränkt und können ebenso innerhalb vielfältig vernetzter wirtschaftlicher Beziehungen wirksam werden. Mit dem Internet entsteht somit ein Wirtschaftsraum, in dem Netzeffekte eine herausragende Rolle spielen.

Netzeffekte liegen vor, wenn der Nutzen eines Gutes sich mit der Anzahl der Konsumenten bzw. Anwender ändert [ZERB99, S. 155]. Sie können in direkte und indirekte Netzeffekte unterschieden werden [KATZ85]. Bei **direkten Netzeffekten** steigt der

Wert einer Leistung unmittelbar mit der Zahl ihrer Nutzer. Ein typisches Beispiel sind Kommunikationsdienste in Virtuellen Gemeinschaften. Je mehr Nutzer in einer Virtuellen Gemeinschaft aktiv miteinander kommunizieren, desto attraktiver ist sie für andere. Wichtig zur Förderung direkter Netzeffekte ist die Realisierung von Kompatibilitäten zwischen Elementen bzw. den Nutzern der angebotenen Leistung.

Indirekte Netzeffekte entstehen, wenn zu einer Netzleistung Komplementärleistungen oder Zusatzprodukte angeboten werden. Die Zahl der Nutzer eines Gutes erhöht den Anreiz zur Entwicklung und Verbesserung von geeigneten Komplementärleistungen. Durch die Bereitstellung standardisierter Schnittstellen und die Förderung von Kooperationen wird dieser Anreiz zusätzlich erhöht. Die Zahl der angebotenen Komplementärleistungen steigert wiederum indirekt den Wert der Netzleistung.

Sowohl bei direkten als auch bei indirekten Netzeffekten steigt der Wert eines Gutes mit der Zahl seiner Nutzer in einem sich selbst verstärkenden Effekt (siehe Abbildung 2-9). Dieser Zusammenhang wird als **positives Feedback** bezeichnet [ZERB99, S. 157].

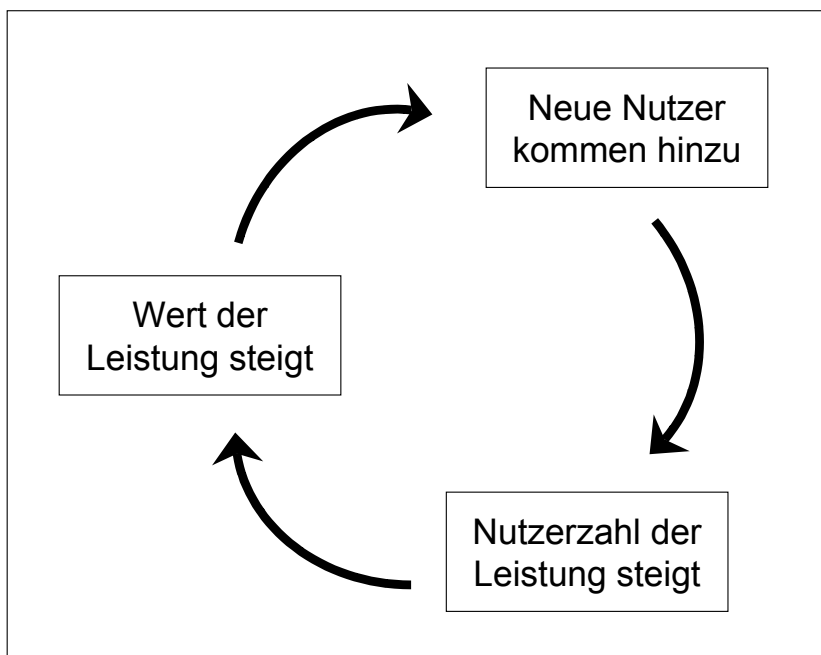


Abbildung 2-9: Wirkung positiver Feedback-Effekte [ZERB99, S. 158]

Positive Feedback-Effekte können bis zur Ausbildung natürlicher Monopole führen. So lange das Wachstum der Nutzerzahlen nicht zu negativen Skaleneffekten auf der

Anbieterseite führt und die Qualität der angebotenen Leistungen leidet, wächst mit jedem neuen Nutzer die Attraktivität des Angebots.

Ein Anbieter im Internet, der direkte Netzeffekte sehr geschickt nutzt, ist der virtuelle Buchhändler Amazon. Kunden von Amazon können zu jedem Buch selbst eine Bewertung abgeben und eine eigene Rezension verfassen. Andere Nutzer profitieren bei der Buchauswahl von diesen Erfahrungen. Gleichzeitig forciert Amazon indirekte Netzeffekte durch den Aufbau von Allianzen mit bekannten Recherchediensten im Internet. Dort erhält der Nutzer bei jeder Recherche sofort einen Verweis auf geeignete Bücher zu dem gesuchten Begriff aus dem Sortiment von Amazon. Dieser Anbieter schafft damit eine attraktive Komplementärleistung zu der Leistung von Suchmaschinen. Aufgrund dieser Strategien ist Amazon mittlerweile zum führenden Buchhändler im Internet geworden [SCHM99a].

Durch indirekte Netzeffekte kann es gelingen, Kunden langfristig auch entgegen massivem Konkurrenzdruck zu binden. Wenn mit der Nutzung eines Netzprodukts Investitionen in den Erwerb von Komplementärprodukten oder den Aufbau von produkt-spezifischem Know-how verbunden sind, entstehen beim Wechsel auf ein Konkurrenzprodukt zusätzliche Kosten. Dies kann dazu führen, dass Kunden sich langfristig auf Nutzung eines erkennbar inferioren Gutes festlegen (Lock-In-Situation) [LIEB94].

Die Wirkung von Netzeffekten hängt nicht allein von der Qualität und der tatsächlichen Entwicklung der Nutzerzahlen eines Netzwerkproduktes ab. Bei der Beurteilung einer Netzleistung spielen die Erwartungen der Nutzer bezüglich der zukünftigen Entwicklung eine entscheidende Rolle. Negative Erfahrungen führen zu Vertrauensverlusten. Positive Feedback-Effekte verkehren sich dann schnell ins Gegenteil. So kann selbst für etablierte Angebote beispielsweise von technologischen Neuerungen eine Bedrohung ausgehen. Wird ein Online-Angebot über längere Zeit nicht an den aktuellen technischen Stand angepasst, kann dies als Hinweis für geringe Innovationsbereitschaft ausgelegt werden. Hohe Imageverluste und die Abwanderung von Kunden zu Konkurrenten sind u. U. die Folge. Das zwingt die Anbieter auf elektronischen Märkten im Internet, auf Neuerungen und Weiterentwicklungen von Konkurrenzangeboten zu reagieren.

2.5 Formen des Electronic Commerce und ökonomische Effekte

Obwohl neue Ideen und Geschäftsmodelle zur Entstehung elektronischer Märkte in den unterschiedlichsten Formen geführt hat, lassen sich dennoch grundlegende Ausprägungsformen identifizieren [SCHI98; TIMM98]. Sie sind auf verschiedene Weise zur Nutzung der dargestellten ökonomischen Effekte geeignet.

Tabelle 2-2: Übersicht der Möglichkeiten zur Nutzung ökonomischer Effekte in Abhängigkeit der Ausprägungsformen elektronischer Märkte

	Kommunikationseffekt	Integrations-effekt	Handelsmittler-effekte	Netzeffekte
Online-Shops	Erst durch Vermarktung des Shops innerhalb des Marketing-Mix werden viele Kunden erreicht.	Anbieterspezifisches Angebot ermöglicht hohe Integration.	Disintermediation kann durch Übertragung eigener Stärken in individuelle Geschäftsmodelle verhindert werden.	Netzeffekte werden erst durch Personalisierung (Lock-In-Effekte) oder durch Kombination mit anderen Marktformen erreicht.
Online-Malls	Jede Vermarktungsaktion innerhalb der Mall kommt allen zugute (Economies of Scale).	Anbindung mehrerer Anbieter an eine Plattform erschwert Integration.	Kostengünstige Einstiegsmöglichkeit, um Handelsmittlereffekte zu realisieren.	Durch geeignete Auswahl der Anbieter können Netzeffekte entstehen.
Online-Auktionen	Wirkungsvolles Instrument, um breite Kundenz Kreise zu aktivieren.	Vollständige Integration aller administrativen Tätigkeiten wichtig, da geringer Ertrag pro Transaktion.	Güter, für die ein Preis schwer zu ermitteln ist, können zu geringen Transaktionskosten vermittelt werden.	Kritische Masse an Teilnehmern ist entscheidend für das Funktionieren des Preisbildungsmechanismus.
Virtuelle Gemeinschaften	Umfassende Kommunikationsdienste zu geringen Kosten steigern die Attraktivität für Nutzer. Die Nutzer stellen Großteil der Inhalte selbst.	Hohe Integration aller Services ist Voraussetzung für effektive Kommunikation zu niedrigen Kosten.	Die Leistung für die Bereitstellung der Plattform kann durch direkte oder indirekte Einnahmen abgeschöpft werden.	Netzeffekte sind erfolgsentscheidend.

Aus Sicht kleiner und mittlerer Unternehmen sind vor allem Online-Shops, Online-Malls, Online-Auktionen und Virtuelle Gemeinschaften interessant. Daneben gewinnen elektronische Beschaffungssysteme zunehmend an Bedeutung. Die Integration von Beschaffungsprozessen stellt jedoch sehr hohe Anforderungen, die bisher nur Großunternehmen bewältigen, die über die notwendigen technischen Voraussetzungen zum Aufbau einer eigenen Beschaffungsplattform und über eine ausreichend große Anzahl von Lieferanten verfügen.

Eine Vielzahl von Softwareherstellern bietet Produkte zur Realisierung der angeführten Marktformen mit entsprechenden betriebswirtschaftlichen Referenzmodellen an (siehe 3.3.2.3). Die Übersicht zeigt, dass sich mit den angeführten Ausprägungsformen Vorteile ganz unterschiedlicher Art erzielen lassen. In der Praxis ist die Kombination mehrerer Ausprägungsformen zu einem Angebot üblich.

2.5.1 Online-Shops

In einem Online-Shop bietet ein einzelnes Unternehmen seine Waren und Dienstleistungen über elektronische Medien zum Verkauf an [SCHI98, S. 20f.]. Der Vorteil von Online-Shops liegt in der uneingeschränkten Freiheit und Unabhängigkeit, soweit es die Umsetzung individueller Geschäftsideen und Angebote betrifft. Die Chance, ihre individuellen Stärken im Internet auszuspielen und ihren Vertrieb in einer überschaubaren Lösung auf einem elektronischen Markt abzuwickeln, macht Online-Shops für viele KMU besonders attraktiv. Allerdings müssen in jedem Online-Shop alle Funktionen des elektronischen Produktvertriebs vollständig implementiert werden. Konzeption und Realisierung eines individuellen Online-Shops sind deshalb mit erheblichem Aufwand verbunden. Dies hat zur Entstehung einer Vielzahl standardisierter Produkte geführt.

Die Funktionalität innovativer Online-Shops geht heute weit über die eines elektronischen Produktkatalogs mit integriertem Bestellsystem hinaus. Vielfach werden Funktionen zur Personalisierung bzw. Produktindividualisierung implementiert.

Mit wachsender Zahl unabhängiger Online-Shops steigt aber auch die Intransparenz der Angebote. Ähnlich wie klassische Ladengeschäfte sind auch Online-Shops stark von zusätzlichen Vermarktungsmaßnahmen über klassische Medien abhängig. Um

Kunden im Netz zu gewinnen und zu binden, bietet sich die Kombination mit Online-Auktionen oder Virtuellen Gemeinschaften an.

2.5.2 Online-Malls

In einer Online-Mall schließen sich mehrere Anbieter unter einer gemeinsamen Organisationsstruktur zusammen, um durch die Zentralisierung von Funktionen Skaleneffekte zu erzielen. Ein weiterer Grund für den Aufbau einer Online-Mall ist die Nutzung positiver Netzeffekte durch Integration und gemeinsame Vermarktung der Sortimente verschiedener Anbieter. Das setzt voraus, dass dem Kunden die Zielrichtung und Angebotsstruktur unmittelbar deutlich wird. Online-Malls ohne klare Konzeption werden schnell unübersichtlich. Diese Erkenntnis hat in der Vergangenheit zur Bildung regionaler branchenspezifischer und themenorientierter Online-Malls geführt. Aus technischer Sicht sind Online-Malls sehr gut geeignet, um durch gemeinsame Nutzung einer Kommunikationsinfrastruktur durch mehrere Unternehmen erhebliche Skaleneffekte zu erzielen [ZIMM95, S. 38].

Gerade für KMU, die den Aufbau einer eigenen Kommunikationsinfrastruktur nicht finanzieren können und alleine nicht über ausreichende Marktmacht für eine erfolgreiche Vermarktung im Internet verfügen, ist die Teilnahme an einer Online-Mall oft eine kostengünstige und erfolgversprechende Alternative zum eigenen Online-Shop.

2.5.3 Online-Auktionen

Online-Auktionen kommen der idealisierten Form eines vollständig mediatisierten elektronischen Marktes am nächsten. Der Abgleich zwischen Angebot und Nachfrage wird hier weitgehend elektronisch unterstützt. Im Gegensatz zu anderen Marktformen wird bei Auktionen der Preis erst im Laufe eines Bietprozesses von der Käuferseite bestimmt. Der Anbieter definiert lediglich die Regeln für diesen Prozess [REYN96]. Nach den Regeln für den Bietprozess können mehrere Auktionsformen unterschieden werden. Die Bekanntesten sind die Englische und die Holländische Auktion sowie die Höchstpreisauktion und die Vickrey-Auktion. Bei allen genannten Auktionsformen erhält jeweils das höchste Gebot den Zuschlag. Die Unterschiede liegen in den

Regeln zur Abgabe von Geboten und der Bestimmung des Kaufpreises [KRÄK92, S. 9-13].

Entscheidend für die Effizienz der Preisbildungsmechanismen von Online-Auktionssystemen ist eine ausreichend große Anzahl von Nachfragern. Die Eigenschaften der gehandelten Güter müssen für einen großen Nutzerkreis transparent gemacht werden oder weitgehend standardisiert sein. Online-Auktionssysteme bieten eine interessante Möglichkeit, um Güter, für die kein festgelegter Preis existiert, über einen längeren Zeitraum weltweit anzubieten und zum bestmöglichen Marktpreis zu verkaufen. Typische Auktionsgüter sind Sammlerstücke. Daneben eignen sich zum Vertrieb Auslaufmodelle, Überschuss- und B-Waren. Für diese Güter lassen sich nur noch 8-40 % ihres ursprünglichen Wertes erzielen. Für den Vertrieb dieser Güter sind Online-Auktionen eine kostengünstige und einfache Alternative zu klassischen Vertriebswegen [THOM00, S. 17].

Betreiber eines Auktionssystems profitieren von der Anzahl der Transaktionen, die über das System abgewickelt werden. Ihr Ziel ist es, durch effiziente Handelsmechanismen und attraktive Produktangebote positive Netzeffekte zu fördern. Für KMU bieten viele Anbieter von Online-Shop-Systemen Erweiterungen für Online-Auktionen an (z.B. OpenShop oder Cartridge-Hersteller für Intershop 4.0 [ISOP00]). Im elektronischen Vertrieb kann durch Online-Auktionen die Attraktivität eines Online-Angebots gesteigert werden.

2.5.4 Virtuelle Gemeinschaften

Virtuelle Gemeinschaften (engl. = Virtual Communities) bieten verschiedenen Interessengruppen eine virtuelle Plattform zum Informationsaustausch. Entscheidend für den Erfolg einer Virtuellen Gemeinschaft sind Netzeffekte. Der Wert einer Virtuellen Gemeinschaft wächst mit der Zahl ihrer Nutzer und der Informationen, die sie innerhalb der Gemeinschaft austauschen. Aufgabe des Betreibers ist es, eine geeignete Systemplattform zu schaffen, die der Zielgruppe einen Anreiz zur Preisgabe von Informationen gibt. Wichtige Merkmale einer Virtuellen Gemeinschaft sind u. a. ein klar erkennbarer Interessenschwerpunkt, Kommunikationsmöglichkeiten für die Mitglieder einer Gemeinschaft und Funktionen, die sie zur Erstellung eigener Inhalte einsetzen können [HAGE97, S. 23].

Als Einnahmequellen für die Betreiber Virtueller Gemeinschaften kommen direkte Einnahmen aus Nutzungs- und Teilnahmegebühren sowie indirekte aus Werbung und Geschäftsprovisionen in Frage. In Anbetracht des Angebots an kostenfreien Community-Diensten werden sich die erwünschten Netzeffekte zumindest zu Beginn durch kostenpflichtige Angebote kaum erreichen lassen. Der wesentliche Erfolgsfaktor ist die Gewinnung einer kritischen Masse an Mitgliedern, um direkte und indirekte Einnahmen erzielen zu können.

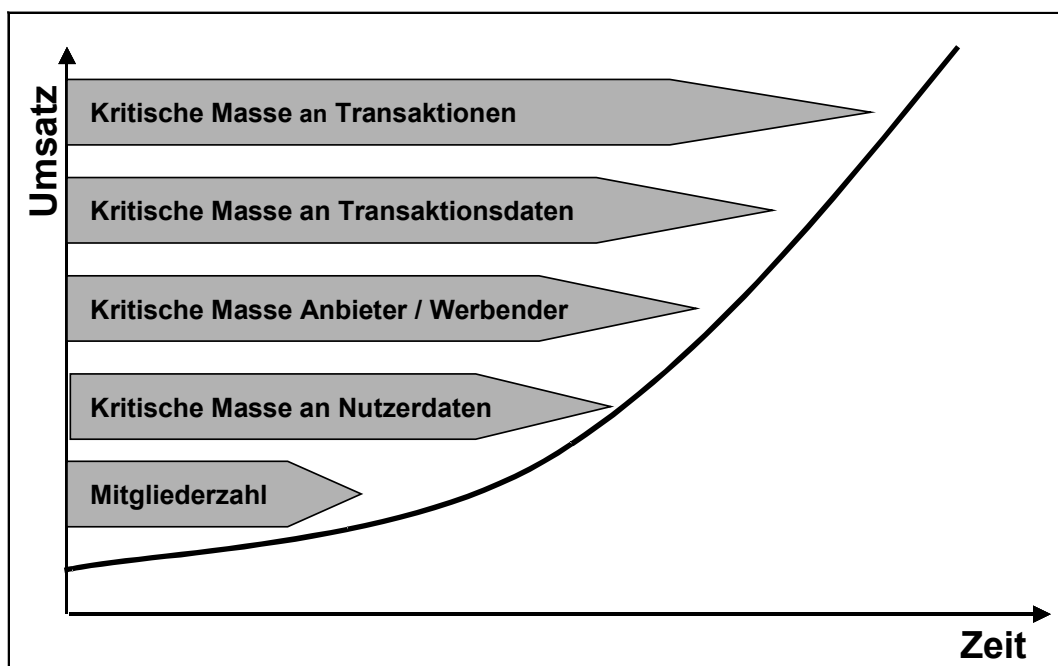


Abbildung 2-10: Erfolgsfaktoren für Virtuelle Gemeinschaften (in Anlehnung an [HAGE97, S. 75])

Neben den Einnahmen aus dem Betrieb bieten Virtuelle Gemeinschaften neue Möglichkeiten im Marketing und Vertrieb von Produkten und Dienstleistungen über das Internet. In Virtuellen Gemeinschaften entstehen Mehrwerte, die ein Anbieter von sich aus nicht schaffen kann. Mit steigender Nutzerzahl der Community steigt die Attraktivität des Angebots für die Kunden. Dies führt zu einer immer höheren Kundenbindung. Aus diesem Grund werden Communities häufig mit anderen Ausprägungsformen zu einem Angebot kombiniert [BORC99].

2.6 Umsetzung ökonomischer Potentiale

Die Darstellung der ökonomischen Effekte zeigt die Vielfalt der Möglichkeiten, die elektronische Märkte gerade KMU bieten. Negative Berichte der letzten Zeit belegen aber auch, dass sich die oft hohen Erwartungen vieler Unternehmen bisher nicht erfüllt haben. Die anfängliche Euphorie im Umfeld der „New Economy“ weicht zunehmend den Erfahrungen aus der Praxis. Projektbeispiele und Untersuchungen zeigen, dass die Gründe für den Misserfolg vielfältig sind.

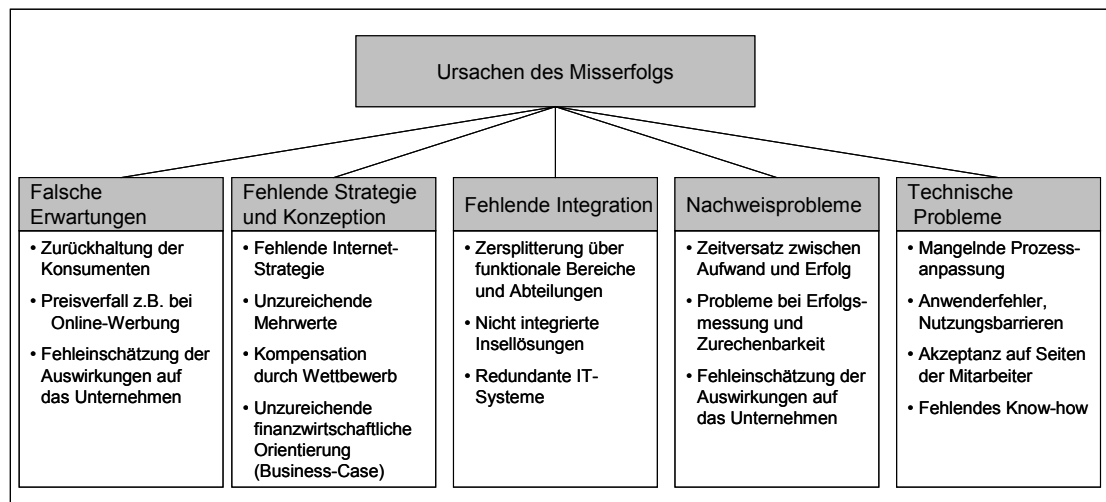


Abbildung 2-11: Ursachen für den Misserfolg in Projekten (in Anlehnung an [STRA02, S. 21])

Hauptursache für das Ausbleiben des Erfolgs ist empirischen Analysen zufolge das Fehlen einer geeigneten „Internet-Strategie“ und fundierter konzeptioneller Grundlagen. Die Folge sind häufig überzogene Erwartungen. Bei Unternehmen, die elektronische Medien bereits mit erkennbarem Erfolg nutzen, ist die Internet-Strategie dagegen integrierter Bestandteil der Unternehmensstrategie [STRA02, S. 28].

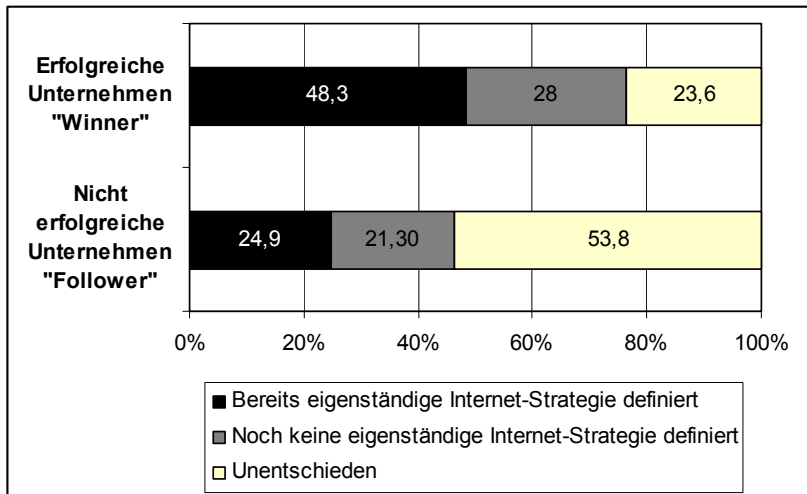


Abbildung 2-12: Verfügbarkeit einer Strategie
 (in Prozent von 465 Unternehmen) [STRA02, S. 28]

Zentraler Ansatzpunkt bei der Entwicklung einer Strategie für den Einsatz elektronischer Märkte ist die Differenzierung vom Wettbewerb [STRA02, S. 29]. Es genügt nicht, elektronische Märkte nur als Mittel zur Verbesserung etablierter Geschäftsprozesse einzusetzen, denn altbewährte Geschäftsmodelle sind kaum unmittelbar auf diese Märkte übertragbar.

Häufig entscheidende Faktoren auf klassischen Märkten, wie Standortvorteile oder etablierte Kundenbindungen, verlieren auf elektronischen Märkten ihre Bedeutung. Differenzierungsstrategien, wie Marktsegmentierung oder Preisdifferenzierung, sind aufgrund der höheren Informationstransparenz auf elektronischen Märkten deutlich schwerer durchsetzbar. Langfristig reicht eine Steigerung der Effektivität alleine nicht aus, um die Wettbewerbsfähigkeit eines Unternehmens zu sichern, denn Effizienzsteigerungen, die auf technologischen Innovationen und/oder Verbesserungen der Prozesse beruhen, sind auf lange Sicht von Konkurrenten nachvollziehbar [PORT97]. Für den Einsatz elektronischer Märkte dürfte dies in Anbetracht der Informationstransparenz und des hohen Standardisierungsgrades der dort eingesetzten Technologien in besonderem Maß gelten. Auch eine starke Orientierung an sog. Best Practice-Beispielen führt leicht lediglich in die Situation des „ewigen Verfolgers“ [STRA02, S. 67f.]. Um die Wettbewerbsfähigkeit eines Unternehmens nachhaltig zu sichern, muss deshalb eine neue Strategie entwickelt werden, die klare Wettbewerbsvorteile bietet [FRIS01, S. 15f.].

Diese Strategie muss klare Ziele definieren und Aussagen über die zu erwartenden Vorteile machen. Zur Umsetzung dieser Strategie im Unternehmen ist ein Konzept zu erarbeiten, das geeignete Lösungen aufzeigt. Dies umfasst eine Beschreibung des geplanten Angebots und des Systems, über das diese Leistungen angeboten werden sollen. Die Konzeption ist Grundlage für die Durchführung der Aufgaben in den nachfolgenden Projektphasen bis zur Einführung der Gesamtlösung.

Erfolgsentscheidend bei der Entwicklung der strategischen Konzeption ist die richtige Einschätzung der Potentiale des Internets für die elektronische Geschäftsentwicklung und der Auswirkungen der Nutzung neuer, elektronischer Medien für das eigene Unternehmen [STRA02, S. 28-30]. Darüber hinaus muss das Konzept eine Beschreibung der Maßnahmen zur technischen und organisatorischen Umsetzung der definierten Ziele enthalten.

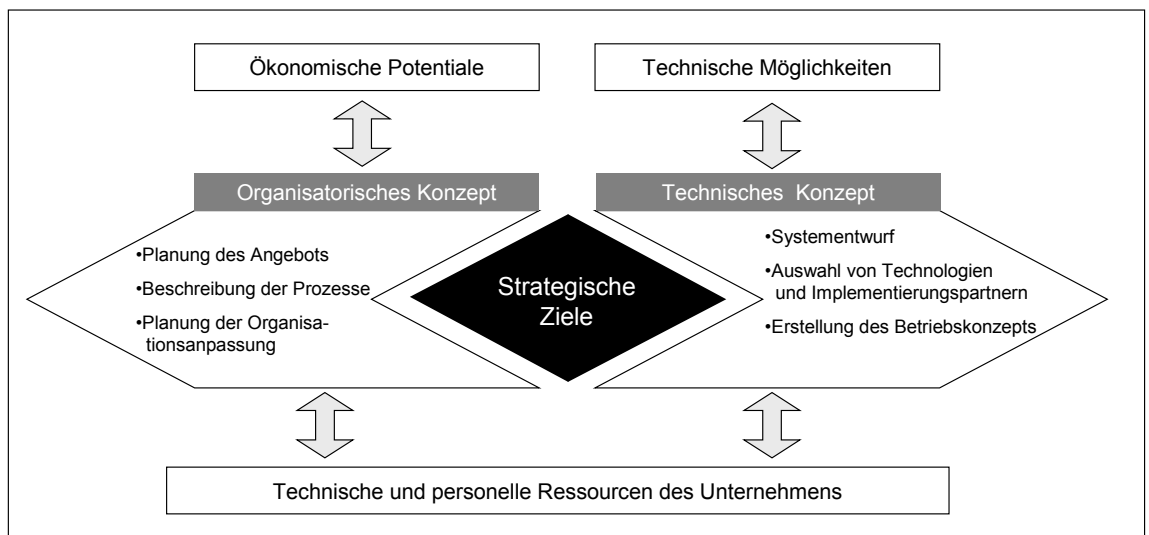


Abbildung 2-13: Inhalte eines strategischen Konzepts

Dabei ist zu berücksichtigen, dass die Nutzung eigener elektronischer Marktsysteme auch in kleinen und mittelständischen Unternehmen zur Veränderung der Aufgabefelder und Tätigkeiten einzelner Mitarbeiter führt [LOHS02, S. 79]. Die nachhaltig erfolgreiche Nutzung elektronischer Medien erfordert deshalb meist die Anpassung der vorhandenen Organisationsstruktur und bestehender Prozesse [STRA02, S. 32f.]. Schon um eine große Anzahl an Transaktionen über einen elektronischen Markt zeitnah im Unternehmen abwickeln zu können, sind geeignete Abläufe zu definieren.

Mangelhafte Abstimmung des Vertriebs über elektronische Medien mit bestehenden Vertriebskanälen kann leicht zu unerwünschten Kannibalisierungseffekten im eigenen Unternehmen führen (siehe 2.4.3).

Für KMU kann die Bindung umfangreicher personeller Kapazitäten durch Umorganisation und Anpassung technischer Systeme schnell zu sprunghaften Aufwandssteigerungen führen, die u. U. sogar das Kerngeschäft gefährden [WOLF96, S. 583]. Für Unternehmen dieser Größenordnung ist daher eine stufenweise Umsetzung neuer Strategien in kleinen, überschaubaren und kurzfristigen Anpassungsschritten zu empfehlen, deren Erfolg unmittelbar überprüft werden kann.

KMU verfügen zudem vielfach noch nicht über genügend eigene Erfahrungen, um die Potentiale elektronischer Märkte für das eigene Unternehmen richtig einzuschätzen. Ebenso fehlt oftmals das nötige Wissen für die softwaretechnische Realisierung internetbasierter Systeme. KMU sind deshalb meist auf externe Dienstleister angewiesen.

Selbst KMU, die das Internet bereits für die eigene Geschäftsabwicklung nutzen, befinden sich oft noch in einer eher experimentellen Phase. Die angebotenen Informationen sind vielfach unvollständig, die vorgesehenen Kontaktmöglichkeiten nicht selten nur unzureichend in die Prozesse des Unternehmens integriert. Grund für die abwartende Haltung vieler KMU ist die Angst vor unkalkulierbaren Investitionen und dem Risiko eines Fehlschlages [LOHS02, S. 11]. KMU müssen daher durch geeignete Methoden und Werkzeuge verstärkt unterstützt werden.

3 Methoden, Werkzeuge und Vorgehensmodelle für die Entwicklung betriebswirtschaftlicher Anwendungssysteme

Die Betrachtung der ökonomischen Effekte im vorhergehenden Kapitel zeigt, welche Bedeutung der Aufbau eines Marktsystems für die Ausrichtung eines Unternehmens hat und welchen Umfang Projekte zur Realisierung und erfolgreichen Nutzung eines solchen Systems haben können. Um Projekte dieser Art durchzuführen, ist eine zielorientierte, methodische und durch geeignete Werkzeuge unterstützte Vorgehensweise erforderlich. KMU, die den Aufbau eines elektronischen Marktsystems planen, stehen damit vor dem Problem, geeignete Methoden und Werkzeuge sowie Vorgehensmodelle für ihren zielgerechten Einsatz auszuwählen.

Ziel dieses Kapitels ist es, einen Überblick über die verbreiteten Werkzeuge und Vorgehensmodelle zu geben und sie aus Sicht der KMU zu bewerten. Dazu erfolgt im ersten Schritt eine Abgrenzung der zentralen Begriffe. Im zweiten Abschnitt werden etablierte Methoden und Denkansätze der Entwicklung betrieblicher Anwendungssysteme vorgestellt, die beim Aufbau elektronischer Märkte wichtig sind. Anschließend wird das Spektrum an Werkzeugen zur Anwendung dieser Methoden betrachtet und auf ihre Eignung für KMU hin bewertet. Abschließend wird vor dem Hintergrund der besonderen Rahmenbedingungen durch Einordnung und exemplarische Beschreibung einiger Vorgehensmodelle die Eignung der verfügbaren Vorschläge zur Projektdurchführung geprüft.

Die Bewertung der etablierten Werkzeuge und Vorgehensmodelle im Hinblick auf ihre Anwendbarkeit durch KMU ist Grundlage für die Einführung eines neuen Ansatzes zur Systementwicklung durch wissensbasierte Adaption in Kapitel 4 und der daran anschließenden neuen Konzepte und Lösungen für elektronische Märkte.

3.1 Begriffsabgrenzung

Bei elektronischen Marktsystemen handelt es sich um betriebliche Anwendungssysteme mit hoher technischer Komplexität und betriebswirtschaftlicher Bedeutung. Für die Entwicklung betrieblicher Anwendungssysteme werden Methoden, Werkzeuge und Vorgehens- bzw. Prozessmodelle benötigt.

Unter einer **Methode** wird eine im Allgemeinen begründete und zur Erreichung festgelegter Ziele planmäßig angewandte Vorgehensweise verstanden. In der Softwareentwicklung im Speziellen werden aber auch Konzepte, Notationen und methodische Vorgehensweisen als Methoden bezeichnet [BALZ00, S. 36-37]. Methodische Vorgehensweisen liefern konkrete Beschreibungen und/oder Regeln zur Anwendung allgemeiner Prinzipien der Softwareentwicklung (z. B. Hierarchisierung, Modularisierung). **Werkzeuge** werden zur Unterstützung bei der Anwendung von Methoden eingesetzt [BALZ00, S. 36-37]. Bei den hier betrachteten Werkzeugen handelt es sich ausschließlich um softwarebasierte Systeme. Der koordinierte Einsatz von Methoden und Werkzeugen muss in einem Projekt nach einem geeigneten Vorgehens- bzw. Prozessmodell koordiniert werden. **Vorgehensmodelle** beschreiben eine methodische Vorgehensweise, die Aufgaben, Produkte und ihre Beziehungen zueinander in Form eines Ablaufmodells definieren. Sie enthalten auch Hinweise auf Methoden und Werkzeuge zur Durchführung der Aufgaben [EICK99, S. 359] sowie eine Rollendefinition für die Verteilung der Projektaufgaben auf die beteiligten Mitarbeiter [BALZ00, S. 54]. Für jedes Projekt muss ein geeignetes Vorgehensmodell ausgewählt und angepasst werden. Das gewählte Modell legt aber lediglich die Rahmenbedingungen fest, in denen in einem konkreten Prozess- oder Projektleitfaden Rollen und Aufgaben definiert, Werkzeuge ausgewählt, eingesetzt und schließlich Dokumente und Lösungen erzeugt werden [MÜLL99, S. 4].

Ein Vorgehensmodell zur Systementwicklung muss im Idealfall folgende allgemeinen Eigenschaften aufweisen [CANT98, S. 82]:

- (1) Beschreibung der grundsätzlichen Vorgehensweise mit allen Aktivitäten zur Planung und Durchführung eines Projekts,
- (2) Definition von (Teil)zielen,

- (3) Beschreibung von Methoden zur Kontrolle des Projektfortschritts und der Projektkosten sowie zur Erkennung und Abschätzung von Projektrisiken.

Neben der Nutzung bewährter Methoden und verfügbarer Werkzeuge bietet es sich an, in Projekten auf vorgefertigte Softwarebausteine, vorhandene Softwaremodelle und –architekturen zurückzugreifen. Bei einem **Modell** handelt es sich in der Softwaretechnik um eine idealisierte, vereinfachte Darstellung des zu entwickelnden Systems mit dem Ziel, bestimmte Eigenschaften deutlich zu machen [BALZ00, S. 100]. Insbesondere in der objektorientierten Softwareentwicklung (siehe 3.2) spielt der Entwurf des sog. Objektmodells, das das gesamte Zielsystem in Objekten beschreibt, eine wichtige Rolle. Teil der Erstellung des Systemmodells ist der Entwurf der Architektur des Systems. Die **Architektur** eines Systems beschreibt die Struktur einer Software, ihre einzelnen Bestandteile (meist Softwarekomponenten) und die Beziehungen zwischen diesen [BALZ00, S. 696]. Eine wesentliche Vereinfachung bei der Entwicklung unternehmensspezifischer Modelle ist die Anpassung vorhandener Referenzmodelle. **Referenzmodelle** bieten Lösungsansätze für eine ganze Klasse betriebswirtschaftlicher oder auch technischer Problemstellungen [BECK01]. Betriebswirtschaftliche Referenzmodelle sind Prozessmodelle, welche die Systemanalyse, die Informationsanalyse und den Sollentwurf von Informationssystemen unterstützen [HUFG94, S. 141].

3.2 Paradigmen und Methoden der Softwareentwicklung

Softwarelösungen können im ersten Ansatz grob in Individual- und Standardsoftwarelösungen unterschieden werden. Mit dem Zukauf vielfach eingesetzter Softwarelösungen verbindet sich häufig auch die Erwartung, mit geringem Aufwand zu einer qualitativ hochwertigen Lösung zu kommen. Standardsoftware ist im Allgemeinen etwa um den Faktor 10 bis 20 billiger als eine vergleichbare Individualentwicklung [SCHE90, S. 141], ist sofort verfügbar und damit schneller einzuführen. Gerade dies macht Standardsoftware für KMU attraktiv. Untersuchungen prognostizieren, dass der Anteil der Standardsoftware an den Unternehmenslösungen stark steigen wird und Individuallösungen zukünftig vorwiegend

von größeren Firmen in Auftrag gegeben werden [BALZ00, S. 33]. Dennoch schrecken die zwangsläufigen Einschränkungen dieser Produkte in ihrer Anpassbarkeit viele vom Einsatz umfangreicher Standardsoftwarelösungen ab. Einen Kompromiss zwischen vollständiger Individualentwicklung und Standardsoftwareanpassung stellt die Verwendung vorgefertigter Softwarekomponenten dar. Gerade für die Entwicklung elektronischer Märkte, bei der in jedem Projekt grundlegende technische Probleme gelöst werden müssen und zunehmend standardisierte Verfahren auch für betriebswirtschaftliche Problemstellungen eingesetzt werden, ist die Verwendung vorgefertigter Softwarekomponenten mittlerweile unverzichtbar geworden. Softwarelösungen lassen sich demnach in die in Abbildung 3-1 angeführten Arten differenzieren.

Art der Softwarelösung		Auftraggeber-Firmen (Personalgröße)
Individual-/ Spezialsoftware	5%	50.000 - 100.000
Standardsoftware & individuelle Anpassungen	10%	3.000 - 50.000
Lösungen zusammengesetzt aus Softwarekomponenten	35%	500 - 3.000
Vollständige Lösungen durch Standardsoftware	50%	< 500

Abbildung 3-1: Trends bei Softwarelösungen [BALZ00, S. 33]

Nach den Arten von Softwarelösungen lassen sich grundlegend verschiedene Denkansätze unterscheiden, die bei ihrer Entwicklung verfolgt werden. Dementsprechend werden verschiedene Methoden eingesetzt. Eine Differenzierung der verwendeten Methoden nach dem zugrunde gelegten Paradigma der Systementwicklung ist zwingend erst für die eigentliche Implementierung eines Systems erforderlich. Dennoch haben sich Methoden durchgesetzt, die alle Phasen der Systementwicklung insoweit durchgängig unterstützen, dass sie auch in den

Anfangsphasen der Softwareentwicklung, der Analyse und dem Systementwurf eingesetzt werden. Eine lange in der Praxis verwendete Methodensammlung, die diese Phasen unterstützt, ist die sog. Strukturierte Analyse. Sie wurde inzwischen von der UML weitgehend abgelöst [BALZ00, S. 432]. Ursprünglich hervorgegangen aus einer Zusammenfassung von Methoden des objektorientierten Systemdesigns, bietet die UML in der aktuellen Version [OMG01] eine Reihe von Diagrammtypen zur Analyse von Geschäftsprozessen und der Spezifikation von Systemanforderungen.

3.2.1 Individualsoftwareentwicklung

Bei Individualsoftwareentwicklung wird unter Verwendung von Elementarmethoden der Programmierung ein System exakt von Grund auf nach den Anforderungen eines Unternehmens erstellt. Die dabei verwendeten Methoden lassen sich nach den zwei wesentlichen Paradigmen innerhalb der Individualsoftwareentwicklung in Methoden der strukturierten Programmierung und der objektorientierten Programmierung unterscheiden.

In der strukturierten Programmierung werden Algorithmen durch Verwendung der Kontrollstrukturen Sequenz, Auswahl, Wiederholung und Aufruf anderer Algorithmen abgebildet [BALZ00, S. 260]. Notationen zum Entwurf strukturierter Programme sind Programmablaufpläne oder Nassi-Schneidermann-Diagramme. Letztergenannte haben den Vorteil, dass sie die Verwendung von Sprüngen verbieten und so eine modulare Strukturierung des Programmcodes erzwingen.

In der objektorientierten Programmierung OOP wird ein System in Form sog. Objekte abgebildet. Grundprinzipien der OOP sind die Kapselung von Objekten, Abstraktion und Vererbung. Durch Abstraktion bei der Objektbildung entstehen Objektklassen, die eine Klassenhierarchie bilden. Innerhalb der Klassenhierarchie werden Eigenschaften von der jeweils abstrakteren an die konkretere Klasse vererbt. Durch Abstraktion lassen sich auch komplexe Systeme sehr effizient beschreiben. Die konsequente Kapselung der Objekte gegen Eingriffe von Außen unterstützt die Wiederverwendung auch komplexer Objekte und Klassenstrukturen, wie Business Objects, Application Frameworks und vorhandene Klassenbibliotheken (siehe 3.3.1). Zentrales Problem bei der OOP ist das Finden geeigneter Objekte und der Entwurf der Klas-

senhierarchie. Hier können Klassen- und Zustandsdiagramme der UML sinnvoll eingesetzt werden.

Neben Methoden zur Programmentwicklung kommen bei der Individualentwicklung auch Methoden zur Modellierung von Datenstrukturen vorwiegend für relationale Datenbanksysteme zum Einsatz. Hierfür geeignet sind Methoden der Entity-Relationship-Modellierung [HANS92, S. 567-570] (vergleiche ER-Modell in Abbildung 5-5).

3.2.2 Komponentenbasierte Softwareentwicklung

Bei **komponentenbasierter Softwareentwicklung** wird versucht, Softwaresysteme aus unabhängigen, miteinander verknüpfbaren Teilen zusammenzusetzen [STRI97, S. 118]. Die Analyse des Angebots an Werkzeugen zum Aufbau elektronischer Märkte in 3.3.1 zeigt, dass die Verwendung von Softwarekomponenten bereits durch eine Reihe verfügbarer Werkzeuge konsequent unterstützt wird und durch die Nutzung vorhandener Softwareelemente bei der Entwicklung von Electronic Commerce-Systemen erhebliche Produktivitätsgewinne erzielt werden können.

Gegenüber der reinen OOP bietet komponentenbasierte Softwareentwicklung den Vorteil, dass sie die Verwendung beliebiger Softwareelemente in unterschiedlichen Kontexten möglich macht. Zentrale Problemstellung bei komponentenbasierter Softwareentwicklung ist die Auswahl von Softwarekomponenten mit den erforderlichen Eigenschaften und ihre Integration in eine funktionsfähige Softwarearchitektur. Dazu sind Schnittstellen zu entwickeln, über die die Kommunikation zwischen den einzelnen Komponenten erfolgt (siehe 3.2.2.2). Eine geeignete Form der Visualisierung komplexer komponentenbasierter Softwarearchitekturen sind Komponentendiagramme der UML (siehe (Abbildung 5-1).

3.2.2.1 Wichtige Eigenschaften von Softwarekomponenten

Der Begriff **Komponente** wird in der Softwaretechnologie sehr vielseitig verwendet und noch unscharf abgegrenzt. Im Rational Unified Process (siehe 3.4.2) ist komponentenbasierte Softwareentwicklung eines der zentralen Konzepte. Dort werden alle

in sich abgeschlossenen, selbstständig ablauffähigen Bestandteile einer Software als Komponenten bezeichnet.

In [FRAN99, S. 386] wird eine Komponente definiert als „Coherent package of software artifacts that can be independently developed and delivered as a unit and that can be composed, unchanged, with other components to build something larger“.

Einige Autoren sehen in der Vermarktungsfähigkeit und der dafür notwendigen Einsetzbarkeit einer Komponente in einer beliebigen Systemumgebung ein entscheidendes Kriterium dafür, dass es sich bei einem Softwareelement um eine Komponente handelt [SZYP99, S. 14-20]. Die Integration beliebiger Komponenten ist aber ohne Anwendung spezieller Adaptionsstrategien kaum möglich [BERG00]. Hilfreich sind dabei klare Schnittstellendefinitionen zur Integration neuer Komponenten (siehe 5.4). Vielfache Verwendung fanden Komponenten bisher im Rahmen spezieller Montageumgebungen, die Mechanismen für das Zusammenfügen (Plugging) und die Definition von Interaktionen zwischen den Komponenten (Scripting) definieren. Beispiele für solche Montageumgebungen sind visuelle Programmierumgebungen oder Komponentenbibliotheken zur Erstellung grafischer Benutzungsoberflächen mit Hilfe grafischer Editoren. Durch Standards wie CORBA und DCOM, die offene und systemübergreifende Architekturen für komponentenbasierte Softwareentwicklung beschreiben, werden darüber hinaus Anwendungen realisierbar, die nicht auf eine einzige Entwicklungsumgebung beschränkt sind.

Die allgemeinen Definitionen und unterschiedlichen Ansätze geben wenig Aufschluss darüber, wie eine konkrete Softwarelösung auf Basis einer komponentenbasierten Softwarearchitektur zu realisieren ist. Sie lassen jedoch wesentliche Eigenschaften von Softwarekomponenten erkennen, die für die Realisierung einer Standardanwendungssoftware zum Aufbau elektronischer Marktsysteme wichtig sind.

Einsatz unabhängig von der Implementierung

Die Forderung, dass eine Komponente unabhängig von anderen Teilen einer Software verwendbar sein soll, setzt voraus, dass jede Komponente ohne Kenntnis ihrer konkreten Implementierung eingesetzt werden kann. Diese Anforderung ist mit dem Paradigma der objektorientierten Systementwicklung vereinbar, das vollständige Kapselung der einzelnen Objekte verlangt. Dort dient das Konzept der Kapselung

vorrangig als Mittel zur Handhabung von Systemkomplexität. Mit der Forderung nach Kapselung ist ein hoher Anspruch an die Disziplin bei Systemdesign und Implementierung der einzelnen Objekte verbunden, der von vielen softwaretechnischen Lösungen nicht erfüllt wird und bei fertigen Softwarebausteinen im Nachhinein auch kaum überprüfbar ist. Vollständige Kapselung jeder Komponente gegen äußere Eingriffe wird deshalb in der komponentenbasierten Softwareentwicklung als wünschenswert, aber nicht als zwingend erforderlich angesehen. Das bedeutet auf der anderen Seite, dass die Stabilität des Systems durch sorgfältiges Design sowie die Implementierung zuverlässiger Schnittstellen und Kollaborationsformen sichergestellt werden muss.

Durch die Unabhängigkeit von Softwarekomponenten von ihrer Implementierung wird gleichzeitig die Austauschbarkeit der Module und die Erweiterbarkeit jeder komponentenbasierten Softwarelösung gewährleistet.

Adaptierbarkeit von Komponentenexemplaren

Die Forderung, dass eine Komponente eine abgeschlossene und unabhängige Einheit bilden muss, bedeutet nicht, dass eine Komponente nicht angepasst werden kann. Erst durch die Adaptierbarkeit wird die Entwicklung standardisierter, mehrfach verwendbarer und damit auch vermarktbarer Softwarekomponenten möglich. Komponenten vereinigen, ebenso wie objektorientierte Klassen, Daten und Methoden zu einer selbstständigen Einheit. Sie können wie diese als abstrakte Datentypen angesehen werden, von denen zur Laufzeit die benötigte Menge an Exemplaren erzeugt wird. Mehrere Exemplare können sich im Rahmen der Implementierung von Variablen in ihren aktuellen Ausprägungen deutlich voneinander unterscheiden. So ist es möglich, Ausprägungen derselben Komponente in unterschiedlichen Kontexten des Systems einzusetzen. Darüber hinaus verfügen viele Komponenten über konfigurierbare Schnittstellen. Sie werden auch als Adapter bezeichnet [STRI97, S. 118f].

Die Anpassung von Komponenten wird vielfach von Entwicklungsumgebungen durch Eigenschafteneditoren unterstützt [BALZ00, S. 860]. Noch weiter gehen Montageeditoren. Diese bieten Funktionen, über die eine Vielzahl von Komponenten interaktiv ausgewählt, angepasst und zu komplexeren Einheiten oder ganzen Anwendungen zusammengefügt werden können. Ein anschauliches Beispiel ist das Produkt Visual Age for Java der Firma IBM [IBM97].

Vermeidung von Vererbungsbeziehungen zwischen Komponenten

Nach der Idee komponentenbasierter Softwareentwicklung soll die Verwendung und Integration beliebiger Softwareelemente möglich sein. Vererbung in einem objektorientierten System bedeutet aber zwangsläufig eine Abhängigkeit zu einer übergeordneten Klasse. Durch Vererbung über die Implementierungsgrenzen einer Komponente hinaus wird folglich eine Abhängigkeit zum umgebenden System geschaffen, auf die möglichst verzichtet werden sollte [FRAN99, S. 390]. Erst durch den Verzicht auf die Einbindung der Komponenten in komplexe Vererbungshierarchien wird ihre Verwendung unabhängig und ohne Kenntnis der Implementierung des Gesamtsystems möglich. Hier liegt ein wesentlicher Unterschied zu anderen Konzepten der Softwarewiederverwendung wie Application Frameworks und Business Objects. Wenn die Komponenten einer Softwarelösung nicht programmtechnisch in eine starre Vererbungshierarchie eingebunden sind, kann die dynamische Austauschbarkeit der Module bzw. Komponenten realisiert werden.

Delegation der Speicherung von Daten

Die Unbeschränktheit der Implementierungsmöglichkeiten bedeutet aber auch, dass eine Funktion zur persistenten Speicherung der Instanzvariablen von Komponenten nicht vorausgesetzt werden kann [ROSE01, S. 2]. Komponenten größeren Umfangs besitzen dennoch häufig eigene Funktionen zur dauerhaften Datenspeicherung. Einzelne Objekte eines objektorientierten Systems werden dagegen zur Laufzeit temporär erzeugt. Die Aufgabe der persistenten Speicherung von Daten übernimmt in rein objektorientierten Anwendungen meist ein über spezielle Schnittstellen integriertes System. Analog muss in komponentenbasierten Systemen die Funktion der persistenten Speicherung relevanter Daten an ein System delegiert werden, das die Inhalte der Instanzvariablen aller Komponenten eigenständig verwaltet. Darüber hinaus müssen in einer komponentenbasierten Architektur der reibungslose Austausch und die Abwicklung von Transaktionen zwischen den Komponenten unterstützt werden. Vielfache genutzte Technologien, wie Enterprise JavaBeans, CORBA und DCOM, bieten hierfür jeweils eigene Implementierungsmöglichkeiten [SOUZ99, S. 392].

Durch die Delegation der Speicherung aller Instanzvariablen an ein Datenbanksystem wird die Forderung nach einem koordinierten Datenzugriff aller Elemente einer adaptionsfähigen Standardsoftware auf eine gemeinsame Datenbank erfüllt und die

dynamische Adaptionfähigkeit der Parametereinstellungen sowie die konsistente Verwaltung von Daten wesentlich unterstützt.

3.2.2.2 Formen der Schnittstellenimplementierung

Das zentrale Problem komponentenbasierter Software ist die Implementierung von Schnittstellen, über die die Kommunikation zwischen den Komponenten erfolgt. Die Formen der Schnittstellenimplementierung lassen sich in direkte und indirekte Schnittstellen unterscheiden. Über **direkte Schnittstellen** greift eine Komponente unmittelbar auf die Schnittstelle einer anderen Komponente zu. Das entspricht einem Prozeduraufruf bei prozeduraler Programmierung oder der Referenz eines Objektes auf ein anderes Objekt über eine Instanzvariable. Fordert eine Komponente (Client) von einer anderen Komponente (Provider) einen Dienst über eine direkte Schnittstelle an, die für diese Komponente spezifisch ist, muss diese die Schnittstellenbeschreibung des Providers genau kennen. Komponentenarchitekturen mit direkten Schnittstellen sind daher nur mit erheblichem Aufwand erweiterbar und die einzelnen Komponentenschnittstellen nur schwer adaptierbar. Die Form der direkten Schnittstellenimplementierung muss daher für die Kommunikation zwischen Komponenten einer adaptionsfähigen Standardanwendungssoftware als zu wenig flexibel gewertet werden.

Erfolgt die Kommunikation über **indirekte Schnittstellen**, wird eine Instanz zwischengeschaltet, die die Vermittlung zwischen Provider- und Client-Komponente übernimmt. Diese vermittelnde Instanz wird häufig als **Request Broker** bezeichnet. Alle Komponenten, die über den Request Broker miteinander kommunizieren sollen, melden ihre Dienste und Schnittstellen bei dem Broker an. Provider und Client werden völlig entkoppelt und kommunizieren nur über die Schnittstellenbeschreibung, die der Broker bereitstellt. Über einen Broker können Kommunikationsstandards etabliert und eine hohe Interoperabilität verschiedenster Komponenten hergestellt werden. Als Standard für komponentenbasierte Architekturen wurde von der OMG die Common Object Request Broker Architecture, kurz CORBA, definiert und ständig weiterentwickelt [SCAL01]. Die Beschreibung von Schnittstellen erfolgt in CORBA durch die Interface Definition Language (IDL). Neben der Abstimmung von Schnittstellen kann der Broker die zeitliche Koordination von Prozessen innerhalb einer Anwendung übernehmen.

3.2.3 Anpassung von betriebswirtschaftlicher Standardsoftware

Ein weiterer vielversprechender Weg zur Entwicklung betrieblicher Anwendungssysteme ist die Anpassung einer umfassenden betriebswirtschaftlichen Standardsoftwarelösung. Allgemein wird hierfür der Begriff des Customizing verwendet. HUFgard unterscheidet Möglichkeiten der Standardsoftwareanpassung aus der Praxis genauer in klassische Customizing-, klassische Adaptionen- und Software-Engineering-orientierte Adaptionenansätze [HUF94, S. 129-140]. Die ersten Customizing-Ansätze beruhen auf der Erzeugung und Modifikation von Programmcode aus vorhandenen Modulen. Diese Methode ist langwierig und wenig flexibel. Der Wechsel auf eine neue Release der Softwarelösung ist nach umfangreichen Code-modifikationen nicht mehr möglich. Wesentliches Prinzip, um die Releasewechselfähigkeit einer Standardsoftwarelösung zu erhalten, ist die Nicht-Modifikation der Module. U. a. aus diesem Grund werden in dem im nächsten Kapitel vorgestellten neuen Ansatz die Methoden zur Adaption einer Standardanwendungssoftware auf die Anpassung im vordefinierten Rahmen beschränkt. Die Releasewechselfähigkeit der Standardsoftware kann zum Teil durch den Einsatz von Software-Engineering-Methoden und -Werkzeugen erreicht werden, die in die Standardsoftware integriert werden können und die Anpassung auf zugelassene Ergänzungsentwicklungen beschränken. Ein Beispiel für diese Art der Softwareanpassung bei Entwicklungswerkzeugen für elektronische Märkte ist die Bearbeitung von Templates mit Hilfe spezieller Editierwerkzeuge in templatebasierten Standardsoftwarelösungen (siehe 3.3.2.3). Klassische Adaptionenansätze gehen in diesem Punkt weiter. Sie beschränken die Anpassungsmöglichkeiten auf das Einstellen von Parametern, die in der Standardsoftware vorgesehen sind. Methoden klassischer Adaptionenansätze sind die Auswahl von Modulen bzw. Funktionen mit Hilfe von Checklisten sowie die Parameter- und Tabellensteuerung zur Modulanpassung. Durch Parametrisierung und getrennte Speicherung der Adaptionenparameter wird eine dynamische Adaption des Systems möglich [HUF94, S. 135]. Problematisch ist das richtige Einstellen und Verwalten der Vielzahl an Parametern einer vollständig parametrisierbaren Standardsoftware. Zur Adaption dieser Systeme werden sehr flexible und leistungsfähige Werkzeuge benötigt. Dies mag ein Grund sein, warum die netzbasierten Assistenten, die auf diesem Adaptionenansatz aufbauen, nur sehr eingeschränkt anpassbar sind.

An diesem Problem setzt das wissensbasierte Konzept zur Adaption an, das im nächsten Kapitel vorgestellt und danach zu einer integrierten Lösung für elektronische Marktsysteme weiterentwickelt wird.

3.3 Werkzeuge

Ziel dieses Abschnitts ist es, einen Überblick über das Angebot an verfügbaren Werkzeugen zur Entwicklung elektronischer Märkte zu vermitteln, der eine Bewertung einzelner Produkte aus der Sicht kleiner und mittlerer Unternehmen ermöglicht. Da die Darstellung des Funktionsumfangs einzelner Werkzeuge in diesem sehr dynamischen Markt nur einer unvollständigen Momentaufnahme der aktuellen Situation mit geringem praktischem Wert entsprechen würde, liegt der Fokus auf der Betrachtung der grundlegenden technischen und betriebswirtschaftlichen Konzepte hinter den verschiedenen am Markt verfügbaren Werkzeugen. Diese sind deutlich weniger dynamischen Entwicklungen unterworfen. Sie können deshalb langfristig zur Bewertung des Marktangebots und seiner Entwicklung wie auch der Leistungsfähigkeit einzelner Produkte herangezogen werden.

3.3.1 Werkzeugklassifikation

Das Angebot an Werkzeugen zur Entwicklung internetbasierter Systeme ist mittlerweile sehr unübersichtlich. Die wachsenden Anforderungen des Marktes und die parallele Entwicklung der verfügbaren Technologien für den Aufbau von Web-Anwendungen führten zur Entwicklung einer breiten Palette an Werkzeugen. Das Angebot reicht von einfachen HML-Editoren für die schnelle Erstellung einfacher statischer Web-Seiten über integrierte Entwicklungsumgebungen zur Realisierung datenbankbasierter, dynamischer Web-Angebote bis hin zu komplexen Standardsoftwarelösungen, die die Entwicklung transaktions- und geschäftsprozessorientierter Lösungen unterstützen. Anbieter sind zum einen junge Unternehmen, die erst aufgrund der Nachfrage nach diesen Werkzeugen entstanden. Bekannte Beispiele hierfür sind die Firmen Intershop und Broadvision. Andererseits zwingt die wachsende wirtschaftliche Bedeutung und Verbreitung elektronischer Märkte im Internet auch etab-

lierte Softwarehersteller, ihre Produktpalette entweder durch Eigenentwicklungen oder Zukauf von Fremdprodukten entsprechend zu erweitern.

Hersteller großer ERP-Systeme, wie SAP und ORACLE, haben ihre Produkte um Funktionen zur Entwicklung von Online-Angeboten und entsprechende Schnittstellen ergänzt. Bei diesen Produkten ist die Integration mit elektronischen Märkten relativ einfach. Allerdings sind so umfangreiche und teure Systeme in der Zielgruppe der KMU kaum verbreitet [LINK99]. Sie werden deshalb hier nicht näher betrachtet. Neben der Entstehung neuer Werkzeuge einzelner Anbieter ist aktuell die Bildung von strategischen Allianzen zwischen Werkzeuganbietern zu beobachten. So hat die Firma Intershop für die Produkte Intershop 4 und Enfinity ein eigenes „Cartridge-Konzept“ entwickelt, das die Integration von Fremdprodukten erleichtert [INTE00, S. 60; INTE00b, INTE01]. Damit kann der Funktionsumfang sowohl horizontal durch Integration von Lösungen für bisher nicht realisierte Funktionen und Ausprägungsformen elektronischer Märkte als auch vertikal über Schnittstellen zu Anbietern spezieller Verfahren erweitert werden.

Eine einfache erste Einteilung des Werkzeugangebots wäre die Differenzierung in Individualentwicklungsumgebungen und Standardsoftwarelösungen. Durch den Einsatz vorgefertigter Softwarebausteine verschwimmt die Grenze zwischen Individual- und Standardsoftwarelösungen (siehe 3.2). SZYPERSKI sieht auch mehrfach verwendbare Softwarekomponenten als ein Stück Standardsoftware an [SZYP99, S. 5]. Für eine genauere Bewertung des Werkzeugangebots muss der Begriff der Standardsoftware differenzierter betrachtet werden. THOME bezeichnet mit dem Begriff **Standardsoftware** „Programmpakete, die so entwickelt wurden, dass sie einer größeren Zahl von Anwendern genügen“ [THOM96, S. 170]. Dabei kann es sich entweder um System- oder Anwendungssoftware handeln. Bei der Entwicklung elektronischer Märkte steht die Realisierung einer anwendungsorientierten Lösung im Vordergrund. Innerhalb der Standardanwendungssoftware wird zwischen funktionsorientierter und betriebswirtschaftlicher Software unterschieden, da die Standardisierung einer Softwarelösung sowohl funktionsorientiert durch Bereitstellung vorgefertigter Elemente für einzelne Programmfunktionen als auch betriebswirtschaftlich durch Standardisierung ganzer Verfahren erfolgen kann [THOM96, S. 34].

VOGELSANG weist aber darauf hin, dass die Differenzierung zwischen funktionsorientierter und betriebswirtschaftlicher Standardsoftware aus Anwendersicht zuneh-

mend schwierig wird. Dies ist auf den wachsenden Funktionsumfang ehemals typischer funktionsorientierter Anwendungen zurückzuführen, die heute so mächtig geworden sind, dass sich auch kleine betriebswirtschaftliche Problemstellungen mit ihnen lösen lassen [VOGE98, S. 12].

Ein geeignetes Differenzierungsmerkmal zwischen funktionsorientierter und betriebswirtschaftlicher Standardanwendungssoftware ist die Ausrichtung der Referenzmodelle, nach denen die Elemente einer Standardsoftware abstrahiert und strukturiert sind. Kennzeichen **betriebswirtschaftlicher Standardanwendungssoftware** ist die Existenz eines oder mehrerer alternativer betriebswirtschaftlicher Referenzmodelle.

Eine **funktionsorientierte Standardsoftware** bietet Modellierungsansätze zur mehrfachen Verwendung funktionaler bzw. technischer Elemente, zwischen denen kein unmittelbarer betriebswirtschaftlicher Zusammenhang besteht. Ansätze zur Abbildung technischer Referenzmodelle finden sich im Bereich der objektorientierten Systementwicklung, die schon seit ihrer Entstehung auf eine möglichst vielfache Verwendung vordefinierter Elemente zielt. Bei parametrisierbaren Softwarekomponenten handelt es sich nach EICKER et al. um Referenzmodelle, die zur Implementierung bestimmter Funktionalitäten vielfach einsetzbar sind. Weitere Möglichkeiten sind Patterns und Frameworks, die aber ausschließlich die Modellierung einzelner Klassen bzw. Anwendungsbereiche unterstützen [EICK99].

Individualentwicklungsumgebungen weisen im Gegensatz zu Standardsoftware keine Referenzmodelle auf. Danach lässt sich das Angebot an Entwicklungswerkzeugen in die drei Grundtypen Individualentwicklungsumgebungen, funktionsorientierte Standardanwendungssoftware und betriebswirtschaftliche Standardanwendungssoftware differenzieren.

3.3.2 Bewertung der Werkzeugklassen

KMU profitieren vor allem von einer umfassenden Werkzeugunterstützung bei der Systementwicklung, Wartung und Anpassung eines elektronischen Marktes. Gleichzeitig sollen die Möglichkeiten zur Umsetzung individueller Wünsche und Anforderungen so wenig wie möglich eingeschränkt werden. Für die Auswahl eines Produktes zum Aufbau qualitativ hochwertiger elektronischer Marktsysteme sind aus der Sicht von KMU folgende Kriterien wichtig:

Tabelle 3-1: Kriterien zur Bewertung der Produktklassen

Abdeckungsgrad	Der Abdeckungsgrad erfasst, inwieweit technische und betriebswirtschaftliche Anforderungen, die im Rahmen der Anforderungsanalyse formuliert wurden, durch vorhandene, standardisierte Funktionen abgedeckt werden können. Hier ist zu untersuchen, welche Funktionen unmittelbar und mit eventuell geringer Anpassung abgedeckt werden können.
Anpassungsunterstützung	Dieses Kriterium beschreibt, in welchem Maß und auf welche Weise die Entwicklung und langfristige Anpassung eines Systems durch Werkzeuge unterstützt werden. Eine umfassende Anpassungsunterstützung ist Voraussetzung, damit Änderungen mit vertretbarem Aufwand im System vorgenommen werden können. Rein technische Adaptionswerkzeuge unterstützen lediglich die technische Korrektheit entstehender Lösungen. Um betriebswirtschaftlich erfolgreiche Lösungen aufzubauen, benötigen KMU meist zusätzlich eine konzeptionelle Unterstützung bei der Systemanalyse und dem Systementwurf.
Flexibilität	Die Flexibilität erfasst den Freiheitsgrad, den ein Werkzeug bezüglich der Anpassungen an die individuellen Anforderungen eines Unternehmens besitzt. Dieser hängt stark von der Modularität und Offenheit der Systemarchitektur des Werkzeugs ab.
Unterstützung der Systemverwaltung	Im laufenden Betrieb eines elektronischen Marktes sind ständig Inhalte zu pflegen, Transaktionsdaten zu bearbeiten und Wartungsarbeiten auszuführen. Ein wichtiges Bewertungskriterium ist daher, inwieweit diese Aufgaben durch vorhandene Funktionen des Werkzeugs unterstützt werden.
Integrationsunterstützung	Für ein Werkzeug zur Entwicklung elektronischer Märkte ist Integrationsfähigkeit ein entscheidendes Qualitätsmerkmal. Die Integrationsunterstützung bewertet, inwieweit die Integration des Zielsystems in bestehende Strukturen der Informationsverarbeitung unterstützt wird. Sie wird durch die Bereitstellung standardisierter Schnittstellen und die Nutzung technologischer Standards gefördert.

Nachfolgend werden die Produktklassen im Detail auf diese Kriterien hin untersucht.

3.3.2.1 Individualentwicklungsumgebungen

Viele der Individualentwicklungsumgebungen für Internet-Systeme sind aus den klassischen Entwicklungsumgebungen entstanden, die zunächst aus Mangel an dedizierten Werkzeugen eingesetzt wurden. Bekannte Beispiele sind Pearl oder Visual Basic. Diese Sprachen enthielten ursprünglich keinerlei spezielle Unterstützung zur Entwicklung von Kommunikationsanwendungen und wurden später um die dafür notwendigen Elemente (Schnittstellen zum WWW-Server, HTTP-Verarbeitung) ergänzt. Andere Technologien wurden für die Web-Entwicklung geschaffen, wie z. B.

Java und PHP oder die ASP-Technologie von Microsoft, die als kostenloser Bestandteil des Betriebssystems MS-Windows schnell eine dominierende Marktverbreitung erlangt hat. Insbesondere Java hat wegen seiner konsequenten Unterstützung der objektorientierten Systementwicklung eine große Bedeutung. Auch PHP wird fortwährend um neue Konzepte erweitert und verbreitet sich zusehends.

Individualentwicklungsumgebungen zielen primär auf eine möglichst große Flexibilität zur Entwicklung spezieller Lösungen. Standardisierte Elemente stehen nur für häufig wiederkehrende technische Problemstellungen, wie beispielsweise die Implementierung von Datenbankschnittstellen oder den Entwurf der Benutzerschnittstelle, zur Verfügung. Deshalb ist mit Werkzeugen dieses Typs zwar prinzipiell jede beliebige Anwendung realisierbar und auch der Entwicklung von Schnittstellen zur Integration mit anderen Systemen sind keine Grenzen gesetzt. Der Aufwand bei der Implementierung einer Lösung ist aber meist recht hoch. Alle Funktionen, die im Back-Office-Bereich zur Systemverwaltung benötigt werden, müssen vollständig ausprogrammiert werden. Jede Modifikation des Systems bedeutet eine Veränderung des vorhandenen Programmcodes.

Bezüglich der Auswahl eines Vorgehensmodells für die Projektdurchführung (siehe 3.4) gibt es bei Verwendung von Individualentwicklungsumgebungen wenig Anhaltspunkte. Mit zunehmender Ausrichtung am Paradigma der objektorientierten Programmierung sind modernere Werkzeuge dieser Klasse auch im Rahmen inkrementell-iterativer Vorgehensweisen einsetzbar. Entsprechende Methoden der Systemanalyse, des Systemdesigns und der Dokumentation werden allerdings von diesen Produkten nicht direkt unterstützt, sondern müssen durch Einsatz separater Werkzeuge erstellt und gepflegt werden. Viele UML-Diagrammwerkzeuge besitzen allerdings bereits Funktionen zur Erzeugung von Klassentemplates aus Klassendiagrammen.

Die aufwändige Entwicklung und Wartung macht individuell entwickelte Lösungen relativ kostspielig, fehleranfällig und inflexibel gegenüber späteren Anpassungen. KMU bieten sie kaum die erforderliche methodische Unterstützung. Diese verfügen zudem in den meisten Fällen nicht über ausreichendes programmiertechnisches Wissen, um Individualentwicklungsumgebungen effizient einsetzen zu können.

3.3.2.2 Funktionsorientierte Standardsoftware

Funktionsorientierte Standardsoftware ist ähnlich flexibel wie Individualentwicklungsumgebungen, stellt aber dem Entwickler zahlreiche vordefinierte Elemente bereit. Möglich wird dies durch Komponenten- und Application Frameworks und als Basisarchitektur. Für funktionsorientierte Standardsoftwarelösungen hat sich am Softwaremarkt die Bezeichnung **Application Server** durchgesetzt. Viele Application Server bauen auf etablierten Standards für Komponentensoftware des Internet, wie etwa ActiveX und Java Beans, auf. Hinter einer Komponente kann sich vom Transaktionsmonitor über komplexe Middleware bis hin zum elektronischen Bezahlverfahren fast alles verbergen. Application Frameworks bieten geeignete Abstraktionsmechanismen, um gängige Verfahren effizient in standardisierter Form abzubilden und Komponenten einzubinden. Damit sind die softwaretechnischen Voraussetzungen zur Bereitstellung eines umfassenden Funktionsvorrats, beliebiger Schnittstellen und eines betriebswirtschaftlichen Profils grundsätzlich gegeben. Marktführende Application Server stellen neben technischen Funktionen zunehmend auch komplexe Komponenten zur Verfügung, die betriebswirtschaftliche Verfahren abbilden. An diesen Komponenten lässt sich eine Fokussierung einzelner Application Server auf ein bestimmtes Anwendungsgebiet erkennen. So ist z. B. der Komponentenvorrat des Produkts der Firma Intershop, Enfinity [INTE00b, S. 17], stark auf die Entwicklung von Online-Shops ausgerichtet.

Tabelle 3-2: Übersicht marktführender Application Server

Hersteller	Produkt	Web-Adresse
Apple	WebObjects	www.Apple.com
BEA Systems	WebLogic	www.Beasys.com
IBM	IBM WebSphere	www.Software.ibm.com/webserves/
Intershop	Intershop Enfinity	www.Intershop.com
Oracle	Oracle Application Server	www.Oracle.com

Tabelle 3-2 zeigt eine Übersicht wichtiger Application Server. Ein umfassender und regelmäßig aktualisierter Vergleich der verfügbaren Application Server findet sich u. a. bei [FLAS02].

Bei Verwendung eines Application Servers erfolgt die Anwendungsentwicklung durch objektorientierte Programmierung, Auswahl, Anpassung und Ergänzung der vorhandenen Komponenten. Soweit diese eigene Editoren besitzen und ihre Verarbeitungslogik nach außen verbergen, ist nur eine Auswahl und Einstellung der benötigten Elemente erforderlich. Komponenten- und Frameworkeditoren sind leistungsfähige Adaptionswerkzeuge. Mit Enfinity können beispielsweise Softwarekomponenten interaktiv in einem grafischen Editor zu Prozessen zusammengefügt werden [INTE00a, S. 27].

Wesentlich schwieriger ist die Anpassung von vorhandenen Klassenstrukturen des Application Frameworks, die nur aus abstrakten Klassen bestehen und durch Einfügen spezieller Klassen für einen Anwendungsfall näher spezifiziert werden. Diese Form der Adaption setzt genaue Kenntnisse der Architektur des Frameworks, der darin enthaltenen Klassen und Fähigkeiten zur Implementierung neuer Klassen voraus. Das erfordert jahrelange Programmiererfahrung und detaillierte Einarbeitung. Allerdings können auf diese Weise vorhandene Funktionen unbegrenzt erweitert und ergänzt werden. Einige Application Server beinhalten in ihrem Funktionsumfang Setup-Assistenten, mit denen sich nach vordefinierten Modellen kleine, teilweise funktionsfähige Anwendungen in kürzester Zeit realisieren lassen (vgl. z. B. [APPL01, S. 29-31]). In umfangreichen Projekten werden solche Prototypen meist nur als Kommunikationsmittel während der Anforderungsdefinition eingesetzt.

Die Offenheit und der Vorrat an vordefinierten Funktionen machen Application Server zu produktiven und flexiblen Werkzeugen für erfahrene Entwickler, die in der Lage sind, aus den vorhandenen Elementen nach den vorher definierten Anforderungen geeignete Softwarebausteine auszuwählen, sie nach Bedarf zu ergänzen und zu betriebswirtschaftlichen Lösungen zu kombinieren. Die Verwendung vorgefertigter Komponenten führt zu einem erheblichen Zuwachs an Produktivität bei der Erstentwicklung eines Systems und unterstützt eine schnelle Anpassung an neue Anforderungen. Viele Application Server sind vollständig objektorientiert und damit für die Anwendung inkrementell-iterativer Softwareentwicklungsmodelle geeignet. Dies begünstigt den Einsatz dieser Produkte in Projekten mit größerem Umfang, hohen Anforderungen in Bezug auf den Innovationsgrad und die technische Komplexität der gewünschten Lösung. Als Werkzeug für KMU sind rein funktionsorientierte Standardsoftwarelösungen aufgrund der geringen methodischen und technischen Unter-

stützung für den effektiven Projekteinsatz allerdings wenig geeignet. Die Erweiterung einiger funktionsorientierter Standardsoftwarelösungen um Werkzeuge zur Abbildung komplexer Verfahren und Modelle lässt aber eine Entwicklung in Richtung betriebswirtschaftlicher Standardanwendungssoftware erkennen.

3.3.2.3 Betriebswirtschaftliche Standardanwendungssoftwarelösungen

Basis jeder Standardanwendungssoftware für elektronische Märkte ist ein betriebswirtschaftliches Grundmodell für ein bestimmtes Anwendungsgebiet. Es basiert im Allgemeinen auf einer der Ausprägungsformen elektronischer Märkte (siehe 2.5). Eine betriebswirtschaftliche Standardsoftware für elektronische Märkte sollte einen umfangreichen Vorrat an betriebswirtschaftlichen Funktionen bieten.

Zur Bewertung des Abdeckungsgrades individueller Anforderungen durch eine betriebswirtschaftliche Standardanwendungssoftware können die Kriterien Funktionsbreite und betriebswirtschaftliches Profil herangezogen werden. Die **Funktionsbreite** quantifiziert die Anzahl der betriebswirtschaftlichen Funktionen und Fachbereiche, die von der Software grundsätzlich abgedeckt werden. Das **betriebswirtschaftliche Profil** bewertet den Detaillierungsgrad und die Zahl der verfügbaren Alternativen bezüglich der Adaptionfähigkeit für eine Funktion [HUFG94, S. 70].

Zum Funktionsumfang einer betriebswirtschaftlichen Standardanwendungssoftware gehören neben dem Funktionsangebot für den Nutzer auch Back-Office-Funktionen, die die Administration des Systems im täglichen Betrieb unterstützen. Daneben profitieren Entwickler von leistungsfähigen Werkzeugumgebungen zur Adaption der Standardsoftware.

Umfang und Qualität der Funktionen im Back-Office-Bereich sind wichtige Qualitätsmerkmale, denn diese Funktionen lassen sich gut standardisieren und definieren einen erheblichen Teil der betriebswirtschaftlichen Grundfunktionalität jeder Anwendung. Der Funktionsumfang marktüblicher Produkte in diesem Bereich ist dennoch sehr unterschiedlich. Besonders problematisch ist die Integrationsfähigkeit betriebswirtschaftlicher Standardanwendungssoftwarelösungen in bestehenden IT-Strukturen. Obwohl die meisten Produkte bereits über Schnittstellen auf Basis standardisierter Datenaustausch- und Integrationstechniken wie XML, RMI und CORBA verfügen,

ist eine vollständige Integration mit bestehenden IT-Systemen aufgrund der datentechnischen Restriktion des Grundmodells der Standardsoftware kaum ohne umfangreiche Anpassungen von Konvertersystemen und Datenstrukturen möglich.

Die Vorgabe eines geeigneten betriebswirtschaftlichen Modells bietet vielfach schon beim Entwurf einer konkreten Lösung wertvolle Unterstützung. Auf der Grundlage des Modells können leistungsfähige Werkzeuge zur Adaption der Standardsoftware zu einer homogenen Umgebung integriert werden, wobei Methodik und Vorgehen bei Aufbau und Anpassung meist weitgehend durch die vorhandenen Adaptionswerkzeuge bestimmt werden. Nach der Adaptionstechnik lassen sich zwei Arten betriebswirtschaftlicher Standardsoftware unterscheiden.

Templatebasierte Standardsoftwarelösungen enthalten schablonenartige Vorlagen (Templates) für die Abarbeitung vorgegebener Teilprozesse des Grundmodells der Standardsoftware. Die Templates können durch strukturierte Programmierung in einfachen, meist proprietären Sprachen angepasst werden. Beispielsweise wurde von der Firma Intershop eine Template Extension Language zur Anpassung eingeführt (vgl. dazu [INTE00b, S. 297-359]). Anforderungen, die den Templatevorgaben entsprechen, sind auf diese Weise schnell zu realisieren. Weitergehende Anpassungen können nur von Entwicklern mit fundierter Kenntnis der einzelnen Templates, der proprietären Programmiersprache und des betriebswirtschaftlichen Grundmodells vorgenommen werden. Projektvorhaben, die das Grundmodell sprengen, sind im Prinzip nur durch umständliche Zusatzprogrammierung und Modifikation der Standardsoftware selbst vorzunehmen. Auch diese Möglichkeit der Adaption wird von vielen Herstellern zugelassen. Allerdings sind so modifizierte Lösungen kaum mehr releasewechselfähig. Umfangreiche Templateanpassungen sind im laufenden Betrieb des Produktivsystems nur auf einer separaten Entwicklungsplattform möglich, die dann im so genannten Staging abgeglichen werden müssen. Jede Modifikation der Funktionalität der Standardanwendungssoftware selbst macht ein automatisiertes Staging unmöglich und führt meist zu erheblichen Deaktivierungszeiten des Produktivsystems.

Technisch deutlich weniger anspruchsvoll ist die Verwendung **netzbasierter Assistenten**. Diese Produkte versuchen, dem Entwickler möglichst alle komplexen Problemstellungen von der Einarbeitung in die Funktion einzelner Templates über die Syntax einer Programmiersprache bis zur Installation und dem Betrieb eines Systems

zu ersparen. Der netzbasierte Assistent besteht aus einer vorgegebenen Folge von Eingabefeldern, in die alle Daten bezüglich einer individuellen Marktlösung eingegeben und als Parameter abgelegt werden. Diese werden im Hintergrund auf dem Server des Standardsoftwareanbieters nach programmierten Verfahren interpretiert und in ein ablauffähiges System umgesetzt. Eine Reihe netzbasierter Assistenten bietet eine sehr ausgereifte Benutzerführung und eine Vielzahl von Wahlmöglichkeiten bei der Anwendungsgestaltung, sodass auch Anwender mit geringsten technischen Kenntnissen in die Lage versetzt werden, eigene Lösungen zu entwickeln. Modifikationen des Modells sind bei diesen Systemen allerdings nicht vorgesehen. Trotz der Vielzahl an Wahlmöglichkeiten ist die technische Umsetzung eigener Strategien sehr stark eingeschränkt. Jede Anpassung kann nur im Rahmen der Freiheitsgrade des verwendeten Assistenten erfolgen.

Tabelle 3-3: Auswahl von Standardanwendungssoftwarelösungen für KMU

Kategorie	Hersteller	Produkt	Web-Adresse
Shop-Systeme	Intershop	Intershop 4.0	www.intershop.com
	OpenShop AG	OpenShop	www.openshop.com
	Strato AG	Intershop ePages (netzbasiert)	www.strato.de
	Yahoo	Yahoo!Store (netzbasiert)	www.yahoo.com
Auktions-Systeme	MOAI Technologies	LiveExchange	www.moai.com
	OpenSite	OpenSite Auction 4.0	www.opensite.com
Community-Systeme	Blaxxun	Blaxxun Community (umfassende Community-Lösung)	www.blaxxun.com
	Durand	Community Ware (Online-Konferenzsystem)	www.durand.com

Tabelle 3-3 zeigt eine Auswahl an betriebswirtschaftlichen Standardsoftwarelösungen, die für KMU geeignet sind. Insbesondere durch netzbasierte Assistenten wird die Schwelle für den Einstieg in die digitale Geschäftsabwicklung stark gesenkt. Dank massiver Werkzeugunterstützung sind auch KMU in der Lage, weitgehend selbstständig eigene Lösungen zu realisieren [ACKE00, S. 4]. Mit niedrigen Ein-

stiegspreisen haben die führenden Anbieter netzbasierter Assistenten beachtliche Zahlen an realisierten Systemen erreicht. In den meisten Fällen werden diese Anwendungen sehr eingeschränkt und oft nur für erste Experimente auf elektronischen Märkten genutzt, denn für ein langfristig erfolgreiches Projekt sind die Differenzierungsmöglichkeiten zu gering.

Tabelle 3-4: Bewertung der Produktklassen

	Individual-entwicklungs-umgebung	Application Server	Betriebswirtschaftliche Standardanwendungssoftware
Abdeckungsgrad	Wenig spezifisch.	Technische Funktionen werden meist umfassend abgedeckt. Betriebswirtschaftliche Verfahren sind nur durch Verwendung vorgefertigter Softwarekomponenten und Business Objects nutzbar.	Abhängig von der Funktionsbreite und dem betriebswirtschaftlichen Profil.
Anpassungsunterstützung	Keine Werkzeugunterstützung.	Editoren und Assistenten unterstützen die Anpassung.	Technische und in Ansätzen auch konzeptionelle Unterstützung gegeben.
Unterstützung der Systemverwaltung	Back-Office-Funktionen müssen individuell entwickelt werden.	Back-Office-Funktionen sind durch vorgefertigte Komponenten relativ schnell zu realisieren.	Back-Office-Funktionen sind Teil des betriebswirtschaftlichen Modells.
Flexibilität	Grundsätzlich jede Lösung realisierbar. Anpassungen im Betrieb erfordern aufwändige programmtechnische Maßnahmen.	Hohe Flexibilität bei Entwicklung und Anpassung durch Nutzung von Softwarekomponenten und Frameworks, langfristige Anpassung erfordert Methodik.	Hohe Flexibilität nur im Rahmen des betriebswirtschaftlichen Modells. Entwicklungen über das Modell hinaus werden schnell sehr aufwändig und führen zu Problemen bei Releasewechseln.
Integrationsunterstützung	Schnittstellen müssen individuell entwickelt werden.	Schnittstellen sind durch Individualentwicklung und Komponentenverwendung zu realisieren.	Integrationsfähigkeit hängt von der Kompatibilität der Schnittstellen des Produkts ab.

Die Deutsche Telekom AG hat einen netzbasierten Assistenten auf Basis des Intershop-Produkts ePages am unteren Ende ihrer Produktpalette für Shop-Systeme positioniert [DT99], um Kunden für höherwertige Projekte zu gewinnen. Ein Migrationspfad zu den Produkten Intershop 4.0 und sogar Enfinity ist vorgezeichnet.

3.3.3 Folgerungen für die Werkzeugauswahl

Die Situation bei der Auswahl eines Werkzeuges zum Aufbau eines elektronischen Marktes lässt sich aus der Sicht kleiner und mittlerer Unternehmen durch folgende Aussagen charakterisieren:

- Individualentwicklungsumgebungen bieten unbegrenzte Flexibilität, aber nur geringe Unterstützung bei der Systementwicklung. Der Aufwand bei der Programmierung und Dokumentation der Änderungen wächst mit zunehmendem Umfang des Systems.
- Durch Rückgriff auf die vordefinierten Softwarekomponenten einer funktionsorientierten Standardsoftwarelösung lässt sich die Produktivität bei der Erstentwicklung und laufenden Anpassung eines elektronischen Marktes gegenüber ausschließlicher Individualentwicklung deutlich steigern. Die konsequente Nutzung bereits vorhandener Funktionen und Modelle ist der Schlüssel, um den wechselnden Anforderungen in der gebotenen Geschwindigkeit qualitativ hochwertige und technisch stabile Lösungen entgegenzusetzen, ohne dass dadurch die Freiheitsgrade zur Realisierung individueller Anforderungen eingeschränkt werden. Voraussetzung dafür ist, dass erfahrene Entwickler einen für das jeweilige Projekt geeigneten Application Server auswählen, dessen Funktionsvorrat sie kennen und in der Lage sind, betriebswirtschaftliche Konzepte umzusetzen.
- Umfassende Unterstützung bietet eine betriebswirtschaftliche Standardanwendungssoftware durch Vorgabe eines betriebswirtschaftlichen Modells. Entspricht das Modell der Standardanwendungssoftware den Anforderungen eines Unternehmens so weit, dass das vorgegebene Lösungskonzept übernommen werden kann, entsteht dadurch ein enormer zusätzlicher Produktivitätsgewinn. Leistungsfähige Adaptionswerkzeuge, die auf dem Modell aufsetzen, bieten KMU ausreichend Unterstützung, um selbstständig eigene Lösungen zu realisieren.
- Die Wahl einer templatebasierten Standardanwendungssoftware oder eines netzbasierten Assistenten als Werkzeug zum Aufbau eines elektronischen Marktes bedeutet aber auch bereits zu Projektbeginn eine relativ starke Festlegung auf ein vorgegebenes Modell. Zukünftige Anpassungsmöglichkeiten sind auf die Funktionalität der jeweils aktuellen Version des gewählten Produkts beschränkt.

Folglich muss bei der Wahl eines Werkzeugs immer zwischen der Flexibilität und dem Grad der Unterstützung durch technische und betriebswirtschaftliche Vorgaben der Entwicklungsumgebung abgewogen werden.

Die Auswahl eines geeigneten Produkts kann erst nach einer exakten Formulierung der Systemanforderungen erfolgen. KMU stellt nicht nur die Definition der technischen und betriebswirtschaftlichen Anforderungen schon zu Beginn des Projektes vor große Schwierigkeiten. Auch die Beschaffung und Bewertung der notwendigen Informationen zur Werkzeugauswahl sind bei der Dynamik des Produktangebots schwierig und erfordern neben fundierten betriebswirtschaftlichen Kenntnissen vor allem technisches Know-how. Diese Unsicherheit muss zwangsläufig dazu führen, dass Netzeffekte bei der Produktwahl eine starke Rolle spielen und man sich vielfach sehr schnell und ohne genaue Betrachtung der eigenen Anforderungen auf die Produkte marktführender Unternehmen verlässt, was dann häufig zu wenig erfolgreichen Lösungen führt.

3.4 Vorgehensmodelle

Da es sich bei elektronischen Märkten definitionsgemäß um softwarebasierte Informationssysteme handelt, liegt es nahe, zum Aufbau von Marktsystemen bekannte methodische Ansätze aus der klassischen Softwareentwicklung einzusetzen. Seit den 70er Jahren werden zur besseren methodischen Unterstützung der Entwicklung von Softwaresystemen verschiedene Vorgehensmodelle beschrieben. Die speziellen Problemstellungen in Electronic Commerce-Projekten haben zur Entwicklung weiterer Modelle geführt, denn die Entwicklung elektronischer Marktsysteme geschieht unter speziellen Rahmenbedingungen, die überwiegend auf die starke Außenwirkung dieser Systeme zurückzuführen sind.

Hohe Anforderungen an Design und Fehlertoleranz der Benutzerschnittstelle

Jeder elektronische Markt ist eine Präsentation eines oder mehrerer Anbieter gegenüber den Kunden. Aus dieser Tatsache ergeben sich hohe Anforderungen an die Benutzerschnittstelle des Systems. Sie muss für möglichst breite Kundenkreise spontan verständlich und einfach zu bedienen sein. Häufig wird in diesem Zusammenhang besonders die Qualität des Designs der Benutzerschnittstelle hervorgehoben

[WARD99, S. 3]. Sie spielt vor allem bei der Gewinnung neuer Kunden im Business-to-Consumer-Bereich eine bedeutende Rolle. Im Web ist unter Berücksichtigung der technischen Restriktionen die Gestaltung aufwändiger, multimedialer Benutzerschnittstellen möglich. Die Entwicklung solcher Benutzerschnittstellen erfordert einen kreativen, typischerweise iterativen Prozess, der im Rahmen des Projekts zulässig sein muss.

Vielfalt der eingesetzten Technologien

In klassischen Softwareentwicklungsprojekten wird projektintern festgelegt, welche Technologien eingesetzt werden. Bei elektronischen Marktsystemen haben die Entwickler dagegen nur relativ geringen Einfluss darauf, welche Technologien die Anwender einsetzen. Diese werden im Internet durch die dort etablierten Standards und Dienste weitgehend vorgegeben. Der wichtigste Dienst im Internet ist das World Wide Web (WWW). Das WWW basiert technisch auf dem HTML-Standard. HTML (Hypertext Markup Language) ist eine Dokumentenbeschreibungssprache auf Basis des SGML-Standards. Auf ihre Strukturelemente reduziert lassen sich umfangreiche Inhalte sehr effizient übertragen und auf Client-Anwendungen (HTML-Browsern) mit standardisierter Technik verarbeiten. Die letztendliche Darstellung erfolgt dabei erst durch den Browser auf dem Rechner des Betrachters. Der Entwickler bzw. Autor von Inhalten, die in HTML dargestellt werden, kann lediglich die Struktur, nicht aber das exakte Layout der Darstellung beim Nutzer beeinflussen. Zudem ist das ursprünglich zur Übermittlung von Texten entwickelte HTTP-Protokoll nur zustandsorientiert [RUTK97] und bietet selbst keine Unterstützung zur Sicherung von Transaktionen. Derartige Schwächen führten dazu, dass eine ganze Reihe neuer Technologien entwickelt wurde, um den Leistungsumfang von HTML zu ergänzen. Bei der Realisierung elektronischer Märkte muss folglich eine Vielzahl verschiedener Technologien eingesetzt werden. Daraus ergeben sich hohe Anforderungen an die Planung und Bereitstellung der Entwicklungs- und Testumgebung sowie die Methoden, die zur Analyse und zum Entwurf des Systems eingesetzt werden.

Erfolg des Projektes wird im Wettbewerb bestimmt

Erfolgsentscheidend für einen elektronischen Markt ist die Akzeptanz der Kunden. Diese hängt nicht allein von der Qualität des jeweiligen Angebotes, sondern auch von der vergleichbarer Angebote ab. Online-Informationssysteme bieten den Vorteil, dass

während ihres Betriebes umfangreiche Datenmengen über die Nutzung anfallen (siehe 6.5). Diese sollten im Systembetrieb regelmäßig ausgewertet und für eine gezielte Erfolgskontrolle eingesetzt werden. Voraussetzung dafür ist, dass zuvor klare Projektziele definiert wurden.

Fortwährender Anpassungsbedarf

Elektronische Marktsysteme müssen häufig bereits unmittelbar nach Inbetriebnahme der Erstentwicklung kurz-, mittel- und langfristig immer wieder modifiziert werden. Schon die Erfahrungen aus der klassischen Softwareentwicklung zeigen, dass die überwiegende Zahl der realisierten Informationssysteme kurz nach ihrer Fertigstellung überarbeitet werden muss. Die Ursache dafür ist eine mangelhafte Erfassung der Nutzeranforderungen während der Entwicklung des Systems. In klassischen Softwareentwicklungsprojekten versucht man, diesem Problem mit Maßnahmen zur Verbesserung der Anwenderkommunikation zu begegnen. Während der Erstentwicklung eines elektronischen Marktsystems kann über die Anforderungen der avisierten Nutzer nur spekuliert werden. Erst die Nutzung des Systems im Betrieb unter realen Bedingungen zeigt das Verhalten unter Last und die tatsächliche Akzeptanz der Kunden. Die Reaktion der Kunden gibt Aufschluss über mögliche Schwachstellen in der strategischen Konzeption oder technischen Realisierung. Marktübliche Produkte bieten deshalb zahlreiche Möglichkeiten, die Nutzung eines Online-Angebots direkt zu messen, und üblicherweise geben Nutzer spontan Feedback über Fehler und Verbesserungsvorschläge. Diese Informationen sollten während der gesamten Betriebsdauer des Systems im Rahmen einer ständigen Erfolgskontrolle kontinuierlich ausgewertet und nötige Verbesserungen möglichst kurzfristig durchgeführt werden.

Mittelfristig entsteht Anpassungsbedarf, wenn sich grundlegende technische Neuerungen abzeichnen, etwa durch die Etablierung neuer technischer Standards. Die oben kurz dargestellte Entwicklung der verschiedenen Technologien macht deutlich, dass im Bereich der Nutzung von elektronischen Märkten innerhalb weniger Jahre bereits mehrfach neue Technologien entstanden sind. Tiefgreifende Innovationen (z. B. Einführung von Framesets zur Strukturierung von HTML-Seiten, Bild- und Tonkompressionsverfahren wie MP3 und Flash) erfordern mittelfristig umfangreiche Überarbeitungen, denn es besteht die Gefahr, im Wettbewerb Kunden zu verlieren, wenn neue Entwicklungen längere Zeit ignoriert werden.

Langfristig kann es auch zu grundlegenden Änderungen der Situation in einem Unternehmen kommen, die eine Modifikation des gesamten strategischen Ansatzes notwendig machen. In diesem Fall muss die gesamte Konzeption des Angebotes überdacht werden. Nicht selten ist dann eine Neuentwicklung notwendig.

Neben den klassischen Vorgehensmodellen der Softwareentwicklung finden sich vereinzelt weitere Vorschläge für die Projektdurchführung in Veröffentlichungen aus den Bereichen Kommunikationstechnologie und –design, die an die Anforderungen von Projekten aus diesem Bereich angepasst, aber weitgehend an die Modelle des Softwareengineering angelehnt sind. Die aus der Literatur bekannten Ansätze zur klassischen Systementwicklung lassen sich nach ihrer zeitlichen Abfolge der beschriebenen Aktivitäten in sequentielle Vorgehensmodelle, Verfahren des Prototyping und inkrementell-iterative Vorgehensmodelle differenzieren [SCHW97, S. 23]. Als eigenständige Verfahren haben nur sequentielle und inkrementell-iterative Vorgehensmodelle nennenswerte Bedeutung im Electronic Commerce. Obwohl viele web-basierte Lösungen immer noch auf schnell realisierten Prototypen aufbauen, können Prototyping-Verfahren nicht als eigenständiger Ansatz zur Projektdurchführung angesehen werden. Mangelhafte Dokumentation und Kontrollmöglichkeiten führen langfristig zum Scheitern von Prototyping-Projekten. Auch der Ansatz von BOEHM, der mit seinem Spiralmodell ein vollständiges Modell zur Entwicklung von Prototypen bis hin zum einsatzfähigen System beschreibt [BOEH88], hat nicht dazu geführt, dass sich Prototyping als methodisches Vorgehen zur Softwareentwicklung durchsetzen konnte. Nichtsdestoweniger kann experimentelles und exploratives Prototyping sowohl im Rahmen sequentieller als auch inkrementell-iterativer Vorgehensweisen einen wichtigen Beitrag zur Klärung von Teilproblemen leisten.

3.4.1 Sequentielle Vorgehensmodelle

Trotz heftiger Kritik wird eine große Zahl an Softwareentwicklungsprojekten immer noch nach sequentiellen Vorgehensmodellen durchgeführt. Sie beschreiben eine Folge von Projektphasen mit definierten Aktivitäten, die nacheinander abzuarbeiten sind. Der klassische Vertreter sequentieller Vorgehensmodelle ist das Wasserfallmodell nach ROYCE [ROYC70]. Im Wasserfallmodell ist eine Wiederholung einzelner

Phasen oder Aktivitäten nicht vorgesehen. Analog zur Metapher des Wasserfalls, in dem das Wasser stets abwärts seinem Ziel zufließt, führt das Wasserfallmodell in klar abgegrenzten, aufeinander folgenden Schritten zum definierten Ziel. Diese Vorgehensweise macht Projekte sehr gut planbar und liefert eindeutige Anhaltspunkte zur Definition und Kontrolle von Aktivitäten und Teilzielen. Bezüglich der einzusetzenden Methoden ergeben sich durch sequentielle Vorgehensmodelle keine Restriktionen oder Empfehlungen. Die Werkzeugauswahl ist in diesen Modellen oft untergeordneter Bestandteil einer Aktivität (siehe Abbildung 3-2).

Auf den ersten Blick scheinen sequentielle Vorgehensmodelle, gerade wegen Koordinationsmöglichkeiten heterogener Teams, für die Entwicklung von elektronischen Märkten geeignet. Erweiterungen um zusätzliche Phasen für besondere Problemstellungen sind ohne Verletzung des Grundkonzeptes der sequentiellen Durchführung von Projektaktivitäten möglich. BRUHN [BRUH97, S. 41-127] und SIEGEL [SIEG97, S. 205-283] schlagen für die Entwicklung elektronischer Märkte erweiterte Wasserfallmodelle vor. Den bekannten Aktivitäten der klassischen Systementwicklung gehen in diesen Modellen Aktivitäten zur Projekt- und Zielgruppenplanung voraus. Zum Projektabschluss erfolgt eine ausführliche Erfolgskontrolle.

In [LOHS02] wurde als zentrales Ergebnis eines mehrjährigen Forschungsprojekts ein Vorgehensmodell zur Einführung von Electronic Commerce-Lösungen in KMU vorgeschlagen (siehe Abbildung 3-2). Der Schwerpunkt liegt dabei auf der Einführung von Marktsystemen. In diesem Modell wird auf die Bedeutung einer individuellen Strategie explizit eingegangen und der Einfluss auf die Organisation und die Mitarbeiter des Unternehmens berücksichtigt. Im Anschluss an die Entwicklung der Strategie wird ein umfassendes Konzept erarbeitet, welches das geplante Produktangebot und die entsprechende Internet-Lösung vollständig beschreibt. Über die Aktivitäten zur Implementierung und den Betrieb des eigentlichen Systems hinaus werden weitere Aktivitäten zur projektbegleitenden Mitarbeiterqualifikation und zur Markteinführung des Systems beschrieben. Die Maßnahmen zur Markteinführung sind zeitgleich zur technischen und organisatorischen Implementierung des Systems durchzuführen. Den Projektabschluss bildet die Erfolgskontrolle, in der Messgrößen aus dem Systembetrieb mit der Zieldefinition vom Beginn des Projekts verglichen werden.

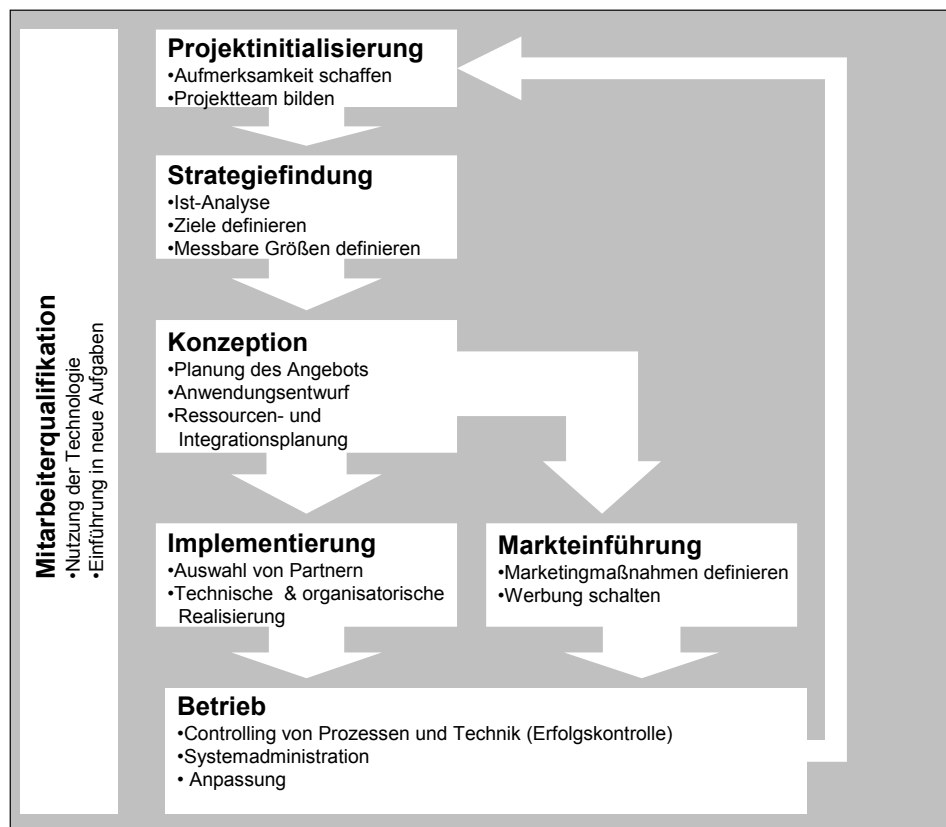


Abbildung 3-2: Vorgehensmodell zur Einführung von Electronic Commerce-Lösungen in KMU [LOHS02, S. 9]

Auch wenn in diesem Modell die Vorschrift der streng sequentiellen Durchführung des Wasserfallmodells zum Teil aufgehoben wird, können damit die Probleme bei der Verwendung sequentieller Modelle nicht beseitigt werden.

Der grundsätzliche Nachteil dieser Modelle liegt in der Prämisse, dass ein Softwaresystem bereits zu Beginn des Projekts vollständig beschrieben werden muss, um später exakt nach dieser Spezifikation realisiert zu werden. Änderungen in der Umwelt eines Unternehmens können nach Beendigung der Entwurfsphase nicht mehr berücksichtigt werden. Durch die strenge Trennung der Aktivitäten erhält der Auftraggeber nur in der ersten Phase des Projekts Gelegenheit, seine Wünsche zu äußern und hat aufgrund der technischen Ausrichtung späterer Phasen danach keinen weiteren Einfluss auf die Realisierung des Systems. Selbst die Erfassung der konkreten Systemanforderungen bereitet in der Praxis Probleme, da eine vollständige Beschreibung komplexer Softwaresysteme gleich zu Beginn eines Projektes kaum möglich ist. Die Folge sind Fehlplanungen, wenig zufriedenstellende Projektergebnisse und ein

hoher Änderungsbedarf oft unmittelbar nach Einführung der Systeme [CANT98, S. 91-93]. Eine Überarbeitung der Ergebnisse ist bei rein sequentieller Vorgehensweise aber nicht vorgesehen. Streng genommen müssten alle Änderungen in einem neuen Projekt durchgeführt werden, was in der Praxis kaum umgesetzt wird. Spätere Anpassungen des Systems sind innerhalb dieser Modelle nicht zulässig.

Diese Prämisse sequentieller Vorgehensmodelle geht nicht nur an den Rahmenbedingungen der Softwareentwicklung im Allgemeinen vorbei, sondern nimmt auch jede Möglichkeit, kurzfristig auf Umweltveränderungen reagieren zu können. Sie wirkt sich zudem hemmend auf kreative Prozesse aus [SCHW97, S. 28]. Sequentielle Vorgehensmodelle können daher trotz vordergründig vorhandener Vorteile bei der Planung und Kontrolle von Projekten kaum für die Entwicklung elektronischer Märkte empfohlen werden. Dass sequentielle Vorgehensmodelle trotz der in der Fachliteratur schon lange diskutierten Nachteile immer noch so häufig Anwendung finden, zeigt deutlich, wie hoch der Wert einer klar strukturierten und dadurch kontrollierbaren Vorgehensweise bei der Softwareentwicklung geschätzt wird.

3.4.2 Inkrementell-iterative Vorgehensmodelle

Im Gegensatz zu streng sequentiellen Vorgehensmodellen beschreiben inkrementell-iterative Vorgehensmodelle eine Folge von Aktivitäten, die in einer Reihe aufeinander aufbauender Phasen mit unterschiedlichen Zielsetzungen mehrfach durchgeführt werden. Besonders geeignet für eine inkrementell-iterative Vorgehensweise sind objektorientierte Softwareentwicklungsprojekte, denn der Entwurf eines vollständigen, sinnvoll hierarchisierten und meist hoch granularen Objektmodells erfordert einen iterativen Prozess der Objektfindung und Abstraktion in mehreren Zyklen. Durch die Prinzipien der Objektkapselung und Klassenbildung der Objektorientierung wird ein inkrementeller Ausbau der Ergebnisse jeder Iteration möglich [BOOC97, S. 289-312].

Voraussetzung für zielorientierte Durchführung der inkrementell-iterativen Systementwicklung ist ein Vorgehensmodell, das genaue Anhaltspunkte zur Kontrolle der Iterationsschritte liefert, und eine Notation, die den Entwurf des Systems effizient unterstützt. Als Standard zur Modellierung objektorientierter Systeme hat sich mittlerweile die Unified Modelling Language (UML) [OMG01] weitgehend durchge-

setzt. Die einzelnen Diagrammtypen der UML erlauben die Beschreibung von Softwaresystemen aus unterschiedlichen Blickwinkeln und Abstraktionsebenen. In [MÜLL99] und [EICK99] werden mehrere inkrementell-iterative Vorgehensmodelle zur Anwendung der UML im Projekt diskutiert. Das derzeit Bekannteste ist der Unified Process der Firma Rational [KRUC00].

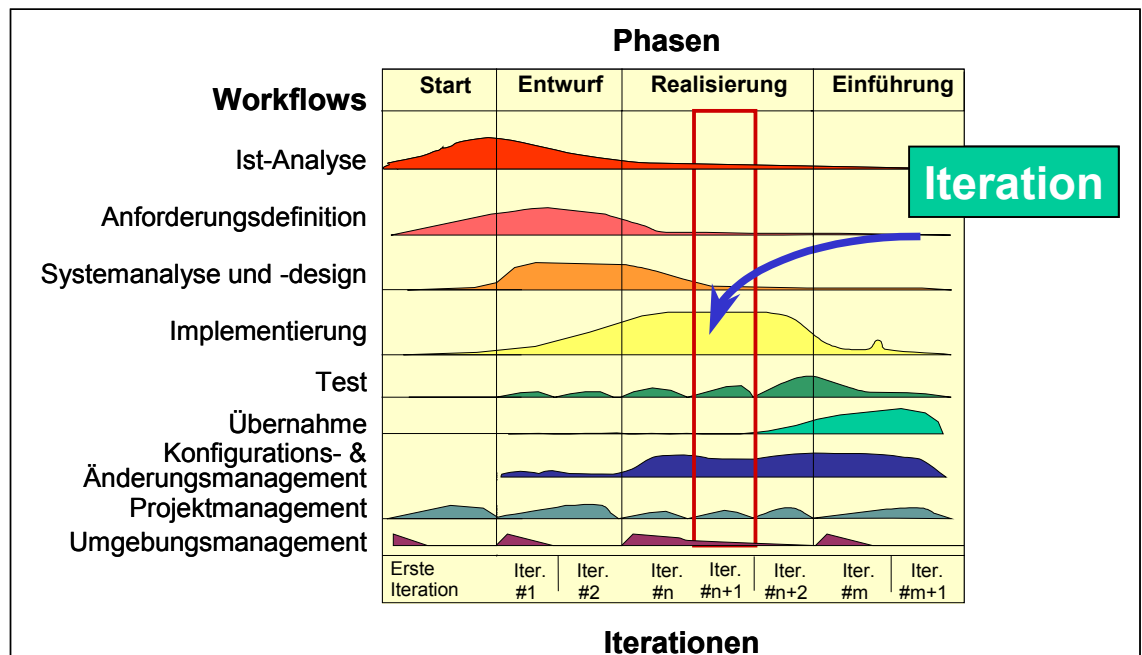


Abbildung 3-3: Aufwandsverteilung im Rational Unified Process (in Anlehnung an [KRUC00, S. 23])

Ziele des Rational Unified Process sind der konsequente Einsatz von Darstellungsmethoden zur Modellierung des Systems, iterative Softwareentwicklung im Team und die Verwendung bereits vorhandener Komponenten. Nach dem Rational Unified Process werden Softwaresysteme allgemein in vier Phasen mit definierten Zielen entwickelt, die inkrementell aufeinander aufbauen. Dabei wird das Zielsystem auf verschiedenen Abstraktionsebenen und aus unterschiedlichen Perspektiven betrachtet. Für alle Aktivitäten ist im Rational Unified Process in einem eigenen Workflow definiert, welche Teilergebnisse in jeder Phase zu erarbeiten sind. Dabei spielt die UML nicht nur zur Modellierung des Systems, sondern auch zur Aufnahme und iterativen Verfeinerung der Nutzeranforderungen eine entscheidende Rolle. Zur Unterstützung beim Entwurf der Benutzerschnittstelle ist zusätzlich die Erstellung eines Prototypen mit geringem Funktionsumfang vorgesehen. Begleitend zu den

einzelnen Phasen werden Aktivitäten des Projektmanagements, der Abstimmung und Anpassung aller im Projekt erzeugten Dokumente und Systeme sowie der Wartung der gesamten Entwicklungs- und Testumgebung beschrieben.

Durch die Darstellungsmittel der UML wird die Kommunikation zwischen den Projektbeteiligten in einem iterativen Softwareentwicklungsprozess effizient unterstützt und so eine frühzeitige Überprüfung aller Teilergebnisse möglich. Die im Rational Unified Process geforderte detaillierte Systemdokumentation und der Rückgriff auf existierende Komponenten tragen dazu bei, dass auch umfangreiche Softwaresysteme schnell umgesetzt und langfristig an veränderte Anforderungen angepasst werden können. Dies sind Eigenschaften, die den Rational Unified Process bzw. ähnliche detaillierte inkrementell-iterative Vorgehensmodelle für die Entwicklung elektronischer Märkte geeignet erscheinen lassen.

In [WARD99] wird eine Anpassung des Rational Unified Process zur Entwicklung web-basierter Systeme beschrieben. In dieser speziellen Ausprägung des Modells gewinnen die Aktivitäten zum Entwurf der Benutzerschnittstelle an Bedeutung. Anwendungsfall- und Klassendiagramme werden eingesetzt, um die Struktur einer Lösung zu definieren und daraus unmittelbar allgemeine Designrichtlinien für die Anwendungsgestaltung abzuleiten. Ergänzt um spezielle Designframeworks und vorgefertigte Systemkomponenten wird damit die Realisierung von Electronic Commerce-Anwendungen stark unterstützt. Die Darstellungshilfen der UML können die Erfahrung qualifizierter Systementwickler nicht ersetzen. Auch bei der Modellierung dynamischer Geschäftsprozesse sind die grafischen Darstellungen der UML nur unzureichende Hilfsmittel [THOM96, S. 21; EICK99, S. 363].

Betriebswirtschaftliche Überlegungen, die Grundlage jedes Electronic Commerce-Projektes sein sollten, werden von der technisch ausgerichteten Modellierungsmethodik allerdings gar nicht erfasst. Alle diese Aspekte müssen im Projekt durch eine umfangreiche zusätzliche Dokumentation ergänzt werden. Mindestens diese Dokumente sind über den gesamten Projektverlauf hinweg weitgehend von Hand zu aktualisieren. Im schnelllebigen Geschäft der Entwicklung für elektronische Märkte werden solche Arbeiten vorwiegend aus Zeitgründen häufig vernachlässigt. Die mangelnde Integration von Entwicklungsmethodik, Systemdokumentation und dem realen System führt dann in recht kurzer Zeit zum Zusammenbruch des Entwicklungsprozesses. Ein langfristig kontinuierlicher Anpassungsprozess ist im Bereich elektronischer Märkte nur

durch weitgehende Integration von Entwicklungsmethodik über alle Stufen des Systementwurfs und der Dokumentation mit der eigentlichen Entwicklungsumgebung zu erreichen.

Darüber hinaus zeigt die Tatsache, dass Prototyping häufig zur Klärung der Nutzeranforderungen in anderen Verfahren eingesetzt wird, dass der Umgang mit einem funktionierenden (Teil)-System ein unverzichtbares Anschauungs- und Kommunikationsmittel ist.

3.5 Zusammenfassung

Aus der Bewertung der verfügbaren Vorgehensmodelle zum Aufbau elektronischer Märkte lassen sich folgende Ergebnisse ableiten:

- Die Entwicklung elektronischer Märkte verlangt nach einer iterativen Vorgehensweise, bei der die Überarbeitung und Anpassung bisheriger Ergebnisse jederzeit kurzfristig möglich ist. Streng sequentielle Vorgehensmodelle hemmen kreative Prozesse und verhindern marktgerechte Reaktionen.
- Das dynamische Verhalten von Systemen und Nutzeranforderungen bezüglich Design und Benutzerführung lässt sich nur an lauffähigen (Teil)-Systemen adäquat veranschaulichen.
- Durch die Nutzung vorgefertigter Softwarekomponenten und Modellvorschläge kann der Aufbau eines elektronischen Marktes deutlich beschleunigt werden. Die Aussagekraft abstrakter Modellierungshilfen ist allerdings begrenzt, sodass bei der Entwicklung und bei jeder Anpassung ein erfahrener Systementwickler eingeschaltet werden muss, der das Modell in ein lauffähiges System überführt und ständig die Integrität zwischen System und Modell sicherstellt.

Die Untersuchung der verfügbaren Werkzeuge und Vorgehensmodelle zum Aufbau elektronischer Marktsysteme macht deutlich, dass die zur Zeit verfügbaren Methoden und Werkzeuge KMU entweder zu geringe Unterstützung bieten oder zu inflexibel sind. Ein Problem, das die am Markt befindlichen Werkzeuge bisher überhaupt nicht lösen können, ist die Unterstützung der Formulierung von Anforderungen und ihre

Umsetzung in ein Lösungskonzept. So sind insbesondere KMU häufig auf die Hilfe externer Berater angewiesen, was nicht nur erhebliche Kosten, sondern auch Probleme bei der Vermittlung der eigenen Vorstellungen schafft. Es entsteht eine langfristige Abhängigkeit, die zum Hindernis für dynamische Reaktionen auf Veränderungen oder neue Marktanforderungen werden kann. Im Zusammenhang mit dem Aufbau und dem Betrieb von Lösungen für elektronische Märkte ist dieser Nachteil besonders gravierend, da diese durch technische Innovationen und die Entstehung neuer Geschäftsmodelle einer besonders hohen Dynamik unterworfen sind.

4 Systementwicklung durch wissensbasierte Adaption

Ähnliche Probleme, wie sie im vorhergehenden Kapitel deutlich wurden, sind – wenn auch mit anderen Auswirkungen - bereits seit langem aus der Softwareentwicklung bekannt. Aus diesem Grunde wurde am Lehrstuhl für Wirtschaftsinformatik der Universität Würzburg ein neuer Ansatz zur Adaption von Softwarelösungen entwickelt. Er basiert auf der Adaption einer Standardanwendungssoftware mit Hilfe eines wissensbasierten Systems und ermöglicht eine völlig neue Vorgehensweise bei der Systementwicklung, die als Continuous Systems Engineering bezeichnet wird. Die in diesem Zusammenhang entwickelten Verfahren und Werkzeuge werden seit 1992 zur Adaption von SAP/R3-Systemen eingesetzt.

In 4.1 wird zunächst die Idee des Continuous Systems Engineering (CSE) vorgestellt. In 4.2 wird ein Konzept zur Realisierung des CSE eingeführt und in 4.3 auf seine möglichen Einschränkungen oder notwendigen Anpassungen zur Verwendung für die Entwicklung elektronischer Märkte geprüft.

4.1 Continuous Systems Engineering

Continuous Systems Engineering beschreibt ein Konzept zur Einführung von Standardanwendungssoftwarelösungen in Unternehmen. Im Gegensatz zu der Idee des **Business Process Reengineering (BPR)** geht CSE dabei nicht von einem einmaligen Umbau der Unternehmensorganisation und der Geschäftsprozesse zum Zeitpunkt der Einführung der Standardsoftware aus, sondern von einem kontinuierlichen Anpassungsprozess, denn in der Realität unterliegt die Organisationsstruktur eines Unternehmens einem ständigen Wandel. Wechselnde Umgebungsbedingungen zwingen jedes Unternehmen zur Anpassung. Die Potentiale moderner Informationsverarbeitung können aber nur dann ausgeschöpft werden, wenn diese bestmöglich auf die jeweilige Organisationsstruktur angepasst ist. Dies kann nur durch eine kontinuierliche Anpassung von Organisationsstruktur und Informationstechnologie erreicht werden.

Hier setzt die Idee des **Continuous Systems Engineering** an. Im CSE spielt die Adaption von Informationssystemen in dem in 1.1 eingeführten Sinn einer kontinuierlichen Anpassung an Umweltveränderungen eine zentrale Rolle. Ziel des Continuous Systems Engineering ist nicht die langfristige Planung von Organisationsformen und Abläufen und deren Abbildung in einem Informationssystem, sondern „die kontinuierliche Anpassung des organisatorischen Geschehens in Verbindung mit der Adaption des Informationssystems...“ [THOM96, S. 78]. CSE steht damit im Gegensatz zu der Vorgehensweise des BPR, nach der bei der Einführung einer Standardanwendungssoftware zunächst die Unternehmensorganisation an die Möglichkeiten der Software und anschließend die Standardanwendungssoftware in einer einmaligen Aktion (Customizing) an die neue Organisationsstruktur anzupassen ist.

CSE wird realisierbar, wenn zwei Grundvoraussetzungen erfüllt sind. Zum einen muss eine Anpassung der Informationsverarbeitung an organisatorische Änderungen ohne aufwändige Softwareentwicklung möglich sein, zum anderen müssen die Maßnahmen zur Anpassung und die sie auslösenden organisatorischen Bedingungen in einer schnellen, aber dennoch unmittelbar nachvollziehbaren und im Anpassungsprozess gezielt verwendbaren Form erfolgen.

Diese beiden Forderungen sind durch den Einsatz moderner softwarebasierter Werkzeuge zu erfüllen. Zum einen wird eine Standardanwendungssoftware mit den Eigenschaften einer **Softwarebibliothek** benötigt, die bereits einen umfassenden Vorrat an standardisierten, aber dynamisch adaptionsfähigen Softwarelösungen für anwendungsbezogene Problemstellungen beinhaltet. Sie wird als **Standardanwendungssoftwarebibliothek** bezeichnet. Zum anderen ist ein **Anforderungsnavigator** erforderlich, in dem betriebswirtschaftliche Grundkenntnisse in Form von Regeln abgebildet sind, der die jeweiligen Anforderungen erfasst und die notwendigen Anpassungen der Softwarebibliothek weitgehend automatisiert vornimmt [THOM96, S. 79]. Welche Anforderungen eine Softwarebibliothek im Allgemeinen und insbesondere als Werkzeug für die Entwicklung elektronischer Märkte zu erfüllen hat, wird im Rahmen des Entwurfs der Standardanwendungssoftware für elektronische Märkte (siehe 5.1) diskutiert.

Durch den Rückgriff auf vorhandene Softwarebausteine einer Softwarebibliothek und ihre werkzeugunterstützte Anpassung mit dem Anforderungsnavigator können informationstechnische Lösungen zunächst sehr schnell realisiert und bei Bedarf mit

geringem Aufwand geändert werden. Beim Continuous Systems Engineering wird deshalb bewusst schon nach kurzer Analyse und Konzeption eine erste Lösung implementiert. Im Unterschied zu den ersten Ergebnissen beim Prototyping besitzt dieses so genannte **Baseline-System** bereits eine hohe Stabilität. Das Baseline-System kann unmittelbar im Produktivbetrieb eingesetzt werden und liefert wertvolle Erkenntnisse über notwendige Verbesserungen. Diese Erfahrungen können kurzfristig zu weiteren Anpassungen genutzt werden. Das betriebswirtschaftliche Grundwissen des Einführungswerkzeugs hilft bei jeder Überarbeitung, ungünstige Anpassung zu vermeiden, sodass in immer kürzeren Adaptionsschritten eine stetig verbesserte Abstimmung von Organisationsstruktur und Informationsverarbeitung bei gleichzeitig kontinuierlicher Weiterentwicklung der abgebildeten Prozesse in Richtung einer optimierten Lösung erreicht wird (gerichtete Navigation).

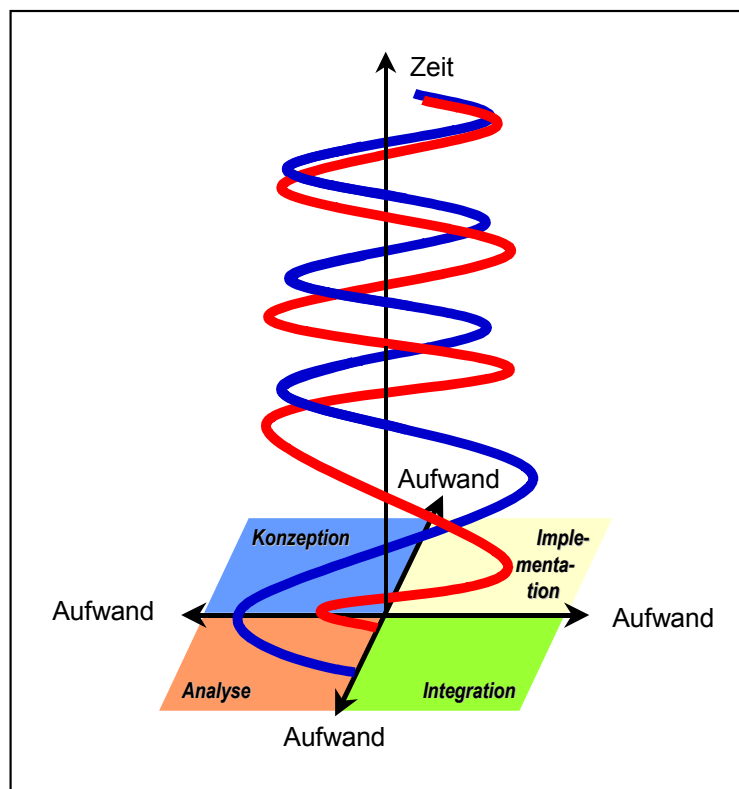


Abbildung 4-1: Integration von Informationsverarbeitung und Organisation zu einer stetig verbesserten Lösung durch CSE [THOM96, S. 80]

In Abbildung 4-1 ist dieser Anpassungsprozess in drei Dimensionen dargestellt. Die Ausdehnung der Fläche zeigt den Aufwand für die einzelnen Entwicklungsphasen.

Die innere Spirale (hell) symbolisiert die Informationsverarbeitung, die äußere Spirale (dunkel) die Unternehmensorganisation. Bei jedem Adaptionsschritt kann durch gerichtete Navigation der Aufwand für die Entwicklung einer neuen Systemversion verringert werden. Durch sofortige Erprobung des Systems im produktiven Unternehmen werden über die Zeit Informationsverarbeitung und Unternehmensorganisation kontinuierlich aufeinander abgestimmt und gemeinsam verbessert. Bei grundlegenden Veränderungen der Umweltbedingungen eines Unternehmens, z. B. durch neue Marktanforderungen, technische Innovationen oder organisatorische Umstrukturierungen im Unternehmen, kann es zu Sprüngen im Anpassungsprozess kommen. Die neue Ausgangssituation ist dann Ansatzpunkt einer neuen Folge gezielter Anpassungsschritte.

Der Einsatz eines Anforderungsnavigators verspricht insbesondere bei der Entwicklung und dynamischen Anpassung elektronischer Märkte für KMU eine Reihe entscheidender Vorteile:

- Unterstützt durch einen leistungsfähigen Anforderungsnavigator sollte ein großer Teil aller Aktivitäten im Projekt vom betriebswirtschaftlichen Fachpersonal weitgehend selbstständig durchgeführt werden können. Erfahrungen aus vielen Projekten zur Einführung von Standardanwendungssoftwaresystemen zeigen, dass die Ist-Analyse und Erarbeitung des Soll-Konzepts nicht nur eine Quelle von Fehlentwicklungen, sondern auch ein erheblicher Kostenfaktor [THOM96, S. 90f.] ist. Werden diese Phasen durch einen Anforderungsnavigator unterstützt, führt das zu einer deutlichen Kostenreduktion. Für KMU wird damit die Schwelle zur Nutzung elektronischer Märkte entscheidend gesenkt.
- Die für elektronische Märkte notwendige fortwährende Anpassung ist zentraler Bestandteil des CSE. Das betriebswirtschaftliche Grundwissen des Anforderungsnavigators führt von Anfang an zu einer Lösung, die die entscheidenden betriebswirtschaftlichen Potentiale und organisatorischen Rahmenbedingungen eines Unternehmens berücksichtigt. Die Analyseergebnisse liegen strukturiert vor, sodass sie gespeichert und bei Bedarf unmittelbar im nächsten Adaptionsschritt verfügbar sind. Auf diese Weise führt jede Anpassung zu einem verbesserten Ergebnis.

- Die Analyseergebnisse des Anforderungsnavigators beruhen auf allgemein gültigen Erkenntnissen und Regeln aus dem betriebswirtschaftlichen Umfeld der Anwendung und nicht auf den subjektiven Erfahrungen einzelner Berater. Das Wissen des Anforderungsnavigators und die Informationen, die zu einer Lösung geführt haben, sind formalisiert und können über lange Zeit gespeichert und KMU kostengünstig zur Verfügung gestellt werden. Daraus ergibt sich ein weiteres erhebliches Einsparungspotential, das speziell vielen KMU die Realisierung unternehmensspezifischer Lösungen überhaupt erst ermöglicht.
- Durch dynamische Adaption einer vorhandenen Lösung können kurzfristig und in vielen Fällen ohne nennenswerte Unterbrechung des laufenden Betriebs Anpassungen vorgenommen werden, auch wenn dies fundamentale Änderungen von Teilen der Anwendung bedeutet. Dadurch wird es möglich, mit geringem Zeitverzug und zu niedrigen Kosten kurzfristig auf Marktanforderungen zu reagieren.

4.2 Adaption von Standardanwendungssoftware nach dem ODYSSEUS-Konzept

Die Umsetzung des CSE im Projekt erfordert eine methodische, zielorientierte Vorgehensweise und geeignete Werkzeuge. Die konzeptionellen, technischen und betriebswirtschaftlichen Grundlagen der Adaption betriebswirtschaftlicher Softwarebibliotheken wurden von HUFgard entwickelt und beschrieben [HUFgard94]. Die Ergebnisse dieser Arbeit führten zur Entwicklung des Adaptionkonzepts ODYSSEUS (**O**rganisatorisch- **d**ynamische Spezifikation von Softwaremodulen entsprechend der Unternehmensstruktur) und zur Realisierung des regelbasierten Anforderungsnavigators LIVE KIT zur Einführung von SAP/R3 durch die Firma Siemens. Durch den Einsatz dieses Adaptionswerkzeuges konnte der Aufwand für die Einführung von SAP/R3 in mittelständischen Unternehmen deutlich verringert werden.

4.2.1 Adaptionarten

Die Adaption einer Standardsoftware kann durch **Auswahl**, **Anpassung** und **Ergänzungsentwicklung** erfolgen [HUFG94, S. 178-182]. Um einen effizienten Werkzeugeinsatz sicherzustellen, müssen diese Adaptionarten nach dem Subsidiaritätsprinzip in eine logische Reihenfolge geordnet werden, d. h., dass zunächst weitestgehend durch Auswahl, dann durch Anpassung und zuletzt durch Ergänzung die Adaption an die Anforderungen eines Unternehmens erreicht wird [HUFG94, S. 177]. Die Adaptionentscheidungen Auswählen, Anpassen und Ergänzen werden durch einen Anforderungsnavigator unterstützt, der die Anforderungen des Anwenderunternehmens aufnimmt, die Konsistenz der Entscheidungen und die betriebswirtschaftliche Korrektheit im Abgleich mit den Systemanforderungen sicherstellt und das Ergebnis für den Anwender transparent macht [THOM96, S. 89].

Auswahl

Der erste Schritt der Adaption ist die Auswahl der benötigten Elemente der Softwarebibliothek. Diese Art der Adaption ist besonders effizient, wenn Komponenten gewählt werden, die möglichst ohne Anpassung anschließend im Zielsystem eingesetzt werden können. Voraussetzung für die Realisierung individueller Informationssysteme durch Auswahl und Kombination vorhandener Softwareelemente ist eine sehr hohe Modularisierung der Softwarebibliothek und eine Minimierung der Schnittstellen zwischen den einzelnen Modulen sowie eine klare Definition für Schnittstellenimplementierung. Wie in 3.2.2.1 gezeigt wird, lassen sich diese Prämissen durch objektorientierte bzw. komponentenbasierte Softwareentwicklung gut erfüllen.

Anpassung

Im ODYSSEUS-Konzept ist Anpassung im Zusammenhang mit der Adaption von Standardanwendungssoftware im Gegensatz zur gezielten Modifikation von Programmcode nur auf Anpassung von Softwareelementen im vordefinierten Rahmen beschränkt. Unter Anpassung im vordefinierten Rahmen versteht man „die Ausprägung von Elementen der Softwarebibliothek durch Wahl von Parameterwerten bez. ihrer funktionalen Eigenschaften, Aktivierung von Datenobjekten und –beziehungen oder Veränderung der Benutzerpräsentation, die vom Entwickler vor-

gedacht und vorbereitet sind, sodass sie immer noch auf Standardvorgaben zurückführbar sind...“ [HUFG94, S. 180].

Diese Einschränkung ist Voraussetzung für eine dynamische, werkzeuggestützte Adaption von Informationssystemen. Programmtechnische Anpassungen, etwa durch Erweiterung oder Überschreiben von Funktionen bzw. Methoden, wie sie in der objektorientierten Systementwicklung üblich sind, fallen nicht unter diese Art der Adaption.

Die Freiheitsgrade bei der Anpassbarkeit einer Software im vordefinierten Rahmen hängen stark von der Qualität des Systementwurfs ab. Er muss eine allgemein gültige Softwarearchitektur und Softwareelemente vorgeben, die einerseits standardisiert, aber gleichzeitig auf möglichst viele Anwendungsfälle hin anpassbar sind. Um die dynamische Adaptionfähigkeit einer Softwarelösung zu gewährleisten, muss ein Datenmodell entwickelt werden, in dem alle Parameterwerte separat von der Implementierung der verwendeten Softwareelemente verwaltet werden. Die Entwicklung eines geeigneten Modells für die Abbildung elektronischer Märkte erfolgt in 5.3.

Ergänzung

Unter Ergänzung wird die Programmierung bzw. der Zukauf zusätzlicher Funktionen verstanden, die auf definierten Schnittstellen der Softwarebibliothek aufsetzen. Nur wenn die Schnittstellen zwischen der Softwarebibliothek und den ergänzten Funktionen exakt definiert sind, bleibt auch bei Ergänzungsentwicklungen die dynamische Releasefähigkeit der Softwarebibliothek erhalten.

4.2.2 Adaptionsprinzipien

Nach der Idee des CSE können die drei Adaptionarten eingesetzt werden, um in kurzer Zeit nach ersten erkennbaren Anforderungen ein funktionierendes System aufzubauen und im Laufe eines stetig fortschreitenden Lernprozesses kontinuierlich zu verbessern. Aus den Potentialen und Zielen des CSE ergeben sich für die Anwendung der Adaptionarten eine Reihe grundlegender Prinzipien [THOM96, S. 90-95].

4.2.2.1 Heuristische Analysestrategie

Akzeptiert man, dass es sich bei jeder Softwareentwicklung um einen Lernprozess handelt, bei dem das Ziel erst allmählich klar wird, die Zielsetzung im Wettbewerb mit Unternehmen und die technische Lösung kontinuierlich überarbeitet und falls nötig modifiziert werden muss, wird auch klar, dass dieser Prozess nicht durch ein optimierendes Verfahren zu einem Ergebnis geführt werden kann.

Die drei Adaptionarten sind daher nach einem heuristischen Verfahren anzuwenden. Im Vordergrund steht der wirtschaftliche Einsatz der Potentiale der Softwarebibliothek durch Abgleich der Anforderungen mit dem dort vorhandenen Lösungsangebot. Nach der Idee des CSE steht nicht eine aufwändig konzipierte „Ideallösung“ am Beginn eines Projekts. Erstes Ziel ist vielmehr die Bereitstellung einer zufriedenstellenden Eröffnungslösung. Dies entbindet auch von der Notwendigkeit, sehr weit in die Zukunft zu planen und über zukünftige Entwicklungen zu spekulieren, die im Bereich des Electronic Commerce ohnehin kaum vorhersehbar sind. Grundlage der Anforderungsanalyse sind jeweils die erkennbaren Bedingungen des Unternehmens. Dabei darf der Aspekt der Wirtschaftlichkeit nicht außer Acht gelassen werden, d. h., es sollten nur Anforderungen formuliert werden, deren erwarteter Nutzen die Realisierungskosten nicht übersteigt.

Der Abgleich der Anforderungen mit der Softwarebibliothek geschieht in drei Phasen (siehe Abbildung 4-2).

Reduktion

In der ersten Phase wird durch Ermittlung der aktuellen Anforderungen und der Auswahl der einführungsrelevanten Objekte aus dem Lösungsvorrat der Softwarebibliothek eine Reduktion des Anpassungsaufwandes erreicht. Nach dem Subsidiaritätsprinzip werden dabei vorrangig Softwareelemente ausgewählt, die ohne oder nur mit geringem Anpassungsaufwand genutzt werden können. Der Anforderungsnavigator muss dazu die Anforderungen an das Zielsystem analysieren und danach durch Bewertung der betriebswirtschaftlichen Profile die geeigneten Softwarekomponenten auswählen. Das Ergebnis der Auswahl legt das Zielsystem in seiner Funktionalität bereits in groben Zügen fest. Es führt zur Reduktion des Einführungsaufwandes in den nachfolgenden Phasen und ist Basis der Einführungsdokumentation.

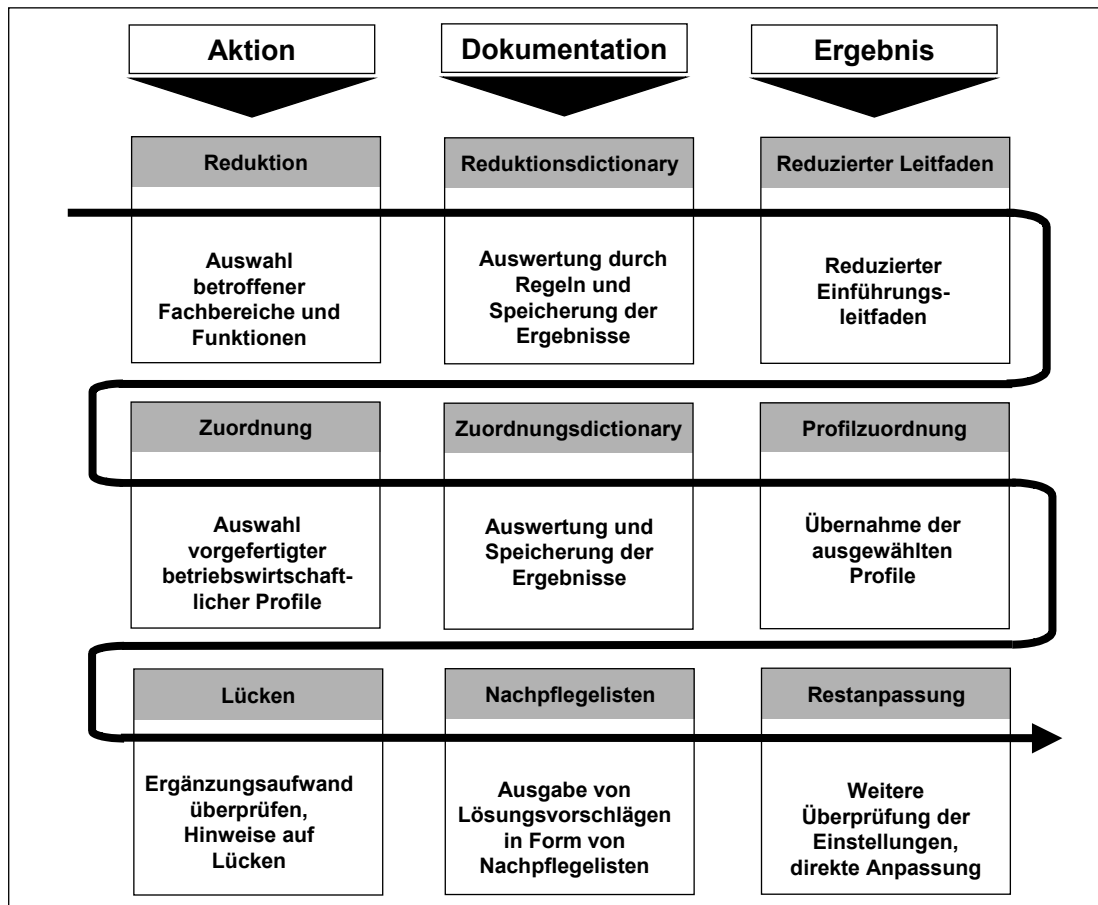


Abbildung 4-2: Phasenweise Realisierung des ODYSSEUS-Konzepts für die R3-Einführung (in Anlehnung an [VOGE98, S. 61])

Zuordnung

In der zweiten Phase wird die reduzierte Menge der ausgewählten Objekte in ihrer Funktion angepasst. Erst durch die Zuordnung eines betriebswirtschaftlichen Profils im Rahmen der vordefinierten Anpassungsmöglichkeiten erhalten die betroffenen Softwareelemente ihre konkrete Ausprägung bezüglich ihrer Funktion innerhalb des Gesamtsystems. Bei der Zuordnung sollten mehrere betriebswirtschaftliche Profile zur Verfügung stehen, aus denen die geeignetste Lösung gewählt werden kann. In einem integrierten System aus Adaptionswerkzeug und Standardanwendungssoftware kann die Zuordnung betriebswirtschaftlicher Profile in einem inkrementellen Prozess erfolgen, sodass verschiedene Alternativen unmittelbar am System getestet werden können. Technische Voraussetzung dafür ist, dass jedes gewählte Profil eine lauffähige Konfiguration definiert.

Nachpflege von Lücken

In der letzten Phase des Adaptionprozesses wird auf Anforderungen hingewiesen, die von der Standardanwendungssoftware nicht abgedeckt und folglich durch Ergänzungsentwicklungen gelöst werden müssen. In [HUFG94, S. 220] wird die Identifikation von Lücken und deren Beseitigung zwar als wichtige Phase des Adaptionprozesses definiert, aber methodisch nicht näher ausgeführt.

Zur Identifikation von Lücken werden bei der Einführung einer Standardanwendungssoftware mit einem Anforderungsnavigator an mehreren Stellen wichtige Informationen gewonnen. Schon die Dokumentation der Reduktionsphase liefert Anhaltspunkte darüber, ob die Auswahl der Komponenten aus der Softwarebibliothek die Anforderungen des Unternehmens abdeckt. Mit Hilfe interaktiver Werkzeuge, die die Profilmzuordnung in der nachfolgenden Adaptionphase visualisieren, können ebenfalls Lücken sichtbar gemacht werden.

Ein zusätzliches Instrument zur Identifikation von Lücken ist die Auswertung von Anfragen und Hinweisen, die im Rahmen der im Internet üblichen Kommunikation zwischen Nutzern und Anbietern der Online-Systeme im laufenden Betrieb anfallen. Durch sie lassen sich Informationen über unvorhergesehene Kundenanforderungen gewinnen, die Ergänzungsentwicklungen oder einen neuen Adaptionsschritt notwendig machen.

4.2.2.2 Frühzeitige Implementierung lösbarer Aufgaben

Durch den Einsatz einer Standardanwendungssoftware wird die schnelle Implementierung einer ersten Eröffnungslösung möglich. Sie spiegelt den technischen und konzeptionellen Entwicklungsstand zu einem frühen Zeitpunkt im Projekt wider und kann als das anschaulichste denkbare Objekt zur Überprüfung des angestrebten Projektziels herangezogen werden.

Vor diesem zentralen Vorteil der CSE-Philosophie schrecken viele zurück, deren Erfahrungswelt sich auf klassische phasenorientierte Softwareentwicklung gründet, in der sich Fehlentscheidungen nur mit großem Aufwand korrigieren lassen [THOM96, S. 94]. Die Angst, Lösungen vorschnell zu realisieren und dann auch in Betrieb zu nehmen, ist bei der Entwicklung elektronischer Märkte, die dann im Internet weltweit

verfügbar sind, oft besonders groß. Man fürchtet, durch unzureichende Lösungen Kunden zu verschrecken und die eigene Position am Markt zu schwächen.

Dem ist entgegenzuhalten, dass das Streben nach ohnehin nie realisierbaren, perfekten Lösungen nicht nur wertvolle Zeit und viel Geld, sondern auch die Chance kostet, die Akzeptanz der Kunden zu testen, die letztlich alleine über den Erfolg der Lösung entscheiden, bevor die Entwicklung am Markt vorbeiläuft. Wird dagegen die Interaktivität der neuen Kommunikationsmedien konsequent genutzt, um Kunden in die Bewertung und Gestaltung von Informationsangeboten und sogar in Unternehmensprozesse einzubeziehen, eröffnet dies u. U. die Chance zur fundamentalen Neuorientierung des Unternehmens [SIEG99, S. 81-101].

Aus technischer Sicht bedeutet dies, dass ein Adaptionswerkzeug zu jedem Zeitpunkt den Entwurf und die Realisierung einer stabilen, technisch funktionsfähigen und ökonomisch sinnvollen Lösung sicherstellen muss. Ein kontinuierlicher Verbesserungsprozess im Sinne der Evaluierung gemachter Erfahrungen ist nur möglich, wenn alle Realisierungsschritte ausreichend dokumentiert werden.

4.3 ODYSSEUS als Leitfaden zur Adaption elektronischer Märkte

Das ODYSSEUS-Konzept definiert klare Anforderungen bezüglich der Anwendung der Adaptionarten und der einzusetzenden Werkzeuge. Trotz offensichtlicher Vorteile dieses Konzepts bleibt zu klären, inwieweit es zur Lösung der speziellen Probleme bei der Entwicklung elektronischer Märkte geeignet ist.

4.3.1 Bezugsrahmen

Das ODYSSEUS-Konzept wurde entwickelt, um eine Standardanwendungssoftwarebibliothek zur Unterstützung innerbetrieblicher Abläufe einzuführen. Folglich wird dort mit der Auswahl der betroffenen Fachbereiche und Funktionen begonnen, für die nach Maßgabe der Softwarebibliothek eine Lösung entwickelt wird (siehe Abbildung 4-2). Den Bezugsrahmen für die Implementierung des Zielsystems definiert eine

existierende Softwarebibliothek, die im Fall von SAP/R3 als Maßstab für die korrekte Umsetzung betriebswirtschaftlicher Anforderungen angesehen wird [THOM92].

Auf elektronischen Märkten werden dagegen völlig neue Lösungen benötigt, aus denen der jeweilige Anbieter klare Wettbewerbsvorteile gewinnt. In den meisten Unternehmen, die sich erstmalig mit der Einführung eines elektronischen Marktes beschäftigen, existiert noch keine detaillierte Vorstellung über die eigentlichen Potentiale, Projektziele oder gar über die Anforderungen an das System. Der Aufbau eines elektronischen Marktes bedeutet neben neuartiger technologischer Anforderungen vor allem den Eintritt in ein neues Marktsegment mit weiteren Konkurrenten und oft unbekanntem Regeln.

Erste Aufgabe bei der Einführung eines elektronischen Marktes ist es, unter Berücksichtigung der individuellen Chancen und Risiken des Unternehmens auf elektronischen Märkten ein Konzept zu erstellen, das Ziele definiert, den Aufbau des Systems beschreibt und die Maßnahmen zur organisatorischen und technischen Umsetzung dieser Ziele definiert. Erst aus diesem Konzept wird ersichtlich, welche Fachbereiche und Funktionen von der Einführung des Marktsystems betroffen sind. Die Auswahl der Funktionen und Fachbereiche ist deshalb nicht Ausgangspunkt, sondern wichtiges Teilziel der Einführung.

Daher ist die Frage zu beantworten, ob ein Bezugsrahmen zur Ableitung eines Konzepts für einen elektronischen Markt überhaupt existiert. Die Betrachtung der ökonomischen Effekte in Kapitel 2 zeigt, dass auf elektronischen Märkten elementare ökonomische Gesetzmäßigkeiten gelten, die unter bestimmten Voraussetzungen zum Vorteil eines Unternehmens genutzt werden können und aus denen sich erkennbare Konsequenzen ergeben. Die Rahmenbedingungen für die Entfaltungsmöglichkeiten ökonomischer Vorteile werden einerseits durch die Marktsituation (Handelsmittlereffekte, Kommunikationseffekt, Netzeffekte) und andererseits durch die vorhandenen Geschäftsprozesse und Ressourcen des Unternehmens bestimmt. Die Ableitung eines tragfähigen Konzepts kann folglich nur durch eine Gesamtbetrachtung der Unternehmenssituation innerhalb dieses Bezugsrahmens geschehen. Es gilt, alle entscheidungsrelevanten Informationen zu erfassen und aus diesen Informationen Ziele abzuleiten, die die Chancen, aber auch die Risiken elektronischer Märkte erfassen. Aus diesen Zielen ergeben sich unmittelbar die organisatorischen Maßnahmen und auch die Auswahl der Funktionen aus der Softwarebibliothek, die zur Umsetzung benötigt

werden. Ziele, organisatorische Maßnahmen und die Funktionsauswahl beschreiben einen ersten konzeptionellen Entwurf, der zur Überprüfung und als Ausgangspunkt für spätere Anpassungen dokumentiert werden muss. Ergebnis der Konzeption ist eine Funktionsauswahl, die den Einführungsaufwand in den nachfolgenden Phasen reduziert.

4.3.2 Freiheitsgrade bei der Anpassung

Im ODYSSEUS-Konzept wird die Modifikation der Standardsoftware auf die Auswahl und Anpassung im vordefinierten Rahmen beschränkt. Das schließt jede Art der Modifikation durch Programmierung an der Standardsoftware selbst aus. Ergänzungsentwicklungen spielen im ODYSSEUS-Konzept eine nachgeordnete Rolle. Das wirft die Frage auf, ob dieses Konzept die erforderlichen Freiheitsgrade bietet, um unternehmensspezifische Eigenschaften auszuspielen und sich erfolgreich von Wettbewerbern zu differenzieren. Die Differenzierung erfolgt auf elektronischen Märkten neben dem Leistungsangebot über die im Marktsystem abgebildeten Prozesse und das Design der Benutzerschnittstelle. Es ist folglich zu untersuchen, ob das ODYSSEUS-Konzept hier ausreichend Differenzierungsmöglichkeiten bietet.

Diese Frage lässt sich durch eine einfache mathematische Abschätzung der nach dem ODYSSEUS-Konzept grundsätzlich möglichen Adaptionvarianten klären. Erster Teilschritt der Adaption im ODYSSEUS-Konzept ist aus softwaretechnischer Sicht die Auswahl der benötigten Softwarekomponenten aus dem Lösungsvorrat der Standardanwendungssoftwarebibliothek. Im Gegensatz zu der Anpassung einer templatebasierten Standardanwendungssoftware oder eines netzbasierten Assistenten (siehe 3.3.2.3) wird die Komponentenauswahl nicht von vornherein durch ein vorgegebenes Modell beschränkt. Mathematisch betrachtet lässt sich die Auswahl von Softwarekomponenten anhand des aus der Wahrscheinlichkeitsrechnung bekannten Urnenmodells adäquat als *Ziehen mit Zurücklegen* beschreiben. Diese Beschreibung der Problemstellung berücksichtigt, dass jedes Softwareelement innerhalb einer Anwendung auch mehrfach eingesetzt werden kann.

Die Zahl der unterschiedlichen Tupel bei Ziehen mit Zurücklegen beträgt [BASL89, S. 22]:

$$(A1) \quad N^K .$$

Das entspricht der durch Auswahl möglichen verschiedenartigen Kombinationen von Softwareelementen, wobei N die Zahl der in der Softwarebibliothek vorhandenen und K die Zahl der für den konkreten Anwendungsfall ausgewählten Elemente bezeichnet.

Ihre individuelle Konfiguration erhalten die ausgewählten Softwareelemente in der zweiten Adaptionphase durch Zuordnung eines betriebswirtschaftlichen Profils. Die Profizuordnung geschieht durch Parametrisierung der Softwarekomponenten. Im gewählten Modell bedeutet dies, dass jedes Element N aus (A1) jeweils so viele Ausprägungen annehmen kann, wie Profile für dieses Element existieren. Geht man vereinfachend davon aus, dass allen Elementen dieselbe Zahl an Profilen zugeordnet werden kann, lässt sich die Zahl der durch Auswahl und Zuordnung gegebenen Adaptionmöglichkeiten durch den Ausdruck (A2) beschreiben:

$$(A2) \quad (N \cdot J)^K, \quad \text{wobei } J \text{ die Anzahl der verfügbaren Profile bezeichnet.}$$

Würden aus einer gedachten Softwarebibliothek mit nur 25 Elementen und einer durchschnittlichen Profilzahl von 3 nur 7 Elemente ausgewählt, ergäben sich schon in diesem vereinfachten Beispiel

$$(25 \cdot 3)^7 = 13.348.388.671.875$$

verschiedene Möglichkeiten der Adaption.

Für die Abbildung realer betriebswirtschaftlicher Problemstellungen ist der Umfang dieser stark vereinfachten Bibliothek bei weitem noch nicht ausreichend. Das Beispiel zeigt aber, wie viele Adaptionmöglichkeiten alleine durch Auswahl und Anpassung von Softwareelementen möglich sind. Mit der Anzahl der Softwareelemente und der Profile wächst diese Zahl exponentiell.

Die Zuordnung der betriebswirtschaftlichen Profile geschieht im ODYSSEUS-Konzept durch Parameterdefinition. Um regelbasiert abgeleitet werden zu können, müssen Parameter lediglich in ihrer Bedeutung und ihrem Wertebereich, nicht aber in ihren möglichen Ausprägungen eindeutig festgelegt werden [MERT91]. Das ODYSSEUS-Konzept sieht auch Parametereingaben durch den Benutzer vor. Die

Anzahl der möglichen Profile ist folglich aus technischer Sicht unbegrenzt, was theoretisch zu einer unendlich großen Vielfalt an Adaptionmöglichkeiten führt.

Natürlich sind nicht alle technisch realisierbaren Lösungen auch betriebswirtschaftlich sinnvoll oder gar für das betroffene Unternehmen geeignet. Die Auswahl und Konfiguration einer bestmöglichen Lösung ist Aufgabe des regelbasierten Adaptionswerkzeuges, das den Anwender schrittweise zu einer ökonomisch sinnvollen Lösung führt, die sich primär an den speziellen ökonomischen Rahmenbedingungen seines Unternehmens orientiert.

4.3.2.1 Modellierungsunterstützung bei der Anpassung von Prozessen

Die Auswahl und Anpassung der Softwareelemente definieren lediglich den funktionalen Rahmen des Zielsystems. Entscheidend ist, wie die ausgewählten Elemente zu Prozessen zusammengefügt werden. Ein effizienter Prozessentwurf ist nur durch geeignete Modellierungsunterstützung möglich.

Das ursprünglich nach dem ODYSSEUS-Konzept entwickelte Werkzeug LIVE KIT Structure unterstützt hauptsächlich die Analyse der Struktur des untersuchten Unternehmens und gibt konkrete Hinweise zu ihrer Abbildung im SAP/R3-System. Der Entwurf von Prozessen wird nicht unmittelbar unterstützt. Aus diesem Grund wurde das ODYSSEUS-Konzept um das von VOGELSANG beschriebene PENELOPE-Konzept (**Prozess-Ebenen-Analyse für Ergänzungsentwicklung, Lückenidentifikation und organisatorische Problemlösung**) erweitert [VOGE98]. Es setzt bei der Vielzahl prozessrelevanter Informationen an, die bei jedem Einsatz des LIVE KIT anfallen. So geht aus der Beschreibung der Organisationsstruktur nicht nur hervor, welche Elemente die Organisation enthält. Die Auswahl und Beschreibung der Funktionen lassen auch Beziehungen erkennen, die zwischen diesen Elementen bestehen. Der Vorschlag zur Abbildung der Organisationsstruktur kann als Grundlage für die Prozessgestaltung angesehen werden, denn sie beschreibt nicht nur die Organisationselemente und ihre Beziehungen untereinander, sondern sie liefert auch Informationen über zwingend notwendige organisatorische Zusammenhänge und damit über die Freiheitsgrade bei der Modellierung individueller Prozesse. Das LIVE KIT Structure nimmt eine sehr genaue Analyse dieser Zusammenhänge vor.

Das nach dem PENELOPE-Konzept realisierte Werkzeug stellt prozessrelevante Informationen in mehreren Sichten und auf sechs unterschiedlichen Detaillierungsebenen grafisch und in textueller Form dar. Die grafische Darstellung erfolgt in Form aktiver Referenzmodelle, die unmittelbar und ohne Modellierungskennnisse modifiziert und an kundenindividuelle Anforderungen angepasst werden können. PENELOPE ist mit der anzupassenden Softwarebibliothek eng über ein Fundamentalmodell verknüpft, das auf der einen Seite ein Abbild der Softwarebibliothek darstellt und auf der anderen Seite die Ergebnisse des LIVE KIT Strukturre Workshops aufgreift. Durch die Verknüpfung adaptionsrelevanter Informationen über die Softwarebibliothek mit prozessrelevanten Informationen aus der werkzeuggestützten Anforderungsanalyse wird eine kundenindividuelle Prozessmodellierung möglich. Ferner können mit PENELOPE sowohl das Lösungspotential der Softwarebibliothek als auch Lücken in der Abdeckung von Kundenanforderungen in Bezug auf einzelne Prozesse sichtbar gemacht und daraus Hinweise auf die Identifikation von Lücken abgeleitet werden.

Das PENELOPE-Konzept und seine Umsetzung in ein in der Praxis angewandtes Werkzeug liefert den Beleg dafür, dass die Prozessmodellierung bei der Adaption von Standardanwendungssoftware durch Werkzeuge, die ODYSSEUS ergänzen und direkt auf den von ihm gewonnenen Informationen aufsetzen, unterstützt werden kann und mit Hilfe grafischer, vollständig integrierter Modellierungswerkzeuge wichtige Informationen für nachfolgende Adaptionsphasen zu gewinnen sind. Das PENELOPE-Konzept kann folglich als Grundlage für die Konzeption eines Werkzeugs zur Prozessmodellierung herangezogen werden (siehe 6.2).

4.3.2.2 Gestaltungsfreiheit beim Design der Benutzerschnittstelle

Viele Unternehmen stellen sehr hohe Anforderungen bezüglich der Umsetzung ihrer Corporate Design Richtlinien in ihren Online-Anwendungen und investieren einen beachtlichen Teil des Budgets in die Entwicklung werbewirksamer Designelemente, um die Aufmerksamkeit der Internet-Nutzer auf sich zu ziehen. Es stellt sich die Frage, ob das ODYSSEUS-Konzept ausreichende Freiheitsgrade zum Design elektronischer Märkte bietet.

Die Umsetzung individueller Anforderungen durch Adaption wird im ODYSSEUS-Konzept auf Auswahl und Anpassung im vordefinierten Rahmen beschränkt. Die Auswahl geeigneter Elemente aus der Softwarebibliothek richtet sich vorrangig nach den funktionalen Anforderungen an das Zielsystem. Aus designtechnischen Gründen mehrere funktional identische Elemente in der Softwarebibliothek zur Auswahl vorzuhalten, ist kaum sinnvoll. Folglich macht für das einzelne Element nur eine standardisierte Abbildung von Designelementen Sinn, die im vordefinierten Rahmen anpassbar ist. Der HTML-Standard setzt ohnehin dem Design von Benutzungsoberflächen enge technische Grenzen. Durch HTML lässt sich nur die Struktur einer Web-Seite beschreiben. Im Rahmen der Strukturbeschreibung ist der Designer auf die Strukturelemente festgelegt, die im HTML-Standard definiert sind bzw. von den verwendeten Browsern interpretiert werden können. Die Anzahl der verfügbaren Auszeichnungsanweisungen ist selbst in der aktuellen Version des HTML-Standards noch auf ca. 60 beschränkt [WERB02]. Diese Anweisungen besitzen meist vorgegebene Attribute, über die unterschiedliche Ausprägungen gewählt werden können. Die gesamte Seitenbeschreibung wird als Zeichenfolge von Auszeichnungsanweisungen und darzustellenden Inhalten übermittelt. Die Werte einzelner Attribute und sogar beliebig lange Zeichenfolgen lassen sich nach den Anforderungen des ODYSSEUS-Konzepts als Parameter definieren und ablegen. Eine zentrale Aufgabenstellung bei der Realisierung einer adaptionsfähigen Softwarebibliothek für elektronische Märkte ist folglich der Entwurf einer Datenstruktur, in der alle Parameter, die das Design eines elektronischen Marktes beschreiben, effizient verwaltet werden können.

4.3.2.3 Erweiterbarkeit von Lösungen

Nach dem Subsidiaritätsprinzip spielt Ergänzungsentwicklung im ODYSSEUS-Konzept nur eine untergeordnete Rolle. Gefordert wird eine nahezu vollständig werkzeugunterstützte Adaption der Standardanwendungssoftware. So sind die Möglichkeiten bei der Realisierung verschiedener Anforderungen nicht allein technisch durch den Funktionsumfang der Standardanwendungssoftware vorgegeben, sondern auch durch die verfügbaren Adaptionswerkzeuge.

Das LIVE KIT Structure orientiert sich an der Softwarebibliothek SAP/R3, die alle Anforderungen der klassischen Betriebswirtschaftslehre in weiten Bereichen nahezu

vollständig abdeckt. Die Aufgabe bei der Adaption von SAP/R3 besteht vor allem darin, die vorhandenen Realisierungsmöglichkeiten systematisch zu erfassen. Für elektronische Märkte existiert dagegen bisher keine entsprechende Standardanwendungssoftwarebibliothek. Dies ist nicht nur auf die Dynamik der technischen Entwicklung, sondern vor allem darauf zurückzuführen, dass fortwährend neue betriebswirtschaftliche Konzepte und Ansätze zur Nutzung elektronischer Märkte entstehen.

Aus dieser Situation ergeben sich verschiedene Anforderungen, die eine Standardanwendungssoftware für elektronische Märkte und die zu ihrer Adaption eingesetzten Werkzeuge erfüllen müssen, um längerfristig sinnvoll eingesetzt werden zu können. Zum Einen muss eine Erweiterung des Funktionsumfangs durch Entwicklung neuer Standardkomponenten durch anbieterspezifische Ergänzungsentwicklung jederzeit und ohne umfangreiche Anpassung der Softwarebibliothek möglich sein. Dies schließt auch die Erweiterung des Funktionsumfangs um Fremdprodukte ein. Zum Anderen müssen alle zur Adaption eingesetzten Werkzeuge mit geringem Aufwand anpassbar sein, um jede Änderung des Funktionsumfangs nutzen zu können. Diese Anforderungen sind nur zu erfüllen durch eine offene Softwarebibliothek mit klar definierten Schnittstellen (siehe 5.4) und ein auf Weiterentwicklung angelegtes, betriebswirtschaftliches Regelwerk hinter dem Anforderungsnavigator (siehe 6.1.2).

5 Entwurf der Standardanwendungssoftwarebibliothek

Nach der Einführung der methodischen und technischen Grundlagen der Adaption wird in diesem Kapitel eine Standardanwendungssoftware konzipiert, die die Anforderungen an Werkzeuge zur Entwicklung elektronischer Märkte erfüllt und die Adaption nach dem oben beschriebenen ODYSSEUS-Konzept möglich macht.

Der vorgeschlagene Lösungsansatz zur Realisierung der Standardanwendungssoftwarebibliothek basiert auf dem Paradigma der komponentenbasierten Softwareentwicklung. Die in 3.2.2 ausführlich dargestellten Methoden und Prinzipien komponentenbasierter Softwareentwicklung werden in 5.2 zum Entwurf der Systemarchitektur einer adaptionsfähigen Standardanwendungssoftwarebibliothek genutzt. Als Voraussetzung für die dynamische Adaptionfähigkeit der Standardanwendungssoftwarebibliothek ist ein Datenmodell zu entwickeln, das die vollständige Abbildung eines Marktsystems in Form von Parametern ermöglicht. Dazu wird in 5.3 ein geeignetes relationales Datenmodell entwickelt. Abschließend werden in 5.4 Möglichkeiten zur Erweiterung der Softwarebibliothek erläutert.

5.1 Formulierung der Anforderungen

Die grundsätzlichen Anforderungen an eine Standardanwendungssoftwarebibliothek sind in [THOM96] bereits formuliert. Danach muss eine Standardanwendungssoftwarebibliothek die folgenden sieben Kriterien erfüllen [THOM96, S. 45]:

- (I) Plattformübergreifende Entwicklung.
- (II) Funktionsumfang mit einem Abdeckungsgrad von mindestens 80 % der in einem beliebigen Unternehmen benötigten betriebswirtschaftlichen Funktionen.
- (III) Der Zugriff aller Module muss koordiniert über eine gemeinsame Datenbank erfolgen.
- (IV) Anpassbarkeit der einzelnen Programmmodule an die unterschiedlichen Ausgestaltungsformen betriebswirtschaftlicher Lösungen durch Parametereinstellung oder andere Verfahren der Adaption.
- (V) Dynamische Adaptionfähigkeit durch Änderung von Parametereinstellungen unter Konsistenzerhaltung des Anwendungssystems.
- (VI) Dynamische Austauschbarkeit einzelner Module.
- (VII) Konsistente Verarbeitung aller zu verwaltenden Daten auch bei asynchroner Anpassung bzw. Weiterentwicklung der Module.

Soweit es eine Softwarebibliothek für elektronische Märkte betrifft, schließt die Forderung nach der Anpassbarkeit der Module an unterschiedliche Ausgestaltungsformen zwingend auch die Forderung nach hoher Flexibilität bei der Prozessanpassung und der Umsetzung individueller Designvorstellungen ein.

Das Existenzkriterium (IV) enthält implizit auch die Forderung nach Erweiterbarkeit des Systems. Die Erweiterung der Softwarebibliothek um neue Funktionen und Elemente sowie die Verwendung von Fremdsoftware im Zusammenspiel mit der Softwarebibliothek können als Sonderformen der Ergänzungsentwicklung angesehen werden.

5.2 Systemarchitektur der Softwarebibliothek

Einige der Anforderungen an Standardanwendungssoftwarebibliotheken sind bei einem komponentenbasierten Aufbau der Systemarchitektur bereits unmittelbar erfüllt. So ist die Austauschbarkeit der Komponenten und die Möglichkeit zur Integration von Fremdsoftware in komponentenbasierten Softwaresystemen prinzipiell gegeben und hängt im Wesentlichen von der Implementierung der Schnittstellen sowie den verwendeten Kollaborationsmechanismen ab.

Auch die Adaption einzelner Komponenten durch Einstellen von Parameterwerten ist mit komponentenbasierten Softwareentwicklungsumgebungen prinzipiell möglich. Im Gegensatz zu den marktgängigen Montageumgebungen für komponentenbasierte Software, die lediglich das Zusammensetzen vorhandener Bausteine zu festgefügt Systemen unterstützen, wird hier eine Lösung benötigt, die die dynamische Adaption der Softwarebibliothek ohne Unterbrechung des laufenden Systembetriebes ermöglicht.

Diese Anforderung wird durch den nachfolgend beschriebenen Lösungsansatz erfüllt. Er basiert auf einem Komponentenframework adaptionsfähiger Softwarekomponenten. Als **Komponentenframework** wird eine Systemarchitektur bezeichnet, die eine Bibliothek von Komponenten enthält und Kooperationsformen für den Austausch von Mitteilungen zwischen Komponenten vorgibt [FRAN99, S. 27; NIER97, S. 11].

Unter einer **adaptionsfähigen Softwarekomponente** wird in diesem Zusammenhang ein Softwareelement verstanden, das unabhängig von allen anderen Systemteilen einen abgeschlossenen Teilprozess selbstständig ausführt, die Ergebnisse darstellt und über Parametereinstellungen in ihrer Funktionalität, der Ergebnisdarstellung und den Schnittstellen im laufenden Betrieb angepasst werden kann. Die Adaptionsparameter müssen zur persistenten Speicherung an ein System übergeben werden, das diese allen Adaptionswerkzeugen bereitstellt.

Im Komponentenframework wird als System zur Speicherung und Verwaltung der Adaptionsparameter ein relationales Datenbanksystem (RDBMS) eingesetzt. In jeweils getrennten Bereichen werden im RDBMS sowohl die im laufenden Betrieb an-

fallenden Geschäftsdaten als auch die Parameterdaten, welche die Funktion und das Layout des Systems bestimmen, persistent gespeichert. So kann die Forderung nach einem koordinierten Zugriff aller Komponenten auf eine gemeinsame Datenbasis erfüllt werden. Gleichzeitig wird durch die Abbildung aller Parameterinformationen in einem unabhängigen System, auf das über eine standardisierte Abfragesprache zugegriffen werden kann, eine hohe Flexibilität und Offenheit gegenüber der Integration verschiedenster Adaptionswerkzeuge sichergestellt.

Abbildung 5-1 zeigt die Architektur der Standardanwendungssoftware in UML-Notation. Der Umfang des Frameworks wird in UML durch ein Package begrenzt, Schnittstellen werden als kreisförmige Symbole dargestellt [OMG01].

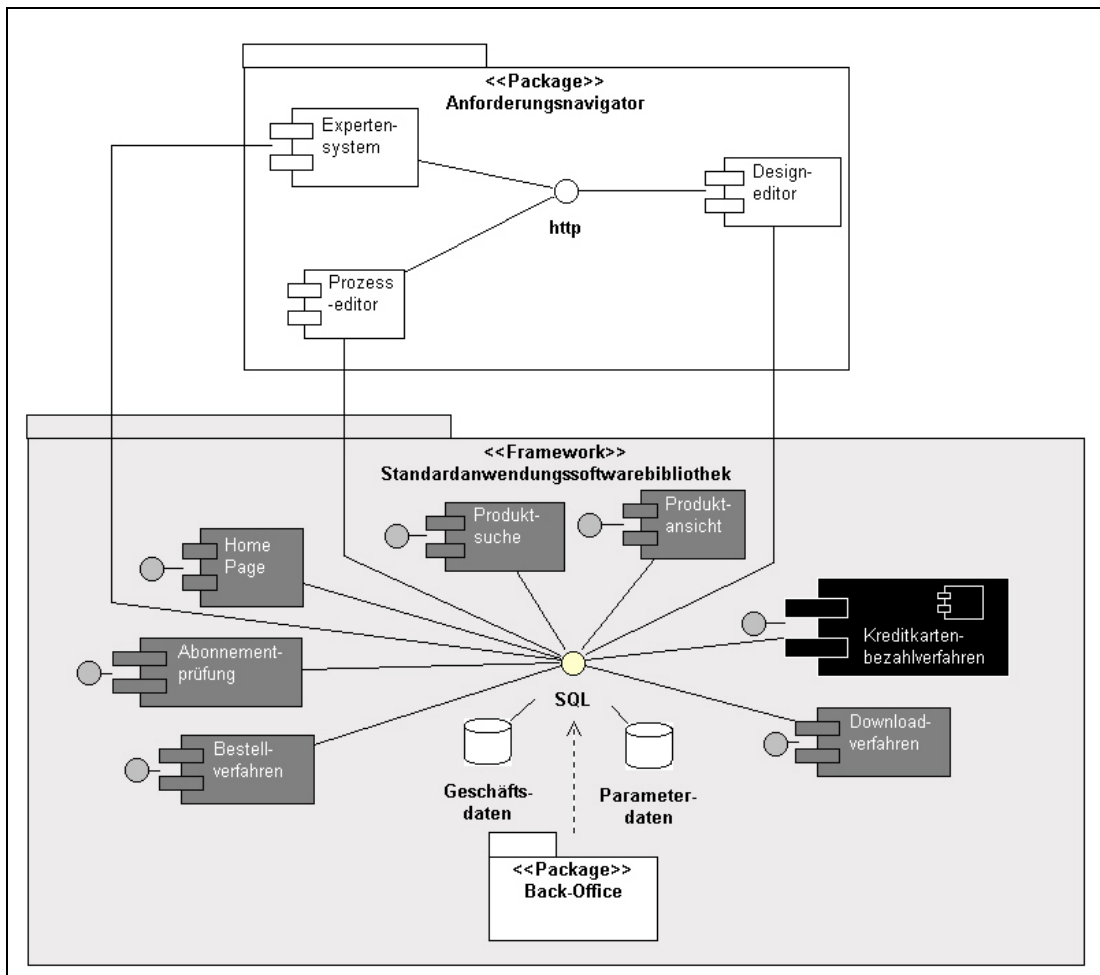


Abbildung 5-1: Architektur des Komponentenframeworks

Statische Referenzen auf Schnittstellen sind als Linie dargestellt. Dynamische Referenzen sind dagegen nicht dargestellt, denn diese werden erst in der Adaption für

einen konkreten Anwendungsfall erzeugt. Um die Integration der Softwarebibliothek mit den verschiedenen Adaptionswerkzeugen zu verdeutlichen, sind die wichtigsten Werkzeuge des Anforderungsnavigators mit ihren Schnittstellen zur Softwarebibliothek ebenfalls dargestellt.

Innerhalb des Frameworks ist eine Auswahl von Komponenten sichtbar, die wesentliche Funktionen in einem elektronischen Vertriebsprozess ausführen. Die Komponente zur Abwicklung des Kreditkartenbezahlverfahrens ist als Fremdkomponente dargestellt, denn zur Abwicklung von Kreditkartenbezahlverfahren existiert eine Reihe vorgefertigter Lösungen. Beispielsweise wurde im Rahmen des IntelliShop-Projektes zu Forschungszwecken die Softwarelösung zur Nutzung des elektronischen Bezahlverfahrens der Firma CyberCash in das Framework integriert. Fremdbezogene Softwarekomponenten müssen durch Schnittstellenimplementierung in das Framework integriert werden (siehe 5.4).

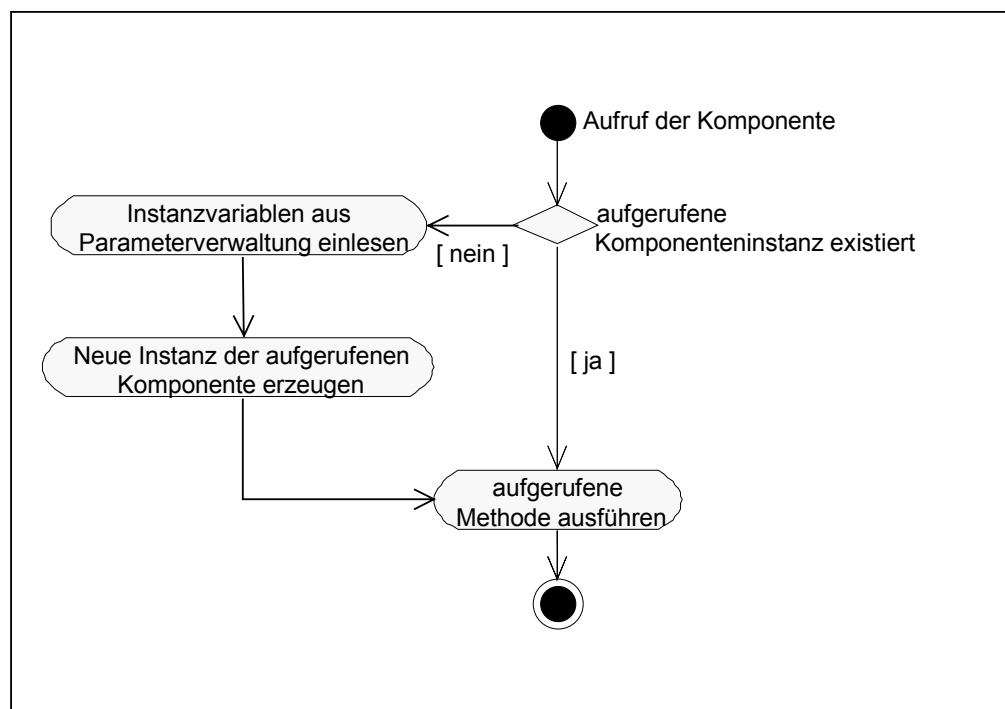


Abbildung 5-2: Aufruf einer Komponente

Alle Komponenten besitzen jeweils zwei Schnittstellen. Jede Komponente des Frameworks verfügt über eine direkte Schnittstelle zum relationalen Datenbanksystem, in dem u. a. die Informationen zur Parametrisierung für ihre jeweiligen An-

wendungsfälle verwaltet werden (siehe 5.3.2). Über die zweite Schnittstelle erfolgt die Kommunikation zwischen den Komponenten untereinander. Sie ist als indirekte Schnittstelle implementiert. Über diese Kommunikationsschnittstelle können alle in der Parameterverwaltung angegebenen Dienste einer beliebigen Komponente angefordert und Ergebnisdaten weitergeleitet werden.

Erfolgt der Aufruf einer Komponente zur Laufzeit, werden zunächst alle Parameterinformationen gelesen, die zuvor im Adaptionprozess für den angeforderten Anwendungsfall definiert wurden. Anschließend wird geprüft, ob durch einen bereits erfolgten Aufruf schon eine Ausprägung der Komponente für diesen Anwendungsfall vorhanden ist. Trifft dies nicht zu, wird eine neue Instanz der aufgerufenen Komponente erzeugt, die gelesenen Parameterdaten den Instanzvariablen zugewiesen und die geforderte Methode parametrisiert ausgeführt (siehe Abbildung 5-2).

So kann die gleiche Softwarekomponente in beliebig parametrisierter Form und an ganz unterschiedlichen Stellen in verschiedenen Geschäftsprozessen verwendet werden. Abbildung 5-3 und Abbildung 5-4 zeigen Beispiele, wie mit den Komponenten der Standardanwendungssoftwarebibliothek zwei unterschiedliche Aufgabenstellungen im elektronischen Vertrieb abgebildet werden können.

Zur Abwicklung von Bestellungen nicht-digitaler Güter genügt eine Lösung, bei der vor der Übermittlung der Bestelldaten keine Authentifizierung des Bestellers notwendig ist, da diese im Rahmen des Bezahlverfahrens erfolgen kann.

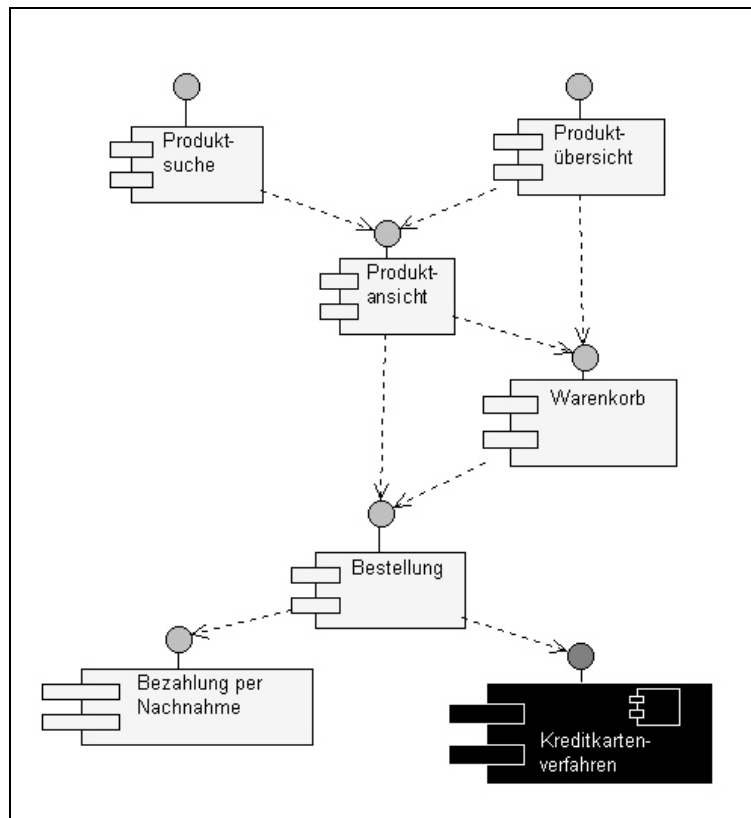


Abbildung 5-3: Elektronischer Vertrieb nicht-digitaler Güter

Beim elektronischen Vertrieb digitaler Güter wird das bestellte Gut ohne die Möglichkeit der Authentifizierung durch Dritte direkt an den Kunden übermittelt. Dazu muss der Kunde in jedem Fall vor der Übermittlung der Leistung sicher authentifiziert und das entsprechende Bezahlverfahren aktiviert werden. Anschließend muss ein Verfahren zur Verfügung gestellt werden, das auch im Falle technischer Probleme bei der Datenübertragung die sichere Übermittlung einer digitalen Kopie des gewählten Produkts an den Kunden gewährleistet.

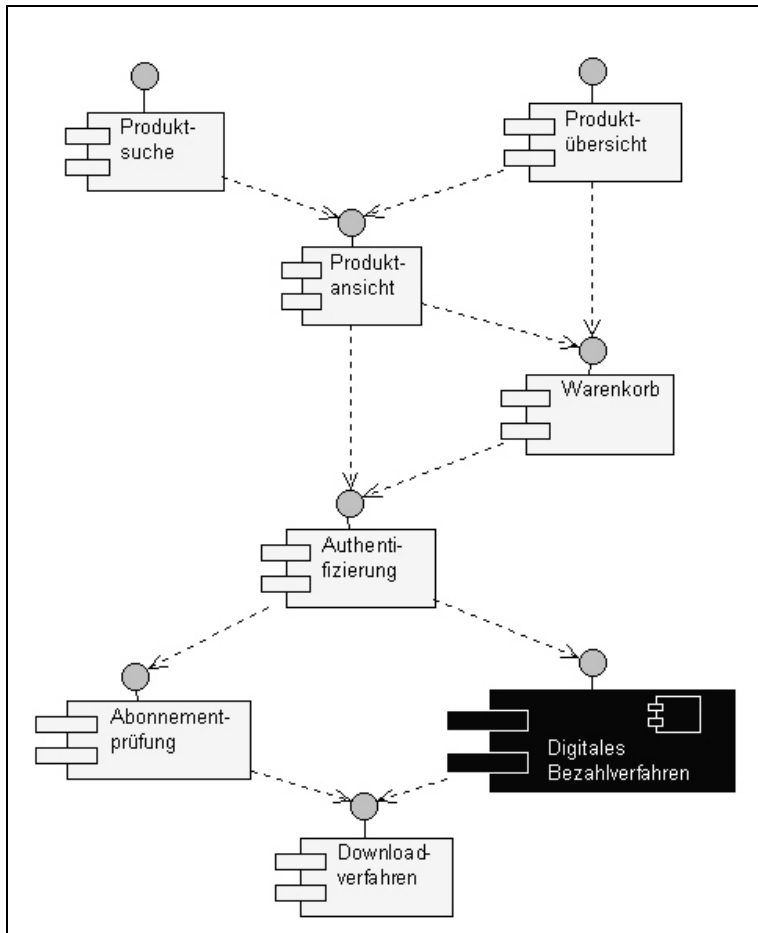


Abbildung 5-4: Elektronischer Vertrieb digitaler Güter

Diese flexible Verwendbarkeit und Adaptionfähigkeit der Komponenten wird u. a. dadurch erreicht, dass alle Informationen bezüglich Funktionalität, Design und der Schnittstellen aller Komponenten in der Parameterdatenbank erfasst und verwaltet werden, wo sie jederzeit dynamisch verändert werden können.

5.3 Datenbankbasierte Anwendungsrepräsentation

Die Abschätzung der Kombinationsmöglichkeiten bei der Adaption eines Systems in Abschnitt 4.3.2 zeigt, wie viele Möglichkeiten der Adaption durch Auswahl von Softwarekomponenten und Profilverordnung durch Parametrisierung in einer Standardanwendungssoftware theoretisch gegeben sind.

Um Komponenten flexibel in Prozessketten einsetzen zu können, müssen neben den funktionalen Alternativen auch die entsprechenden Informationen über Methoden- bzw. Funktionsaufrufe parametrisierbar gespeichert werden. Für die Abbildung eines individuellen Designs der Benutzungsoberflächen eines elektronischen Marktes muss zudem eine Vielzahl weiterer Parameter definiert werden. Um alle diese Parameter effizient in einem Datenbanksystem zu verwalten, wird für die Standardanwendungssoftware ein Datenmodell benötigt, das folgende Anforderungen erfüllt:

- Das Modell muss eine hohe Allgemeingültigkeit besitzen, sodass eine große Zahl unterschiedlicher Ausprägungsformen elektronischer Märkte abgebildet werden kann.
- Freiheitsgrade bei der Prozessmodellierung und dem Design individueller Benutzungsoberflächen dürfen durch das Datenmodell nicht eingeschränkt werden.
- Die Integration von Adaptionswerkzeugen und die Parametrisierung umfangreicher Anwendungen müssen durch den Aufbau des Modells effizient unterstützt werden.

5.3.1 Hierarchische Parameterstruktur

Abbildung 5-5 zeigt ein Entity-Relationship-Modell (ER-Modell) der Parameterverwaltung in IDEF1X-Notation [NIST93]. Einzelne Entitäten der Parameterverwaltung (z. B. die Entitäten *Anwendung*, *Bereich*) stehen mit Entitäten der Geschäftsdatenverwaltung in direkten Beziehungen.

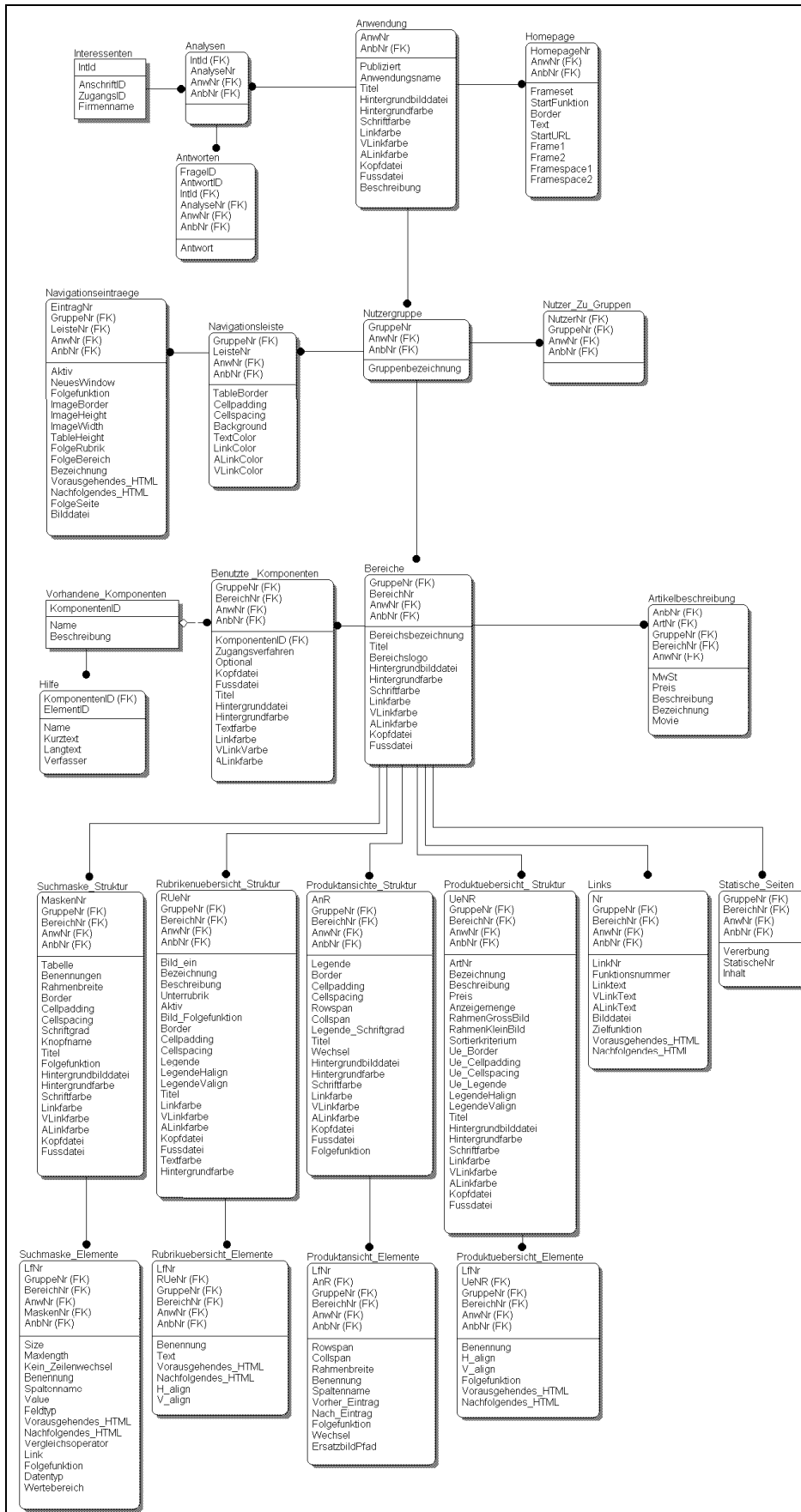


Abbildung 5-5: Entity-Relationship-Modell der Parameterverwaltung

Im Datenmodell wird ein elektronischer Markt als hierarchische Struktur mit mehreren Abstraktionsstufen abgebildet. Jede Hierarchieebene wird durch eine oder mehrere Entitäten repräsentiert. Die hierarchische Beziehung zwischen den Entitäten der Stufen wird im ER-Modell durch 1:N-Abhängigkeiten zwischen einer übergeordneten und ihren untergeordneten Entitäten ausgedrückt. Jeder übergeordneten Entität können so beliebig viele der nächsten Stufe zugeordnet werden. Die untergeordneten Entitäten erhalten dabei jeweils die Primärschlüssel der übergeordneten Entität als Fremdschlüsselattribute.

Die erste Abstraktionsstufe ist die parametrisierte Abbildung der Anwendung selbst. Die Attribute der zugehörigen Entität legen Eigenschaften fest, die für die gesamte Anwendung und folglich auch für alle untergeordneten Abstraktionsebenen gelten. Durch Eingabe der gewünschten Werte in die entsprechende Tabelle können sehr einfach die Designrichtlinien für die gesamte Anwendung festgelegt werden. Der Entität *Anwendung* zugeordnet sind die Stammdaten des Anbieters und die Entitäten *Navigationsleiste* und *Navigationsleisteneinträge*, die alle Parameter der Navigationsleiste einer Anwendung aufnehmen, sowie die Fakten und Ergebnisse aus der Analyse und Konzeption (siehe 6.1), die zu ihrer Entwicklung geführt haben. In der Gesamtsicht des Datenmodells wird über eine N:1-Beziehung zu der Entität *Anwendung* die Verknüpfung zu den Datenstrukturen der Anbieterdaten hergestellt. So ist jeder Anbieter, der die Standardanwendungssoftwarebibliothek nutzt, in der Lage, beliebig viele Anwendungen zu realisieren.

Auf den nächsten beiden Stufen wird die Anwendung durch die Entitäten *Nutzergruppen* und *Bereiche* näher beschrieben. Durch die Differenzierung einer Anwendung in Nutzergruppen können alle nachgeordneten Inhalte und auch Funktionen nutzergruppenspezifisch angepasst werden. Diese beiden Hierarchiestufen können sehr vielseitig eingesetzt werden, um das Angebot bestimmter Produkte und Dienstleistungen kundengruppenspezifisch zu gestalten. So können z. B. mehrsprachige Anwendungen sehr einfach abgebildet werden.

Über die Abstraktionsstufe *Bereiche* kann das gesamte Angebot in Produktbereiche unterteilt werden. Die Definition einer klaren Angebotsstruktur ist eine wichtige Orientierungshilfe für den Benutzer. Typische Strukturierungsansätze sind die Bildung von Produktgruppen nach Produkt- oder Angebotseigenschaften (z. B. Sonderangebote, Aktionsmodelle). Um die Präsentationsform jedes Artikels bereichsspezifisch

festlegen zu können, ist die Entität *Bereiche* über eine 1:1-Beziehung der Entität *Artikelbeschreibung* zugeordnet. Über diese Entität erfolgt die Verknüpfung zur Artikelverwaltung im Gesamtmodell. In Folge der hierarchischen Anordnung der *Bereiche* unterhalb der *Nutzergruppen* kann die Bereichsdefinition wiederum nutzergruppenspezifisch vorgenommen werden.

In direkter Beziehung mit der Entität *Bereiche* stehen die Entitäten *Vorhandene_Komponenten* und *Benutzte_Komponenten*. Diese Entitäten bilden Hilfsrelationen ab, die Informationen für verschiedene Adaptionswerkzeuge enthalten. Die Entität *Vorhandene_Komponenten* ist ein Verzeichnis aller im Framework integrierten Komponenten. Die Entität *Benutzte_Komponenten* ist ein Verzeichnis aller Softwareelemente, die in einem Anwendungsbereich eingesetzt werden. Dieses Verzeichnis wird aufgrund der Komponentenauswahl aus der wissensbasierten Analyse gefüllt und im Prozesseditor genutzt, um die Modellierungsmöglichkeiten entsprechend der festgelegten Konzeption zu beschränken.

Auf den letzten beiden Hierarchiestufen werden schließlich die spezifischen Eigenschaften jeder Komponente zugeordnet, die durch ihre Parametrisierung die Funktionalität und das Design der Anwendung bestimmen. Alle Entitäten, die zur Ablage der Instanzvariablen der Komponenten dienen, stehen in Beziehung mit der Entität *Bereiche*. So kann Design und Funktionalität der gesamten Anwendung bereichs- bzw. nutzergruppenspezifisch festgelegt werden.

Die hierarchische Parameterstruktur schafft eine hohe Flexibilität beim Anwendungsaufbau und trägt gleichzeitig zur Verringerung des Anpassungsaufwandes bei. Beispiel dafür ist die leichte Realisierbarkeit beliebig vieler nutzergruppenspezifischer Angebotsbereiche. Mit anderen Standardsoftwarelösungen sind solche Angebote meist nur durch mehrfache Programmierung weitgehend identischer Strukturen aufzubauen. In der beschriebenen Datenbankstruktur werden diese lediglich durch die Parameter der jeweiligen Nutzergruppe und der nachgeordneten Hierarchiestufen spezifiziert. Alle Inhalte, die für übergeordnete Hierarchiestufen definiert wurden, werden automatisch in die Bereiche übernommen.

5.3.2 Abbildung von Komponentenexemplaren

Die Ausprägungen einer einzelnen Komponente werden im Modell stets durch eine Kopfentität und mindestens eine untergeordnete Elementeentität am Ende der Hierarchie der Parameterstruktur abgebildet. Die entsprechenden Relationen im relationalen Datenmodell werden als Komponentenstruktur- bzw. Elementerelationen bezeichnet. Die Strukturentität besitzt als Attribute alle Parameter, die grundsätzlich angegeben werden müssen, um die Komponente in das Framework zu integrieren. Daneben besitzt eine Komponente im einfacheren Fall eine Anzahl Parameter, die durch ihre interne Programmierung vorgegeben werden. Sie können durch die Entität *Parameter* und eine 1:N-Beziehung der jeweiligen Komponente zugeordnet werden. Auf diese Weise lässt sich für jede Komponente eine beliebige Anzahl gleichartiger Parameter festlegen. Dies ist entscheidend, um beispielsweise eine variable Anzahl von Interaktionselementen als Teil der Benutzerschnittstelle einer Komponente definieren zu können. Aufgrund der Restriktionen des HTML-Standards ist die parametrisierte Abbildung von Interaktionselementen und ihrer Anordnung innerhalb der Benutzerschnittstelle in dieser Form nahezu ohne Einschränkungen möglich, denn der HTML-Standard gibt eine beschränkte Anzahl eindeutig definierter Auszeichnungsanweisungen vor, mit der die Struktur des Bildschirmlayouts definiert wird (siehe 4.3.2).

Analog lassen sich für die Interaktion bei der Abwicklung von Teilprozessen auf dem elektronischen Markt Strukturen erkennen, die eine variable Anzahl gleichartiger Designelemente enthalten. Nach den Normalformregeln müssen alle Attribute, die Parameter dieser Wiederholgruppen beschreiben, in einer eigenen Relation abgebildet werden. Die beiden nachfolgenden Tabellen zeigen exemplarisch die Relationen, die benötigt werden, um beliebige Suchmasken für die Produktsuche in einem Online-Shop parametrisiert in HTML-Standard abzubilden. So können z. B. durch mehrfache Parameterdefinition in der Entität *Suchmaske_Elemente* für eine Suchmaske beliebig viele Eingabefelder und die zu jeder Eingabemaske gehörigen Tasten angelegt und flexibel angeordnet werden.

Tabelle 5-1: Strukturrelation zur Abbildung von Suchmasken

(1) Komponenten-ID	(2)-(5) Fremdschlüsselattribute	(6) Tabellarisch	(7) Benennungen (Ja/Nein)	(8) Rahmenbreite	(9) Schriftgrad	(10) Kopfdatei	(11) Fußdatei	usw.
1	...	J	J	0	B	KopfForm.html	FussForm.html	...
2	...	J	N	2	I			...

In den Spalten 2-5 der Komponentenkopfrelation werden die Schlüsselattribute angegeben, die die Komponente eindeutig identifizieren bzw. ihre Beziehungen zu den übergeordneten Stufen der Parameterstruktur definieren. Die Spalten 6-9 enthalten Attribute, die das grundsätzliche Design des Formulars bestimmen. Über den Eintrag in Spalte 6 kann z. B. festgelegt werden, ob die Formularelemente tabellarisch oder zeilenweise angeordnet werden. Die Angabe in Spalte 8 dient zum Ein- oder Ausblenden der in Spalte 8 der Elementerelation angegebenen Elementebezeichner. Durch Angabe entsprechender Dateinamen in Spalte 10 und 11 können oberhalb bzw. unterhalb des Bereiches, in dem die Formularelemente angezeigt werden, frei definierbare Designelemente eingefügt werden.

Tabelle 5-2: Elementerelation von Suchmasken

(1) Komponenten-ID	(2) Lfd. Nr.	(3) Vorausgehendes HTML	(4) Parametername	(5) Wert	(6) Typ	(7) Länge	(8) Folgekomponente	(9) Folgefunktion	(10) Nachfolgendes HTML	usw.
1	1	 Suchbegriff:	Suchbegriff		Text	40				...
1	2	 Maximaler Preis:	Max_Preis		Text	10				...
1	3		Such-Button	Suchen	Submit		Uebersicht	Select		...
1	4		Korrektur-Button	Korrigieren	Reset					...

Die Attribute 1–4 der Elementerelation dienen der Zuordnung der Eingabeelemente zu den übergeordneten Entitäten. In HTML-basierten Benutzerschnittstellen ist die Positionierung nicht frei wählbar, sondern wird in erster Linie durch die Reihenfolge der Auszeichnungsanweisungen innerhalb der Strukturbeschreibung definiert. Jedes Designelement erhält eine eindeutige Nummer, die die Reihenfolge der Elemente auf der Benutzerschnittstelle bestimmt. Die Formatierung eines Designelements kann durch Angabe entsprechender Formatanweisungen in den Spalten 3 bzw. 10 beliebig verändert werden.

Die Angaben der Spalten 4-7 definieren die Ausprägung und Funktion eines Formularelements. Der Parametername ist die Bezeichnung, mit der ein Wert über die Komponentenschnittstelle weitergegeben wird. Die Attribute Wert, Typ und Länge definieren die Ausprägung eines Formularelements entsprechend dem HTML-Standard.

Aufrufe von Funktionen anderer Komponenten werden durch Angabe der betroffenen Komponente und der gewünschten Funktion in den Spalten 8 bzw. 9 formuliert. Aus diesen Daten wird zur Laufzeit dynamisch ein entsprechender Funktionsaufruf generiert. Die Werte der angegebenen Parameter werden mit dem Aufruf als Funktionsparameter übermittelt.

Mit der in Tabelle 5-1 und Tabelle 5-2 vereinfacht dargestellten Relation sind beliebige Suchmasken für unterschiedlichste Zwecke beschreibbar. In der dargestellten Form sind nahezu alle Interaktionselemente zu verwenden, die im aktuellen HTML-Standard enthalten sind. Werden neue Elemente in den HTML-Standard aufgenommen, lässt sich die beschriebene Datenstruktur problemlos um neue Attribute ergänzen. Darüber hinaus ist durch die Möglichkeit zur Integration von HTML-Anweisungen in den Attributen 3 und 10, wo dies die Funktion der Komponente nicht beeinträchtigt, die Umsetzung beliebiger Designvorgaben nahezu uneingeschränkt möglich. Über die dynamische Parametrisierbarkeit der Aufrufe von Funktionen anderer Komponenten können Prozessketten frei definiert werden.

Analog zu der exemplarisch dargestellten Form zur Abbildung von Suchfunktionen lassen sich die Parameter weiterer Komponenten in einer relationalen Datenbank abbilden. Durch die Einordnung in die Hierarchie der Parameterverwaltung wird die Parametrisierung einer Vielzahl von Komponenten stark vereinfacht.

5.4 Erweiterung der Softwarebibliothek

Mit der Architektur des Komponentenframeworks und der Abbildung der Parameter in der relationalen Datenbank ist für die Integration beliebiger Komponenten eine einheitliche Struktur vorgegeben. Alle Komponenten, die innerhalb des Frameworks verwendet werden sollen, müssen zur Verarbeitung der Parameterinformationen eine Folge von Verarbeitungsschritten ausführen. Diese ist für alle Komponenten identisch und lässt sich allgemein beschreiben.

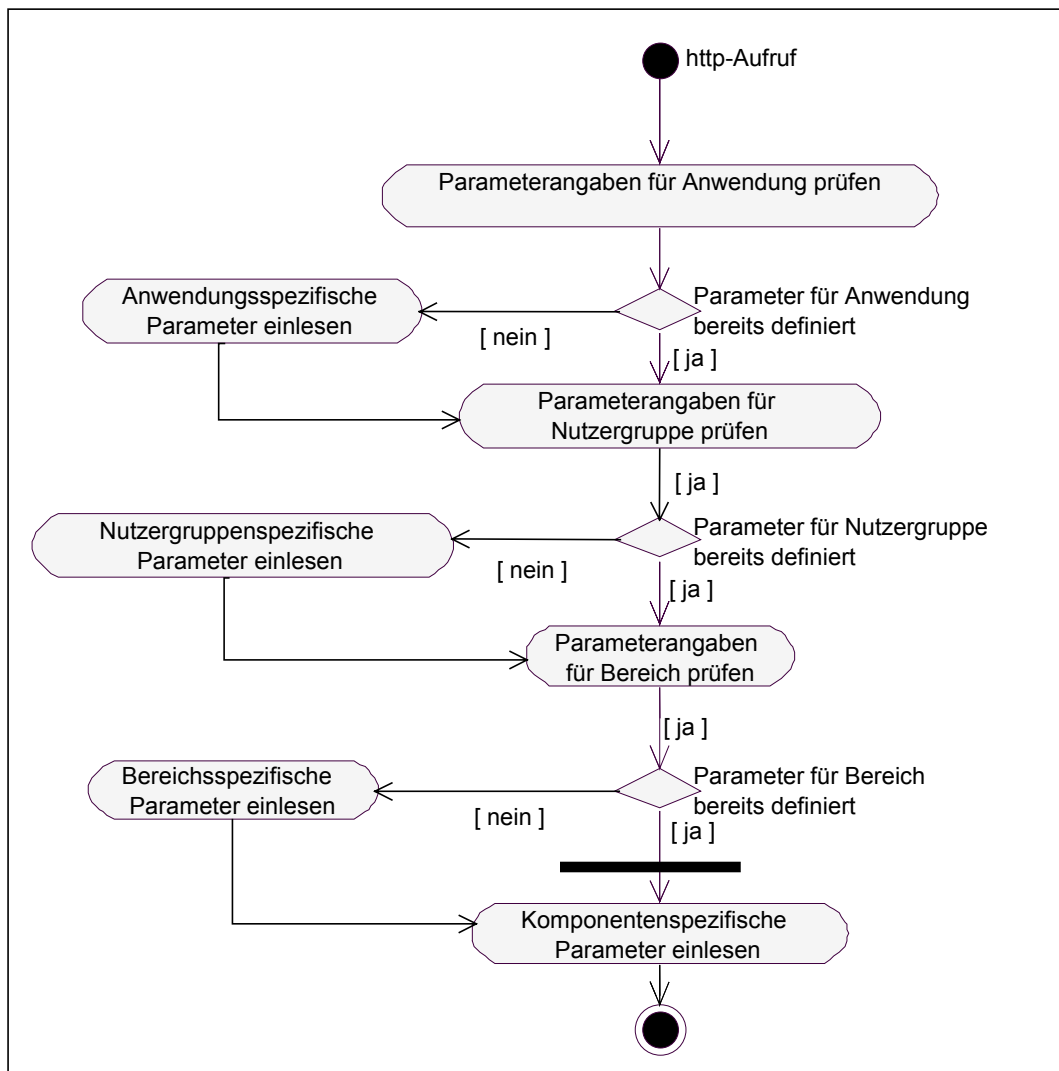


Abbildung 5-6: Ablauf beim Einlesen der Komponentenparameter

Wird eine Komponente durch ein Ergebnis aktiviert, müssen aus der Parameterdatenbank zuerst alle Parameterwerte ausgelesen werden, die auf den einzelnen Abstraktionsebenen der Anwendung mit Bezug auf die betroffene Komponente definiert wurden. Sind auf verschiedenen Abstraktionsebenen Parameter mit identischer Funktion definiert, gilt nach dem Prinzip der Vererbung jeweils der Wert der niedrigsten Abstraktionsstufe.

Zuletzt werden die Werte der Instanzvariablen der Komponente ausgelesen und die bereits ermittelten Parameter an die Komponente übergeben. Nur dieser letzte Verarbeitungsschritt ist komponentenspezifisch zu implementieren. Er definiert eine einfache und klar abgegrenzte Schnittstelle zur Integration von Komponenten in das Framework. Die übrige Programmstruktur kann bei der Implementierung der Standardanwendungssoftwarebibliothek in eine Unterprogramm- bzw. Klassenbibliothek ausgelagert und von mehreren Komponenten genutzt werden. Zur Integration neuer Komponenten sind dann lediglich die durch dieses Unterprogramm vorgegebenen Schnittstellen zu implementieren. In der Arbeit von [HESS00] konnte gezeigt werden, dass sich nach dem oben beschriebenen Verfahren neu entwickelte Komponenten auf einfache Weise in das Framework integrieren und zum Teil unmittelbar mit den vorhandenen Adaptionswerkzeugen anpassen lassen.

Die in Abbildung 5-6 dargestellte Programmstruktur kann auch zur Integration fremdbezogener Komponenten in das Framework eingesetzt werden. Bei fremdbezogenen Komponenten soll der Programmcode typischerweise nicht verändert werden oder ist erst gar nicht zugänglich. Um das Framework um solche Komponenten zu erweitern, müssen Schnittstellen für den Austausch der relevanten Eingangsparameter und der Ergebnisse der Komponentenexemplare geschaffen werden. Dazu genügt es, die in Abbildung 5-6 zuletzt dargestellte Aktivität zum Einlesen der komponentenspezifischen Parameter und die Angaben für die Funktionsaufrufe an die Schnittstellendefinition der zu integrierenden Komponente anzupassen. Eine vorgefertigte Programmstruktur, die die Programmierung einer Komponente so weit ergänzt oder erweitert, dass sie in ein Komponentenframework integriert werden kann, wird als Wrapper bezeichnet [O.A.97].

Bei der Ergänzung der Standardanwendungssoftwarebibliothek ist zu beachten, dass jede neue Komponente zur Lösung der Anforderungen mehrerer Anwender geeignet sein muss. Aus betriebswirtschaftlicher Sicht lassen sich die Anforderungen an das

Profil einer Komponente entweder induktiv aus der Forschung oder deduktiv aufgrund von Anforderungen des Marktes ableiten. Bei deduktiver Ableitung des Profils wird für Anforderungen aus mehreren Projekten ein Verfahren gesucht, das sich durch Parametrisierung möglichst gut an die jeweiligen Wünsche anpassen lässt. Dieses wird anschließend in der neuen Komponente abgebildet. Ein Beispiel wäre die Ableitung einer Komponente zur Präsentation variantenreicher Erzeugnisse wahlweise auf einer einzigen oder mehreren Web-Seiten. Bei induktiver Ableitung des Profils wird das abzubildende Verfahren nach betriebswirtschaftlichen, technischen oder auch juristischen Anforderungen entwickelt und entsprechend der zulässigen Anwendungsvarianten parametrisierbar abgebildet. Ein typischer Fall induktiver Ableitung wäre die Anforderungsdefinition für eine Komponente zur Verwaltung und Abrechnung von Online-Abonnements, die verschiedene Bezahlverfahren zulässt.

Die Mehrfachverwendbarkeit einer Komponente für verschiedene Anwendungsfälle hängt vom Standardisierungsgrad und der Adaptionfähigkeit der Komponente ab. Komponenten, die am Markt verfügbar sind, besitzen meist einen hohen Standardisierungsgrad, denn nur dann sind sie für einen breiteren Kundenkreis von Interesse. Sie besitzen jedoch häufig nur eine begrenzte Adaptionfähigkeit.

Aus technischer Sicht spielen der zu erwartende Aufwand für die Integration und Wartung der Komponente eine Rolle. Bei der Bewertung der Integrationsfähigkeit sind mögliche soft- und ggf. hardwaretechnische Inkompatibilitäten sowie der Aufwand für die Schnittstellenimplementierung zu berücksichtigen. Der Aufwand für die Wartung einer Komponente wird durch ihre Granularität bestimmt. Die Granularität gibt an, wie viele Funktionen in einer Komponente zusammengefasst werden. Sie ist bei fremdbezogenen Komponenten vorgegeben. Aus softwaretechnischer Sicht ist eine Komponente mit relativ geringem Funktionsumfang leichter zu warten und flexibler einsetzbar. Auf der anderen Seite steigt mit der Granularität der Kommunikationsaufwand zwischen den Komponenten. Im Idealfall führt jede Komponente einen für sich abgeschlossenen Teilprozess aus.

6 Konzeption des Anforderungsnavigators

Ziel dieses Kapitels ist die Konzeption eines Anforderungsnavigators zur Adaption der Standardanwendungssoftwarebibliothek aus Kapitel 5 nach dem in Kapitel 4 eingeführten ODYSSEUS-Konzept. Im CSE-Konzept ist der Anforderungsnavigator ein softwarebasiertes Werkzeug, das die Analyse der Systemanforderungen weitgehend unterstützt. Zentrale Anforderung aus der Zielsetzung der Arbeit ist darüber hinaus eine einfache Benutzerführung, die Mitarbeiter von KMU ohne technische Detailkenntnisse und mit begrenztem Aufwand in die Lage versetzt, selbstständig voll funktionsfähige Lösungen zu realisieren. Es müssen daher Ansätze gefunden werden, die alle Aktivitäten der Entwicklung eines elektronischen Marktes komfortabel unterstützen und sich nach dem ODYSSEUS-Konzept in einem werkzeuggestützten Adaptionenleitfaden integrieren lassen.

Abbildung 6-1 zeigt den Aufbau des Adaptionenleitfadens einschließlich aller werkzeuggestützten Aktivitäten und Informationsflüsse. Der Leitfaden gliedert die Einführung eines elektronischen Marktes in fünf aufeinander aufbauende Phasen. Innerhalb der Phasen wird im Abgleich mit der Standardanwendungssoftwarebibliothek durch Reduktion und Zuordnung die Konzeption für ein Marktsystem entsprechend der individuellen Rahmenbedingungen eines Unternehmens ermittelt und die Standardanwendungssoftwarebibliothek an die Anforderungen aus diesem Konzept angepasst. Durch die konsequente softwaretechnische Unterstützung aller Aktivitäten und vollständige Integration der Teilkomponenten des Adaptionenleitfadens ist ein Rückgriff auf bereits ermittelte Informationen jederzeit möglich. So ist der Nutzer nicht an die sequentielle Abarbeitung des Leitfadens gebunden und kann bisherige Ergebnisse jederzeit überarbeiten. Durch ständigen Abgleich aller gewonnenen Informationen wird dabei die Integrität der Lösung stets sichergestellt.

Der Leitfaden strukturiert das gesamte Projekt nach einer nachvollziehbaren Vorgehensweise in klar abgegrenzte Phasen mit logisch aufeinander aufbauenden Resultaten und führt so gezielt zur Realisierung des Projektes.

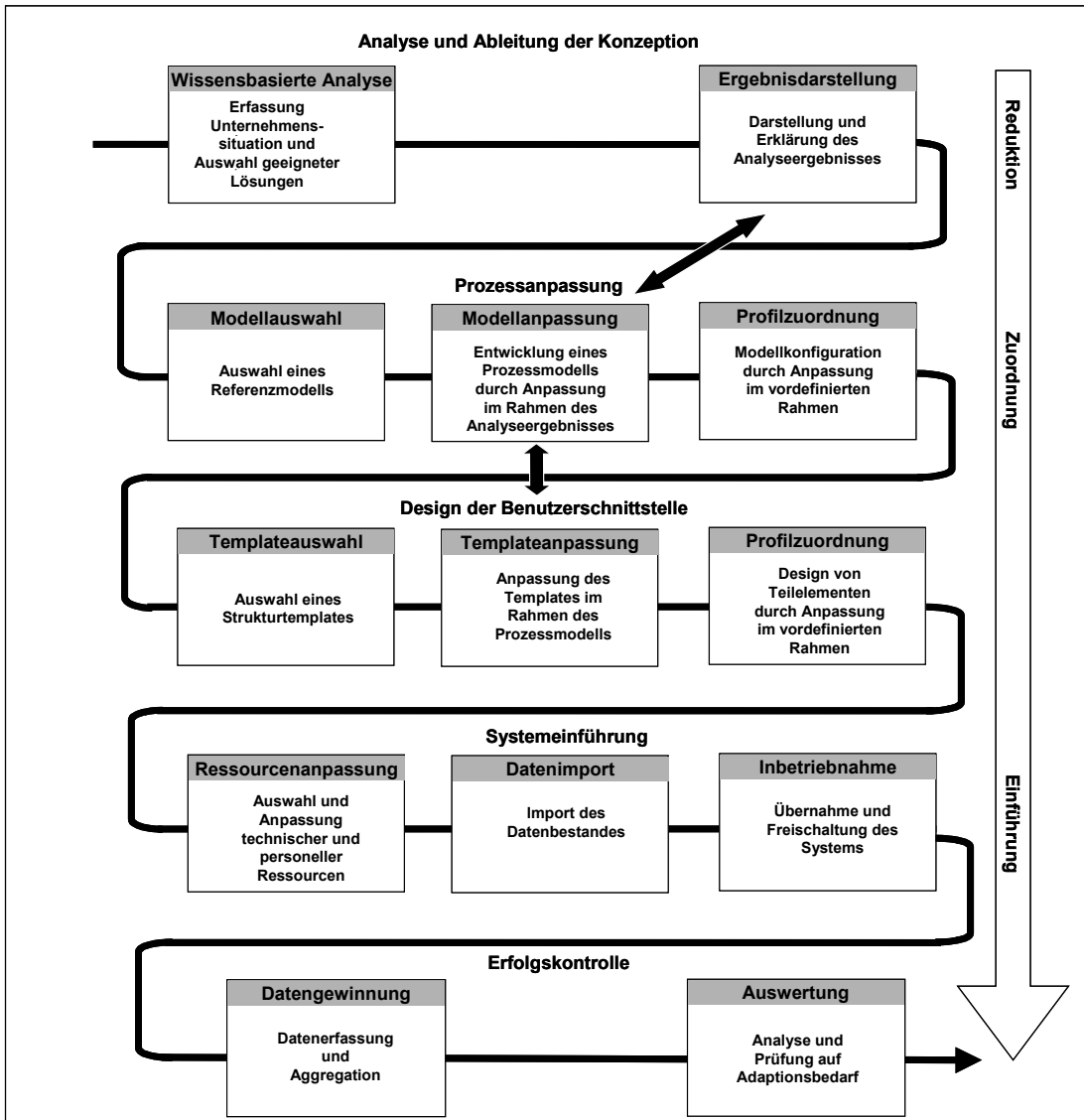


Abbildung 6-1: Gesamtdarstellung des Adaptionleitfadens

Die Gesamtdarstellung des Adaptionleitfadens zeigt, wie die Systementwicklung analog zu den Phasen des ODYSSEUS-Konzepts nur unter Verwendung der beiden dominierenden Adaptionarten Auswahl und Anpassung erfolgen kann. Die abschließende Erfolgskontrolle kann in diesem Zusammenhang als Hilfsmittel zur Erkennung eines entstehenden neuen Adaptionbedarfs und zur Identifikation von Lücken angesehen werden.

Es wird deutlich, dass der Einsatz flexibler Standardanwendungssoftware und die konsequente Werkzeugunterstützung bei der Adaption die Entwicklung elektronischer Märkte fundamental verändert.

Mit Hilfe des wissensbasierten Werkzeugs können in der Analyse aus ermittelten Fakten durch Anwendung von Inferenzstrategien unmittelbar Ziele für das gesamte Projekt abgeleitet und ein dazu passender konzeptioneller Vorschlag generiert werden. Im Gegensatz zur Individualentwicklung wird kein abstraktes System entworfen, das anschließend programmtechnisch umgesetzt werden muss. In allen Phasen des Adaptionleitfadens erfolgt eine konsequente Orientierung an den Möglichkeiten der Standardanwendungssoftware. Die Vorgehensweise der klassischen Softwareentwicklung wird damit quasi auf den Kopf gestellt.

Durch Auswahl der benötigten Komponenten und Zuordnung der betriebswirtschaftlichen Profile ist das Zielsystem nach Abschluss des Designs der Benutzerschnittstelle bereits vollständig beschrieben. Wird die gesamte Softwareanpassung auf die in 4.2.1 definierten Adaptionarten beschränkt, ist in einem vollständig integrierten Werkzeug eine automatisierte Überführung der Ergebnisse aller Adaptionphasen in ein lauffähiges System ohne jede Programmierung möglich. Das Ergebnis baut auf einer betriebswirtschaftlich fundierten Analyse auf, die eine klare Zielsetzung für das gesamte Projekt und den späteren Betrieb des Systems vorgibt. Programmieraufwand fällt nur noch dort an, wo Ergänzungsentwicklungen über die Funktionalität der Softwarebibliothek hinaus notwendig sind.

6.1 Wissensbasierte Abbildung der Analyse

Zu Beginn des Adaptionprozesses nach dem erweiterten ODYSSEUS-Konzept ist zu analysieren, welche Chancen sich einem Unternehmen auf elektronischen Märkten bieten. Diese sind in ein ökonomisch sinnvolles, technisch wie organisatorisch realisierbares Konzept umzusetzen (siehe 2.6). Dazu ist umfangreiches Faktenwissen über das jeweilige Unternehmen zu ermitteln und unter Berücksichtigung der ökonomischen Zusammenhänge und technischen Rahmenbedingungen auf elektronischen Märkten in ein Lösungskonzept zu überführen. Nach dem ODYSSEUS-Konzept müssen diese Aufgaben durch ein regelbasiertes Werkzeug unterstützt werden, das dieses Fachwissen bereitstellt. Damit soll auch sichergestellt werden, dass die bisher in zwei aufeinander folgenden Schritten durchgeführte Ist- oder Potential-Analyse und anschließende Konzeption (siehe Abbildung 3-2, S. 65) so weit integriert

werden, dass eine ständige Abstimmung der Erkenntnisse aus der Analyse mit der Projektzielsetzung möglich wird.

6.1.1 Auswahl einer Problemlösungsmethode

Systeme, in denen Wissen erfasst und abgebildet werden kann, werden allgemein als **wissensbasierte Systeme** bezeichnet. Kennzeichen aller wissensbasierten Systeme ist die Trennung zwischen einem Vorrat an allgemeinen Problemlösungsstrategien und dem abgebildeten Wissen [KURB92, S. 17 f.]. Ausprägungen wissensbasierter Systeme sind neben Sprach- und Bilderkennungssystemen u. a. Expertensysteme [HAUN00, S. 31]. Ein **Expertensystem** ist ein Informationssystem, welches das Fachwissen von Experten in einem eng abgegrenzten Anwendungsbereich verfügbar macht [HANS92, S. 452]. Wichtig ist zudem, dass dieses System spezielle Funktionen zur Verwaltung der Wissensbasis bietet [HAUN00, S. 31]. **Wissen** ist in diesem Zusammenhang von Fakten und Daten, etwa aus einer Datenbank, dadurch zu unterscheiden, dass es immer mit Verfahren zu seiner Interpretation gekoppelt ist [PUPP90, S. 1]. Für den Einsatz eines Expertensystems bedeutet dies, dass erst durch die korrekte Interpretation vorhandener und/oder beobachteter Fakten und durch Anwendung geeigneter Problemlösungsmethoden Wissen richtig verarbeitet werden kann. Zur wissensbasierten Abbildung der Analyse beim Aufbau eines elektronischen Marktes muss folglich untersucht werden, welche Problemlösungsmethoden geeignet sind.

Eine Einteilung der Anwendungsgebiete von Expertensystemen nach Problemklassen, denen später geeignete Lösungsstrategien zugeordnet werden können, nimmt PUPPE in [PUPP90, S. 42-180.] vor. Tabelle 6-1 fasst diese Einteilung zusammen.

Tabelle 6-1: Anwendungsgebiete von Expertensystemen nach Problemklassen

Problemklasse Ausprägungen	Charakteristische Eigenschaften der Problemklasse	Typische Anwendungs- bereiche
Klassifikation <ul style="list-style-type: none"> • Sichere Klassifikation • Heuristische Klassifikation • Überdeckende Klassifikation • Funktionale Klassifikation • Statische Klassifikation • Fallvergleichende Klassifikation 	<ul style="list-style-type: none"> • Der Problembereich besteht aus zwei disjunkten Mengen von Problemmerkmalen und Problemlösungen und aus mehrstufigem, typischerweise unsicherem Wissen über die Beziehungen zwischen beiden Mengen. • Das Ergebnis ist eine Auswahl einer oder mehrerer Problemlösungen. • Das Problem ist evtl. durch eine unvollständige Menge an Merkmalen charakterisiert. Wenn die Qualität der Lösung durch Erfassung zusätzlicher Merkmale verbessert werden kann, ist es Teilaufgabe der Klassifikation zu entscheiden, ob und welche Merkmale zusätzlich erfasst werden sollen. 	<ul style="list-style-type: none"> • Medizinische Diagnostik, Therapieauswahl • Qualitätskontrolle • Wertpapierberatung • Kreditberatung • Prüfung der Anwendbarkeit juristischer Bestimmungen
<ul style="list-style-type: none"> • Konstruktion • Heuristisches Konstruieren • Modellbasiertes Konstruieren • Fallvergleichende Konstruktion 	<p>Im Unterschied zur Klassifikation wird die Lösung nicht aus einer Lösungsmenge ausgewählt, sondern meist unter Restriktionen aus Lösungselementen zusammengesetzt.</p>	<ul style="list-style-type: none"> • Arbeitsplanerstellung • Maschinenbelegungsplanung • Konfiguration von Computern

Die Verwendung des Begriffs **Klassifikation** ist in der Literatur nicht eindeutig. Puppe verwendet ihn als Synonym für die Problemklasse „Diagnostik“ [PUPP90, S. 23; PUPP96, S. 1]. In [STEF95] wird „Classification“ als Problemlösen durch Auswahl verstanden und der Begriff **Diagnose** nur auf biologische Systeme bezogen. Eine genaue Abgrenzung zwischen Klassifikation und Konstruktion ist nicht in allen Fällen möglich. Besteht das Ergebnis einer Klassifikation aus mehreren Diagnosen, die mit einander interagieren, kann dies bereits als Zusammensetzung einer Lösung betrachtet werden.

Im ersten Schritt der Adaption einer Standardanwendungssoftware nach dem ODYSSEUS-Konzept steht das Ziel der Auswahl problemrelevanter Softwarebausteine im Vordergrund. Die Problemstellung kann deshalb als Klassifikationsproblem mit folgenden Eigenschaften abgebildet werden (vgl. auch Abbildung 6-4, S. 130):

- Die aktuelle Situation des Unternehmens wird durch die Ausprägungen der Problemerkmale erfasst.
- Das Ergebnis der Konzeption wird durch eine strukturierte Menge an Problemlösungen der Klassifikation beschrieben.
- Das Wissen über die Beziehungen zwischen den erfassten Informationen bezüglich der Situation eines Unternehmens und den sich daraus ergebenden Konsequenzen für die Konzeption eines elektronischen Marktes kann durch Anwendung von Problemlösungsstrategien spezieller Ausprägungen der Klassifikation abgebildet werden. Strategien zur Problemlösung wissensbasierter Systeme werden allgemein als Inferenzstrategien bezeichnet. Mit den Inferenzstrategien der Sicherer und der Heuristischen Klassifikation können sowohl gesicherte Erkenntnisse als auch unvollständiges oder ungesichertes Erfahrungswissen ausgewertet werden. Von Methoden weiterer Ausprägungen der Klassifikation kann abgesehen werden, da die fallvergleichende Klassifikation ihre Folgerungen aus Ähnlichkeiten mit Lösungen aus anderen Projekten ableitet, die aber auf elektronischen Märkten bewusst vermieden werden sollten. Die Überdeckende Klassifikation sucht nach Lösungen, die möglichst alle erkannten Merkmale erfüllen, was im vorliegenden Fall nicht zu ökonomisch sinnvollen Ergebnissen führt.

Die richtige Anwendung der verschiedenen Inferenzstrategien und die Formulierung von Regeln zur Abbildung des relevanten Wissens sind Problemstellungen, die beim Aufbau der Wissensbasis zu lösen sind.

6.1.2 Aufbau der Wissensbasis

Um Expertenwissen für eine konkrete Aufgabenstellung in einer Wissensbasis abzubilden, muss das problemrelevante Wissen ermittelt und unter Anwendung der gewählten Problemlösungsmethode so formalisiert und verallgemeinert werden, dass es auf eine möglichst große Anzahl von Aufgabenstellungen aus dem Problembereich übertragbar ist und das System für jede dieser Aufgabenstellungen aussagekräftige Ergebnisse liefert.

Die Formalisierung von Wissen zur Ableitung allgemein gültiger Aussagen scheint zunächst mit dem Ziel der Konzeption eines elektronischen Marktes im Widerspruch

zu stehen. Diese soll eine Gesamtlösung beschreiben, mit der sich ein Unternehmen vom Wettbewerb differenzieren und die eigenen Stärken bestmöglich ausspielen kann. Die Diskussion der ökonomischen Grundlagen in Kapitel 2 zeigt, dass auf elektronischen Märkten allgemein gültige Regeln existieren, die auf die Situation jedes Unternehmens anwendbar sind, um Chancen zu erkennen und Risiken zu vermeiden.

Grundsätzlich können beim Erwerb und bei der Formalisierung von Wissen zur Lösung eines Klassifikationsproblems zwei komplementäre Vorgehensweisen unterschieden werden [PUPP96, S. 68f.]:

- Bei der **lösungsorientierten Vorgehensweise** wird mit der Erfassung und Strukturierung aller im System abzubildenden Lösungen begonnen und anschließend werden die Merkmale ermittelt, die zu ihrer Herleitung relevant sind.
- Bei der **merkmalsorientierten Vorgehensweise** werden zunächst alle Merkmale erfasst und strukturiert, die im Zusammenhang mit der Problemstellung als wichtig erkannt werden. Anschließend werden die Lösungen aus den erfassten Merkmalen abgeleitet, strukturiert und die Abhängigkeiten zwischen Merkmalen und Lösungen als Regeln formuliert.

Auf den ersten Blick mag die lösungsorientierte Vorgehensweise zur Ableitung eines Ansatzes für den Einsatz einer Standardanwendungssoftware vorteilhaft erscheinen, da die Menge der technisch realisierbaren Lösungen bereits durch die Standardanwendungssoftware vorgegeben ist. Dies würde eine Beschränkung des Ergebnisses auf den aktuellen Umfang der Standardsoftware bedeuten. Der Nachteil der lösungsorientierten Vorgehensweise liegt in der geringen Flexibilität gegenüber Erweiterungen der Lösungsmenge, die oft umfangreiche Modifikationen nicht nur des Regelwerks, sondern auch der Merkmalsstruktur und damit der Benutzerführung des Systems erfordern.

Die merkmalsorientierte Vorgehensweise geht von einer vollständigen Analyse aller problemrelevanten Informationen aus. Damit können beim Aufbau der Wissensbasis für die vorliegende Aufgabenstellung nicht nur Informationen in Betracht gezogen werden, die mit technischen Aspekten des Systementwurfs zu tun haben, sondern auch organisatorische Maßnahmen und strategische Zielsetzungen im Zusammenhang mit der Einführung und dem Betrieb des Systems auf ihre Eignung untersucht werden. Werden dabei Informationen gewonnen, die nicht unmittelbar umsetzbar

sind, ist dies ein Hinweis auf Lücken. Wegen dieser entscheidenden Vorteile wird nachfolgend die merkmalsorientierte Vorgehensweise gewählt.

6.1.2.1 Formulierung des Fragenkatalogs

Die überwiegende Zahl der Merkmale zur Ableitung geeigneter Lösungen muss im interaktiven Dialog mit einem Mitarbeiter aus dem jeweiligen Unternehmen ermittelt werden. Daher ist ein Katalog an Fragen zu erarbeiten, der folgende Voraussetzungen erfüllt:

(1) *Sinnvoller Umfang*

Der Fragenkatalog muss so aufgebaut sein, dass bei vollständiger Abarbeitung alle relevanten Informationen zur Ableitung einer sinnvollen Analyse erfasst werden.

(2) *Klarheit und Übertragbarkeit der Fragestellung*

Die Formulierung der Fragen muss für einen möglichst breiten Anwenderkreis unmissverständlich und zugleich auf die Problemfelder einer hinreichend großen Zahl an Unternehmen übertragbar sein.

(3) *Reduktion des Informationsbedarfs*

Die Beantwortung der Fragen muss für den einzelnen Anwender mit minimalem Aufwand verbunden sein. Art und Umfang der Fragen müssen sich daher dynamisch an das Problemumfeld des Unternehmens anpassen.

Ein wissensbasiertes System bietet gegenüber anderen Medien zur Verarbeitung und Vermittlung von Wissen entscheidende Vorteile [PUPP99, S. 7]. Bereits ermittelte Informationen können sofort auf nachfolgende Fragestellungen angewendet werden. So kann der Umfang und die Formulierung der Fragen ausgehend von allgemeinen Fragestellungen mit fortschreitender Analyse auf spezifische Problemstellungen angepasst werden. Durch gezielte Rückfragen können widersprüchliche Aussagen des Anwenders, die auf Fehlinformationen oder missverstandene Fragestellungen hindeuten, sofort überprüft werden. Um diese Vorteile nutzen zu können, muss der Fragenkatalog unter zwei Aspekten strukturiert werden:

Bildung von Frageklassen

Es muss eine thematische Struktur des gesamten Katalogs entwickelt werden, die zusammengehörige Fragen zu Klassen gruppiert. Die Bildung von **Frageklassen** ist ein effizientes Mittel zum Aufbau einer guten Dialogsteuerung und fördert zudem das Verständnis der einzelnen Fragen. Der Aufbau der Frageklassenstruktur hat darüber hinaus wesentlichen Einfluss auf die langfristige Erweiterbarkeit der Wissensbasis. Innerhalb einer allgemein gültigen, aber dennoch aussagekräftig aufgebauten Frageklassenstruktur lassen sich ergänzende Fragestellungen leichter zuordnen und einfacher formulieren. Die einzelnen Frageklassen sind untereinander so anzuordnen, dass frühzeitig gewonnene Informationen zur Reduktion des Informationsbedarfs in den nachfolgenden Themenbereichen führen. Daraus ergibt sich eine stufenweise Konkretisierung der Fragestellung.

Fragendifferenzierung

Innerhalb der Frageklassen muss zwischen unbedingten Fragen und Folgefragen differenziert werden. **Unbedingte Fragen** werden in jedem Fall gestellt, **Folgefragen** dienen zur Gewinnung zusätzlicher Informationen bei Anwendung der Hypothesize-and-Test-Strategie. Um missverständliche Eingaben erst gar nicht zuzulassen, sollten Fragestellungen nach Möglichkeit in ihren Antwortalternativen auf eine Auswahl entscheidungsrelevanter Angaben beschränkt werden.

Abbildung 6-2 zeigt die Klassenstruktur eines Fragenkatalogs zur wissensbasierten Analyse und Ableitung der Konzeption für einen elektronischen Markt.

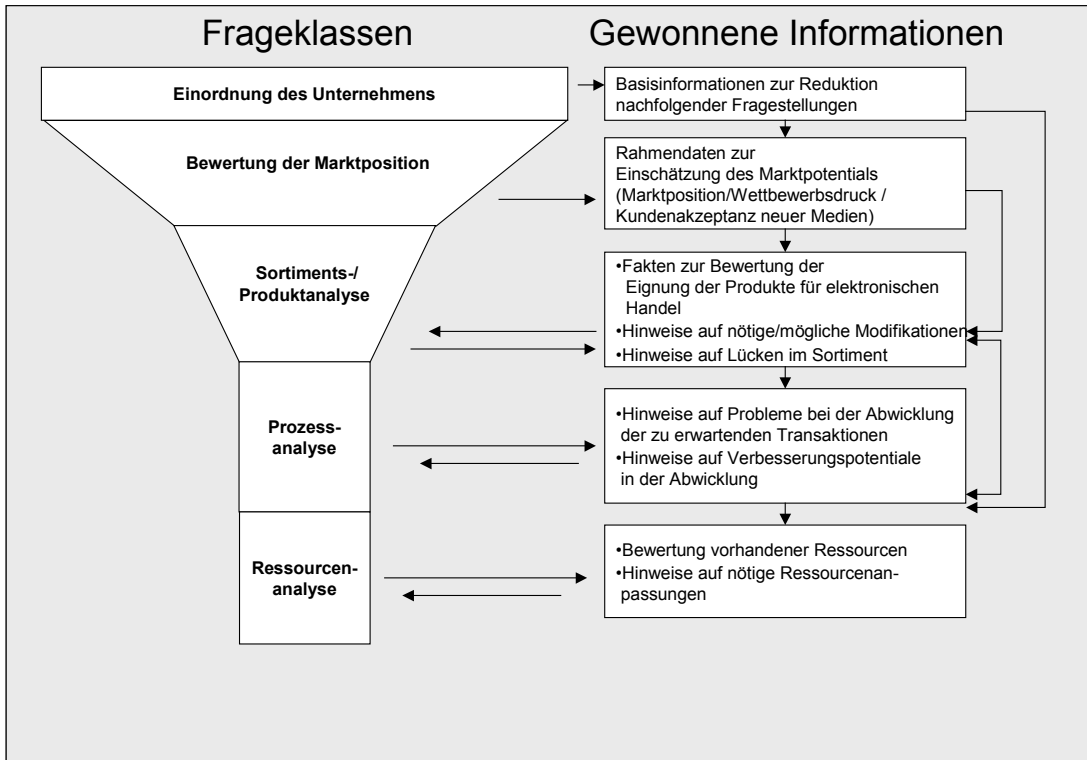


Abbildung 6-2: Klassenstruktur des Fragenkatalogs

Der gesamte Fragenkatalog ist in fünf Stufen gegliedert. Auf jeder Stufe wird eine klar abgrenzbare Fragestellung behandelt, die jeweils Informationen für die nachfolgenden Stufen liefert. Auf diese Weise wird eine Reduktion des Informationsbedarfs erreicht und die langfristige Erweiterbarkeit des Fragenkatalogs gewährleistet. Die gezielte Aufwandsreduktion in der wissensbasierten Analyse wird in Abbildung 6-2 durch die Y-Form der Frageklassenstruktur sichtbar.

Auf den ersten drei Stufen werden Informationen darüber ermittelt, welche Potentiale sich einem Unternehmen auf elektronischen Märkten bieten. Mit den Fragestellungen der Stufen vier und fünf wird dann gezielt untersucht, inwieweit das Unternehmen mit seiner Organisationsstruktur und den vorhandenen Ressourcen aktuell in der Lage ist, diese Potentiale zu nutzen. Aus dem Vergleich von erkannten Chancen und limitierenden Faktoren innerhalb des Unternehmens lassen sich Hinweise auf mögliche Verbesserungen ableiten.

Eine vollständige Darstellung aller in der Wissensbasis abgebildeten Fragestellungen und dynamischen Zusammenhänge ist in Papierform kaum sinnvoll möglich. Nachfolgend werden deshalb die zentralen Problemstellungen der Stufen des Fragenkatalogs

loges erläutert und zur Verdeutlichung des Aufbaus der Wissensbasis einzelne Fragestellungen beispielhaft ausgeführt. Eine grafische Darstellung des Fragenkatalogs findet sich im Anhang A1. Im Anhang A2 wird exemplarisch eine Analyse anhand des Fragenkatalogs durchgeführt.

6.1.2.1.1 Einordnung des Unternehmens

Bei Beginn der Analyse werden allgemeine Angaben zum Unternehmen erfasst. Sie dienen zur Einordnung des Unternehmens und der Reduktion der Menge relevanter Fragen. Anhand der gewonnenen Informationen können nachfolgende Fragestellungen unternehmensspezifisch ausgewählt werden.

Zur Klassifizierung von Unternehmen finden sich in der Literatur eine Reihe verschiedener Kriterien. In [MERT99, S. 71] werden Unternehmen nach Wirtschaftszweigen, vorherrschendem Produktionsfaktor und Betriebsgröße unterschieden. WÖHE gibt die Art der Leistungserstellung sowie die Rechtsform und die Standortabhängigkeit eines Unternehmens als weitere Unterscheidungskriterien an [WÖHE93, S. 104]. In einem wissensbasierten System lassen sich mehrere dieser Klassifizierungsschemata in einer Fragefolge miteinander kombinieren, soweit sie verwertbare Informationen liefern. Im Rahmen der Analyse für einen elektronischen Markt ist die Rechtsform des Unternehmens kaum von Bedeutung. Sinnvoll ist dagegen, zunächst eine grobe Zuordnung des untersuchten Unternehmens nach Zugehörigkeit zu Wirtschaftszweigen vorzunehmen [THOM00a, S. 7].

Da diese Einordnung nicht für alle Unternehmen eindeutig ist, sollten die Antwortmöglichkeiten additiv angeboten werden. Die erste Einordnung liefert Hinweise für eine Präzisierung nachfolgender Fragestellungen. So kann später im Rahmen der Bewertung der Marktposition und der Sortiments- und Produktanalyse bei produzierenden Unternehmen geprüft werden, ob Handelsmittlereffekte (z. B. durch Ausschaltung von Handelsmittlern mittels Direktvertrieb) zu realisieren sind. Bei Handelsunternehmen ist dagegen die Gefahr drohender Disintermediation zu untersuchen.

Innerhalb der gewählten Sektoren wird jeweils eine detailliertere Differenzierung nach Branchen vorgenommen. Um missverständlichen Zuordnungen vorzubeugen, sind für die Einteilung nach Branchen gängige und dem Anwender vertraute Branchenverzeichnisse geeignet. Die Angabe der Branche wird später verwendet, um

branchenspezifische Besonderheiten (z. B. Buchpreisbindung u. ä.) zu berücksichtigen.

Die Einordnung des Unternehmens nach Produktionsfaktor und Art der Leistungserstellung liefert Hinweise auf die Automatisierbarkeit der Abwicklung von Prozessen. So kann ein Unternehmen mit Werkstattfertigung u. U. eher in der Lage sein, auf individuelle Kundenanforderungen zu reagieren, als ein massen- oder serienfertigendes Unternehmen. Die Standortabhängigkeit spielt aus logistischer Sicht, aber auch bei der Entwicklung einer Marketing-Strategie für den elektronischen Markt eine Rolle.

6.1.2.1.2 Bewertung der Marktposition

Aufbauend auf der Einordnung des Unternehmens wird untersucht, welche Chancen und Risiken im Wettbewerb die Nutzung elektronischer Märkte für das Unternehmen bringt. Dabei sollen Informationen erhoben werden, die dem Kleinunternehmer leicht zugänglich sind.

Möglichkeiten zur Festigung oder gar Ausweitung der Marktposition bieten sich vor allem durch Nutzung von Kommunikationseffekten. Daneben müssen aber auch die Aktivitäten von Konkurrenten erfasst werden, um die eigenen Wettbewerbschancen richtig einzuschätzen oder auch drohende Handelsmittlereffekte rechtzeitig zu erkennen.

Um Potentiale zur Ausweitung der eigenen Absatzmöglichkeiten zu bewerten, ist zu ermitteln, ob und wenn ja welche Kundengruppen über elektronische Märkte zu erreichen sind. In diesem Zusammenhang ist die Bereitschaft der Kunden zur Nutzung der neuen Medien im Allgemeinen und die zu erwartende Akzeptanz des Produktangebots über Online-Medien näher zu untersuchen. Entscheidend für die Akzeptanz eines neuen Online-Angebots ist vor allem der Mehrwert, der im Vergleich zu existierenden Online- und Offline-Angeboten für die einzelnen Kunden entsteht [POON99, S. 14]. Für die Aufbereitung dieser Inhalte sind demographische und sozioökonomische Merkmale wie Alter, Vorbildung, die zu erwartende Bereitschaft zum Umgang mit Technologie und auch das Einkommen für eine zielgruppen-gerechte Anwendungsgestaltung relevant.

Grundsätzlich kann davon ausgegangen werden, dass sich die Demographie der Internetnutzer immer weiter der des Bevölkerungsdurchschnitts annähert [SCHN99, S. 34]. Jüngere Benutzer mit hohem Bildungsgrad und Einkommen sind allerdings weiterhin überproportional vertreten [FIT99, S. 77f.]. Diese Bevölkerungsgruppe hat vermutlich sehr hohe Ansprüche bezüglich der Informationsqualität und des Servicegrads eines Online-Angebots. Zur Differenzierung der Kunden in relevante Zielgruppen für den Einsatz multimedialer Systeme finden sich in [BRUH97, S. 59] eine Reihe von Kriterien. Das allgemeinste Unterscheidungskriterium zur Kundendifferenzierung ist danach die Information, ob es sich vorwiegend um Privat- und/oder Geschäftskunden handelt. Im Kontakt mit Geschäftskunden rücken Fragen der Geschwindigkeit der Abwicklung durch bestmögliche Prozessintegration und der Art und Häufigkeit der Geschäftsbeziehung vor marketingbezogenen Fragestellungen in den Vordergrund. Endkunden legen dagegen Wert auf umfassende Produktinformationen [THOM00, S. 4].

Eine Quelle umfangreicher Informationen für die Konkurrenzanalyse ist das Internet selbst. Unternehmensdarstellungen und vielfach veröffentlichte Pressemitteilungen geben einerseits Auskunft über die aktuelle Situation eines Unternehmens und liefern andererseits Informationen über geplante Aktivitäten. Der Nachteil dieser Informationen ist ihre geringe unmittelbare Vergleichbarkeit [GEOR01]. Für die Entscheidung zwischen Handlungsalternativen genügt aber in vielen Fällen schon eine grobe Bewertung der allgemein verfügbaren Informationen. Entsprechende Angaben lassen sich über das wissensbasierte System gezielt erfragen und auswerten.

So liefert beispielsweise schon die einfache Einstiegsfrage nach der Anzahl der Konkurrenten im Netz mit den Antwortalternativen: „Keine“, „Einige“, „Die überwiegende Zahl“ einen wichtigen ersten Hinweis auf den zu erwartenden Konkurrenzdruck. Diese Information hat Konsequenzen für die Durchführung von Marketingmaßnahmen und die Gestaltung von Preisen bei Markteinführung des Systems, die auf Stufe drei und vier der Analyse näher untersucht werden. Sind bereits Konkurrenzangebote im Netz, können diese nach einer Vielzahl unterschiedlicher Kriterien bewertet werden. Antworten auf die Frage, ob die Angebote wichtiger Konkurrenten bereits sehr umfassend sind oder nicht, lassen Rückschlüsse auf das Engagement der Wettbewerber zu. Gezielte Nachfragen, ob und in welchem Umfang

einzelne Marketingmaßnahmen (z. B. Auktionen oder Gewinnspiele) auf Websites der Konkurrenz häufig durchgeführt werden, lassen eine Bewertung der Erfolgsaussichten dieser Maßnahmen zu.

Die Evaluierung von Online-Angeboten liefert zudem oft wertvolle positive Anregungen, aber auch Hinweise auf Fehlerquellen. Sie kann nach einer Vielzahl an Kriterien erfolgen. Ein Beispiel für einen entsprechenden Kriterienkatalog findet sich unter [O.A.02]. In einem wissensbasierten System kann die Abfrage der Kriterien dynamisch angepasst und gezielt zur Gegenprüfung der abgeleiteten Ergebnisse herangezogen werden.

6.1.2.1.3 Sortiments- und Produktanalyse

Kommunikationseffekte lassen sich auf elektronischen Märkten nur realisieren, wenn die relevanten Eigenschaften der angebotenen Produkte über elektronische Medien gut zu vermitteln sind. Auch die Möglichkeiten, Integrations- und Netzeffekte zu erzielen, sind stark vom eigenen Sortiment und Produktangebot abhängig. Im Anschluss an die Frage, wie das Unternehmen am Markt positioniert ist, muss folglich untersucht werden, inwieweit die Produkte für ein Angebot über einen elektronischen Markt geeignet sind. Am Anfang stehen dabei zunächst einige Fragen nach Umfang und Aufteilung des Produktsortiments. Die gewonnenen Informationen ermöglichen eine produktgruppenbezogene Fragestellung im weiteren Verlauf der Analyse.

Wie die Vielfalt der mittlerweile im Internet verfügbaren Leistungen und Artikel zeigt, ist nahezu jedes Produkt über elektronische Medien zu handeln. Für eine umfassende Analyse wird deshalb die weit gefasste Definition des Produktbegriffs nach KOTLER zugrunde gelegt. Danach handelt es sich bei allem um ein Produkt, was am Markt als Objekt der Aufmerksamkeit, zum Erwerb oder Konsum angeboten werden kann. Dazu zählen neben konkreten Gegenständen und Dienstleistungen auch Personen, Organisationen und Ideen [KOTL95, S. 8-10]. KOTLER splittet den Produktbegriff in die Ebenen generisches Produkt, erwartetes Produkt, augmentiertes Produkt und potentiell Produkt mit Zusatznutzen auf. Diese Differenzierung trägt der Tatsache Rechnung, dass in Industriegesellschaften der Wettbewerb zunehmend über den Zusatznutzen (z. B. Service, Garantie, Image) ausgetragen wird, da vielfach

eine hohe Vergleichbarkeit und Imitierbarkeit der Produkte besteht [KOTL95, S. 659-667].

Aus Sicht der Anbieter ist entscheidend, ob sich die spezifischen Merkmale ihres Produktangebots zur Differenzierung gegenüber dem Wettbewerb erfolgreich über elektronische Medien vermitteln lassen. In der Literatur findet sich eine Reihe von Ansätzen, um die Eignung eines Produktes für den Handel über elektronische Kommunikationssysteme zu bewerten. Sie können in informationsökonomische und produktanalytische Ansätze unterschieden werden [ROHR97, S. 67-82].

In informationsökonomischen Ansätzen wird ein Produkt nach dem Aufwand zur Gewinnung der relevanten Produktinformationen eingestuft. Sie differenzieren zwischen digitalen Informationsgütern und Sachgütern. Informationsgüter sind digitale Produkte (z. B. Software, Bilder oder Musik) und über elektronische Medien angebotene Informationsdienstleistungen (z. B. Online-Recherchedienste, Forumdienste, elektronische Produktkataloge, Unternehmenspräsentationen). Für die Bewertung von Sachgütern hat sich im informationsökonomischen Ansatz nach dem Informationsbedarf zur Überprüfung der Qualität die Unterscheidung in Such-, Erfahrungs- und Vertrauensgüter durchgesetzt. Als Suchgüter werden Produkte eingestuft, deren Merkmale durch Informationsbeschaffung objektiv überprüfbar sind. Erfahrungsgüter sind dagegen Produkte, deren Eigenschaften erst nach längerem Gebrauch bewertbar sind. Vertrauensgüter besitzen Eigenschaften, die nicht vom Kunden überprüfbar sind, sodass er sich auf die Aussagen des Anbieters verlassen muss. Die Vermittlung von Such-, Erfahrungs- und Vertrauensmerkmalen lässt sich auf elektronischen Märkten in unterschiedlicher Weise unterstützen. Für eine exakte Differenzierung ist diese Einteilung allerdings wenig geeignet, da Produkte im Regelfall sowohl Such- wie auch Erfahrungs- und Vertrauenseigenschaften besitzen. Die jeweiligen Leistungseigenschaften sind zudem erst nach Vertragsabschluss zu erkennen. Der Wert des informationsökonomischen Ansatzes für die wissensbasierte Bewertung von Sachgütern durch den Anbieter ist daher begrenzt.

In produktanalytischen Ansätzen werden Produkte nach bestimmten Eigenschaften gruppiert. Da es kaum möglich scheint, einen Katalog von Eigenschaften zu entwickeln, der für alle Absatzleistungen Anwendung finden kann [MEFF82, S. 493], wird hier analog zu [ROHR97, S. 82-100] der umgekehrte Weg eingeschlagen. Es wird

davon ausgegangen, dass sich alle Produkte medial handeln lassen, und versucht, Eigenschaften zu identifizieren, die die Handelsfähigkeit von Produkten beeinflussen.

Diese Vorgehensweise bietet bei der Lösung der vorliegenden Aufgabenstellung entscheidende Vorteile. Ausgangspunkt der Analyse ist nicht eine umfassende Beschreibung von Produkteigenschaften, sondern die Prüfung auf entscheidende Faktoren für den elektronischen Handel, die sich allgemein gültig abbilden lassen. Die Beschränkung auf diese Faktoren führt zu einer starken Reduktion des Befragungsaufwandes. Die gewonnenen Informationen lassen unmittelbar Rückschlüsse auf mögliche Probleme zu.

In [ROHR97, S. 82-100] wird die Untersuchung darauf beschränkt, welche Produkteigenschaften den Kunden über elektronische Medien schwer zu vermitteln sind. In dieser eingeschränkten Betrachtung werden produktionstechnische oder logistisch relevante Einflussgrößen vernachlässigt. Das kann gerade dann zu gravierenden Problemen führen, wenn das Angebot bei den Kunden die erhoffte Akzeptanz findet. Auch lässt die Beschränkung auf limitierende Faktoren nicht erkennen, welche Produkteigenschaften den Handel über elektronische Medien besonders begünstigen. So werden beispielweise geringwertige Güter oft erst durch eine effizientere Abwicklung über elektronische Märkte zu minimalen bestellfixen Kosten für den Handel als eigenständige Produkte interessant. Für eine genaue Analyse muss deshalb die Betrachtung auf die der Produktpräsentation vorgelagerten Glieder der Wertschöpfungskette ausgedehnt und eine genauere Bewertung der Eigenschaften vorgenommen werden. Zudem ist eine Unterscheidung nach dem informationsökonomischen Ansatz sinnvoll, da Informationsgüter gerade bei der internen Abwicklung gegenüber Sachgütern zahlreiche Vorteile für den elektronischen Handel bieten. So lassen sich die relevanten Eigenschaften in eine Matrix einordnen (siehe Tabelle 6-2).

Tabelle 6-2: Auswahl relevanter Produkteigenschaften für den elektronischen Handel

	Produktbereitstellung (Produktion / Beschaffung)	Logistische Abwicklung	Vermarktung/ After Sales	Produkt- präsentation
Relevante Eigenschaften für In- formations- güter		▽ Umfang der Datenmenge bei der Übermittlung	▽ Technische Komplexität der Bereitstellung beim Kunden (z. B. Installation von Software)	<input type="checkbox"/> Ablauffähigkeit im Netz gegeben <input type="checkbox"/> Downloadfähige Demo-Version verfügbar <input type="checkbox"/> Produkt ist geringwertig △ Wert der Aktualität
Relevante Eigenschaften für alle Güter	▽ Variantenvielfalt ▽ Anpassungsbedarf beim Kunden ▽ Änderungshäufigkeit		▽ Wert des Produktes ▽ Servicebedürftigkeit ▽ Länderspezifität ▽ Verbundfähigkeit /-notwendigkeit mit Komplementärprodukten △ Anteil der bestellfixen Kosten am Verkaufspreis △ Anzahl der Bedarfsträger △ Geografische Verteilung der Bedarfsträger	▽ Komplexität des Produktes ▽ Relative Höhe des Preises ▽ Dringlichkeit des Bedarfes
Relevante Eigenschaften für Sachgüter	▽ Länge des Bereitstellungszeitraums △ Höhe des Deckungsbeitrags	△ Dauer der Lagerfähigkeit ▽ Versandkosten bzw. Versandfähigkeit		▽ Bemusterungsnotwendigkeit ▽ Höhe des Bedarfs an produktbezogenem Fachwissen

Legende:

- | | |
|--|--------------------------------------|
| <input type="checkbox"/> Eigenschaft prädestiniert das Produkt für den elektronischen Handel | ▽ Zunehmend limitierende Eigenschaft |
| △ Eignung des Produktes steigt mit zunehmender Ausprägung der Eigenschaft | |

Die tabellarische Darstellung der Auswahl an Eigenschaften macht bereits die wesentlichen Vorteile von Informationsgütern für den elektronischen Handel deutlich. Sie liegen in der automatisierbaren Bereitstellung und problemlosen elektronischen Übermittelbarkeit. Die Wertung der Eigenschaften muss allerdings differenziert betrachtet werden. Beispielsweise sind variantenreiche Erzeugnisse im Netz nur mit erhöhtem Aufwand in ihren Ausprägungen darstellbar. Gelingt es dennoch, diese über einen Produktkonfigurator dem Kunden vorzustellen, kann gerade dadurch eine hohe Kundenbindung erreicht werden. Die Bewertung einzelner Eigenschaften kann daher nur als Hinweis auf besondere Chancen oder den Bedarf an speziellen Lösungen angesehen werden. Zudem geht aus der zweidimensionalen Darstellung nicht hervor, welche Konsequenzen die Ausprägung einzelner Merkmale für die Bewertung anderer hat. So ist eine hohe Bemusterungsnotwendigkeit zunächst ein Nachteil für den elektronischen Handel, der aber durch eine aufwändige Informationsdarstellung im Web kompensiert werden kann. Ein hoher Bedarf an Fachwissen bei der Produktberatung könnte als Hinweis auf realisierbare positive Netzeffekte gewertet werden. Diese wären durch ein entsprechendes Angebot an Zusatzdienstleistungen zu nutzen. Je nach Zahl der Bedarfsträger und der avisierten Zielgruppen kann dies durch einen elektronischen Newsletter, durch Aufbau elektronischer Fachforen für Geschäftskunden oder auch eine breiter angelegte Virtuelle Gemeinschaft für Endkunden geschehen. Weitere Merkmale sind nur unter bestimmten Voraussetzungen sinnvoll auszuwerten. Ein Beispiel hierfür ist der Deckungsbeitrag eines Produktes, der nur dann absatzfördernd genutzt werden kann, wenn Preisdifferenzierung möglich ist.

In einem Klassifikationssystem können solche komplexen Zusammenhänge durch ein Netzwerk von Regeln exakt abgebildet werden, das die erfassten Merkmalsausprägungen zueinander in Beziehung setzt und zur Ableitung geeigneter Lösungen führt.

6.1.2.1.4 Prozessanalyse

Auf den bis hierher beschriebenen Stufen der wissensbasierten Analyse wird untersucht, welche Möglichkeiten sich einem Unternehmen aufgrund seiner aktuellen Situation auf elektronischen Märkten bieten und welche Maßnahmen daraus abgeleitet werden können. In der Prozessanalyse wird ermittelt, inwieweit sich diese im Unternehmen umsetzen lassen.

Nach der CSE-Philosophie ist das Ziel der Prozessanalyse nicht die radikale Veränderung etablierter Abläufe, sondern ihre Anpassung an die bestmögliche Nutzung eines elektronischen Marktsystems in tragbaren Schritten. Ausgangspunkt ist folglich die Fragestellung, welche relevanten Abläufe durch die bisherige Organisationsstruktur bereits unterstützt werden. Diese kann so aufgrund der bereits gewonnenen Informationen in vielen Fällen stark fokussiert werden. Wird beispielsweise bei der Auswertung der erfassten Produkteigenschaften vom wissensbasierten System im Rahmen einer Hypothesize-and-Test-Strategie (siehe 6.1.2.3) zur Kompensation einer hohen Bemusterungsnotwendigkeit eine stark multimediale Produktpräsentation als Hypothese vorgeschlagen, ist in der Prozessanalyse zu untersuchen, wie das Produkt bisher am Markt präsentiert wird und welche Dienstleister in diesen Prozess eingebunden sind.

Die Beschreibung der bisherigen Prozesse liefert Anhaltspunkte dafür, wo existierende Abläufe verbessert werden können oder neue entwickelt werden müssen. Zum Beispiel wäre zu prüfen, auf welchen Wegen Produkte zur Zeit beworben werden, um eventuelle Einsparungspotentiale durch elektronischen Produktvertrieb aufzuzeigen.

6.1.2.1.5 Ressourcenanalyse

Im Rahmen der Ressourcenanalyse wird schließlich erfragt, welche personellen und technischen Ressourcen benötigt werden, um die in der Prozessanalyse geplanten Abläufe ausreichend unterstützen zu können. Ziel ist die weitestgehend automatische Abwicklung aller in der Prozessanalyse vorgeschlagenen Prozesse. In der Ressourcenanalyse müssen deshalb im Dialog zwischen Anwender und wissensbasiertem System zunächst die vorhandenen Ressourcen erfasst und nach ihren Kapazitäten bewertet werden.

Im ersten Schritt ist zu klären, welche personellen Ressourcen in den betroffenen Bereichen des Unternehmens vorhanden sind und zur Unterstützung der geplanten Prozesse eingesetzt werden können. Dabei ist auch die Qualifizierung der verfügbaren Mitarbeiter zu berücksichtigen. Im zweiten Schritt ist zu klären, durch welche technischen Systeme (z. B. Warenwirtschaftssysteme, Office-Lösungen) die Mitarbeiter aktuell unterstützt werden und über welche Schnittstellen und Anbindungen zu weiteren Systemen diese verfügen (z. B. Datenbankschnittstellen, Online-Anbindung). Aus diesen Informationen kann unmittelbar der Bedarf an Qualifizierungs- und Beschaf-

fungsmaßnahmen abgeleitet werden. Mit Blick auf die Zielsetzung einer schrittweisen Anpassung des Unternehmens muss dabei eine Priorisierung der Maßnahmen im Abgleich mit der Prozessanalyse und den anderen Bereichen der heuristischen Analyse erfolgen. Im Beispiel aus der Prozessanalyse wäre in der Ressourcenanalyse zu prüfen, inwieweit ausreichend ausgebildete Mitarbeiter und entsprechende Bildbearbeitungssysteme vorhanden sind, um die Produktdarstellung im Web im Unternehmen selbst aufzubereiten. Wird dieser Vorschlag abgelehnt, ist zu untersuchen, ob externe Dienstleister in Anspruch genommen werden können.

Hier wird besonders deutlich, wie stark die Problembereiche der Konzeption eines elektronischen Marktes untereinander vernetzt sind, wie vielfältig die verschiedenen Fragestellungen sind und wie detailliert einzelne Teillösungen aufeinander abgestimmt werden müssen. Durch Anwendung der Inferenzstrategien eines wissensbasierten Systems passt sich die Fragestellung automatisch den Ergebnissen der zuvor abgearbeiteten Problembereiche an.

6.1.2.2 Strukturierung der Lösungsmenge

Nach der merkmalsorientierten Vorgehensweise ist im Anschluss an den Aufbau des Fragenkatalogs die Menge der relevanten Lösungen des Diagnosesystems zu definieren und zu strukturieren. Aus softwaretechnischer Sicht wird die Menge der Lösungen durch die Anzahl der verfügbaren Bausteine der Standardanwendungssoftware vorgegeben. Wie u. a. in der Ressourcenanalyse deutlich wurde, müssen Lösungen für einen elektronischen Markt nicht nur softwaretechnisch abgebildet, sondern auch durch organisatorische Maßnahmen unterstützt werden. Die Lösungsmenge ist deshalb um organisatorische Elemente zu ergänzen. Als Teil eines sinnvollen Konzepts muss zudem jeder Einsatz technischer und organisatorischer Mittel an klaren Zielvorgaben orientiert sein. Diese können folglich als übergeordnete Lösungselemente betrachtet werden. Solche unmittelbaren Zusammenhänge zwischen Lösungselementen können sehr effizient in mehrstufigen Hierarchien abgebildet werden [PUPP96, S. 102]. Dazu wird die Menge der Lösungselemente eines Diagnosesystems aus Grob- und Feindiagnosen aufgebaut. Daraus ergibt sich eine Strukturierung der Lösungsmenge, die zugleich eingesetzt werden kann, um bereits ermittelte Lösungsvorschläge weiter auszuwerten. Führt die Auswertung von Merkmalen zur Etablierung einer Grobdiagnose, können ihre untergeordneten

Feindiagnosen nach definierten Regeln für die Lösung in Betracht gezogen werden [PUPP00, S. 17]. In umgekehrter Richtung lassen sich ebenfalls Abhängigkeiten definieren, die zur Bestätigung der Relevanz einer Grobdiagnose herangezogen werden können.

Auf das Problem der strategischen Konzeption eines elektronischen Marktsystems übertragen, können Abhängigkeiten zwischen Lösungselementen genutzt werden, um geeignete Zielvorgaben abzuleiten und gleichzeitig die Vollständigkeit ermittelter Lösungen sicherzustellen. Abbildung 6-3 zeigt den Aufbau einer Lösungshierarchie für den Problembereich Elektronischer Vertrieb. Allgemeine Ziele bilden die übergeordneten Stufen des Hierarchiebaumes. Die Blätter sind in Abbildung 6-3 jeweils als Lösungselemente der Informationstechnologie mit (I) oder als organisatorische Maßnahmen mit (O) gekennzeichnet. Alle Lösungselemente sind mit unterschiedlicher Gewichtung ihren übergeordneten Elementen zugeordnet. Wird im Rahmen der Analyse eine Zielvorgabe empfohlen, führt dies automatisch dazu, dass alle technischen Komponenten und Realisierungsmaßnahmen zur Umsetzung dieses Ziels ebenfalls in Betracht gezogen werden. Durch unterschiedliche positive bzw. negative Gewichtung können dabei differenzierte Empfehlungen oder Ablehnungen formuliert werden. Gewichtungen unter 100% entsprechen lediglich einer Empfehlung, die evtl. durch weitere Auswertung von Fragen erhärtet werden muss. Elemente, die zur Realisierung eines übergeordneten Zieles zwingend erforderlich sind, erhalten dagegen die Gewichtung 100%. Durch die unmittelbare Zuordnung von Lösungselementen zu übergeordneten Zielen ist sichergestellt, dass stets alle notwendigen technischen Komponenten und organisatorischen Maßnahmen zur Realisierung eines Zieles berücksichtigt werden.

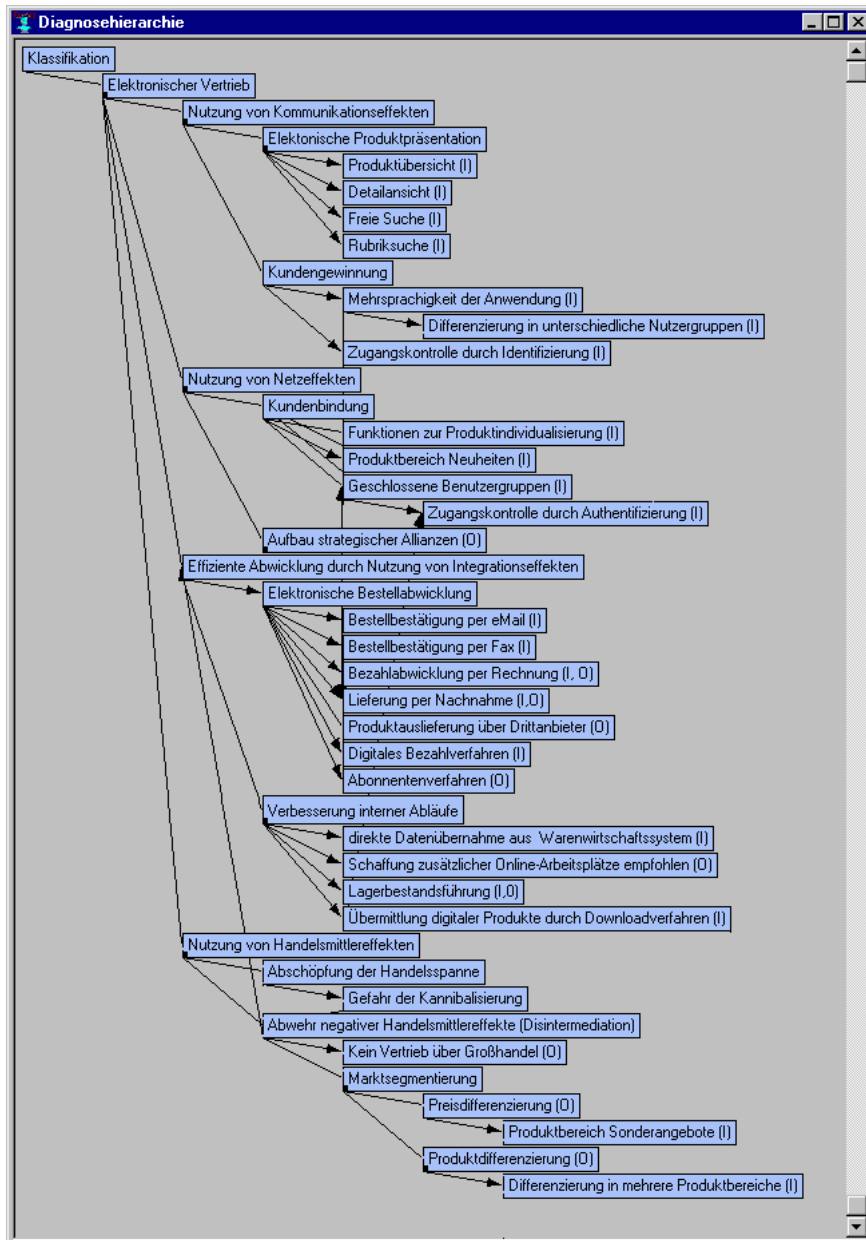


Abbildung 6-3: Heterarchisierung von Lösungselementen am Beispiel des elektronischen Vertriebs

Umgekehrt können durch Verknüpfungen zwischen Lösungselementen in entgegengesetzter Richtung aus Empfehlungen einzelner Lösungselemente Hinweise auf übergeordnete Ziele gewonnen werden. So ist z. B. die Empfehlung einer Funktion zur Produktindividualisierung schon ein geeignetes Mittel, um über das Angebot auf einem elektronischen Markt eine hohe Kundenbindung zu erreichen, die ggf. durch weitere Maßnahmen verstärkt werden sollte.

Über die hierarchische Verknüpfung mit mehreren übergeordneten Elementen können einzelne Lösungselemente auch mehrfach zur Realisierung verschiedener Ziele eingesetzt werden, sodass eine multiple Heterarchie entsteht. Beispielsweise ist die *Zugangskontrolle durch Authentifizierung* ein wichtiges Element zum Aufbau geschlossener Benutzergruppen, aber auch unerlässlich, wenn Produkte kostenpflichtig zum Herunterladen über ein Downloadverfahren angeboten werden sollen.

Die Heterarchisierung der Lösungsmenge nach Abbildung 6-3 bildet komplexe Beziehungen zwischen Lösungselementen unmittelbar ab, die nicht nur die Vollständigkeit der ermittelten Lösungsvorschläge sicherstellen, sondern in nachfolgenden Aktivitäten des Adaptionleitfadens auch zur Dokumentation des Analyseergebnisses (siehe 6.1.3) und zur Auswahl geeigneter Prozessmodelle (siehe 6.1.2.1.4) eingesetzt werden können.

6.1.2.3 Anwendung der Inferenzstrategien

Letzter Schritt beim Aufbau der Wissensbasis ist die Anwendung von Inferenzstrategien zur Ableitung von Lösungselementen aus den Problemmerkmalen, die durch die Abarbeitung des Fragenkatalogs ermittelt wurden. Zur Abbildung der wissensbasierten Analyse sind die Strategien der sicheren und der heuristischen Klassifikation geeignet (siehe 6.1.1).

Die Inferenzstrategie der sicheren Klassifikation ist die einfache Vorwärtsverkettung uneingeschränkt gültiger Regeln. Bei der **einfachen Vorwärtsverkettung** wird von bestimmten Problemmerkmalen unmittelbar auf ein Lösungselement verwiesen. Diese Inferenzstrategie ist nur dann einsetzbar, wenn der Zusammenhang zwischen Merkmalen und Lösungen durch induktiv belegbare Argumentation oder Erkenntnisse gesichert ist. Aussagen dieser Art besitzen allerdings einen hohen Wert. Lösungselemente, die durch sichere Klassifikation abgeleitet wurden, sind zwingend notwendige Bestandteile des ermittelten Konzepts.

Inferenzstrategien der heuristischen Klassifikation sind die Vorwärtsverkettung gewichteter Regeln, die Hypothesize-and-Test-Strategie und die Establish-and-Refine-Strategie, wobei die Establish-and-Refine-Strategie als Spezialfall der Hypothesize-and-Test-Strategie angesehen werden kann [PUPP96, S. 102], der geeignet ist, Entscheidungen zwischen ungesicherten Nachfolgeelementen einer Lösungshierarchie

herbeizuführen. Bei der **Vorwärtsverkettung gewichteter Regeln** wird über einen Wahrscheinlichkeitswert festgelegt, wie stark eine bestimmte Merkmalsausprägung auf ein Lösungselement schließen lässt. In der **Hypothesize-and-Test-Strategie** wird eine ermittelte Lösung zunächst hypothetisch angenommen. Durch Rückwärtsverkettung von Regeln wird versucht, weitere Informationen zu ermitteln, die diese Hypothese erhärten. Die Hypothesize-and-Test-Strategie ist auf viele Problemstellungen sehr flexibel und dennoch einfach anwendbar. Die Inferenzstrategien der heuristischen Klassifikation sind einzusetzen, wenn Unsicherheit bezüglich der Vollständigkeit der gewonnenen Informationen und/oder der Anwendbarkeit einer Regel auf das aktuell untersuchte Unternehmen besteht.

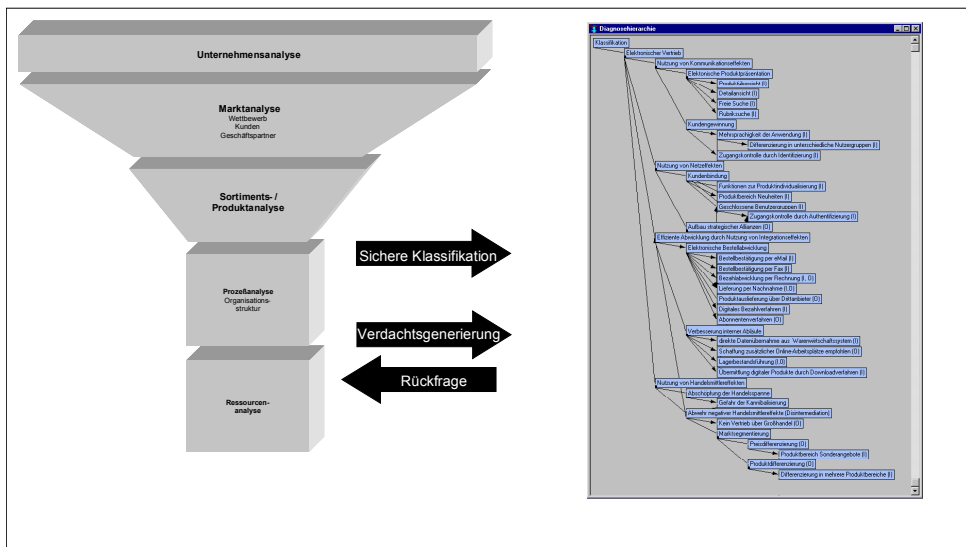


Abbildung 6-4: Abbildung der wissensbasierten Analyse als spezielles Klassifikationsproblem

Gerade bei der Bewertung von Strategien im Umfeld elektronischer Märkte sind Informationen sehr häufig nicht eindeutig abzusichern. Die Unterstützung des wissensbasierten Systems besteht bei der Auswertung dieser Informationen darin, durch Wertung der verfügbaren Informationen Lösungsvorschläge zu ermitteln und durch Rückfragen so weit wie möglich zu verifizieren. Diese gezielte Verarbeitung von Unsicherheiten unterstützt den Dialog und den Erkenntnisprozess mit dem Anwender und macht das System auf eine breite Zahl von Anwendungsbereichen übertragbar.

Nach der Anwendung der Inferenzstrategien auf die bekannten Merkmale der Problemstellung ist als Ergebnis der konzeptionelle Vorschlag für den elektronischen Markt ermittelt. Er besteht aus einer Anzahl von Lösungsansätzen, die z. T. mit zwingender Sicherheit aus den ermittelten Merkmalen folgen oder unbedingt abzulehnen sind. So enthält das Ergebnis u. a. Lösungselemente, die als Empfehlungen der heuristischen Klassifikation nicht mit Sicherheit bestätigt oder abgelehnt werden konnten. Sie müssen als optionale Vorschläge in das Lösungskonzept aufgenommen werden. Ein Beispiel für die Anwendung von Inferenzstrategien findet sich in Anhang A2.

6.1.3 Generierung der Ergebnisdokumentation

Aus dem Ergebnis der wissensbasierten Analyse kann vollständig automatisiert eine Ergebnisdokumentation erzeugt werden. Für die Akzeptanz des Resultates der wissensbasierten Analyse ist es unerlässlich, dass die Anwender die ermittelten Ergebnisse nachvollziehen und verstehen. Zu diesem Zweck bieten wissensbasierte Systeme im Allgemeinen eine eigene Erklärungskomponente. Im Adaptionleitfaden (siehe Abbildung 4-2) dient die Ausgabe der Erklärungskomponente nicht nur der Ergebnisdarstellung, sondern ist gleichzeitig Teil der Dokumentation der Reduktionsphase und damit Ausgangspunkt für die anschließenden Adaptionphasen.

In dem von HUFgard vorgeschlagenen Adaptionkonzept wird der Ergebnisbericht technisch über ein so genanntes Reduktionsdictionary in den Adaptionleitfaden integriert, das die Auswahl an Lösungselementen speichert und eine Beschreibung zu jedem Element enthält [HUFg94, S. 228f.]. In der Parameterverwaltung der Standardanwendungssoftwarebibliothek für elektronische Märkte hat diese Funktion das Verzeichnis *Benutzte_Komponenten*. Durch Zuordnung der Lösungselemente zu entsprechenden Anweisungen des DBMS können die Ergebnisse der wissensbasierten Analyse in das Verzeichnis übernommen werden. Die ausgewählten Lösungselemente werden durch weitere Zuordnung erläuternder Textbausteine von einem Berichtsgenerator automatisch zu einem Ergebnisbericht zusammengefügt. Ergänzt man das Verzeichnis *Benutzte_Komponenten* um Informationen über die Beziehungen zwischen den Lösungselementen aus der Lösungsheterarchie, lässt sich ein nach Themengebieten geordneter Ergebnisbericht erzeugen (siehe Abbildung 7-5, S. 152). Weitere Hintergrundinformationen zu den verschiedenen Lösungselementen können

ohne Hinzufügen neuer Attribute des Komponentenverzeichnisses oder durch Einfügen von Verweisen in die Erläuterung der Lösungselemente integriert werden, sodass ein Ergebnisbericht in Form eines Hypertextes entsteht, der die Konzeption des elektronischen Marktes detailliert beschreibt und Informationen in beliebiger Tiefe liefert.

Die Bewertung des Ergebnisberichts kann zu unterschiedlichen Reaktionen führen. Entweder wird das Ergebnis vollständig angenommen, abgelehnt oder in Teilen akzeptiert. Ursache für ein unbefriedigendes Ergebnis sind möglicherweise unvollständige oder fehlerhafte Angaben des Benutzers. In diesem Fall ist eine erneute Analyse mit besseren Informationen durchzuführen. Der Fehler kann aber auch auf Seiten des Systems liegen, wenn die individuelle Situation unzureichend oder gar falsch erkannt wird. Solche Fälle müssen zur kontinuierlichen Überprüfung und Verbesserung der Inhalte der Wissensbasis genutzt werden.

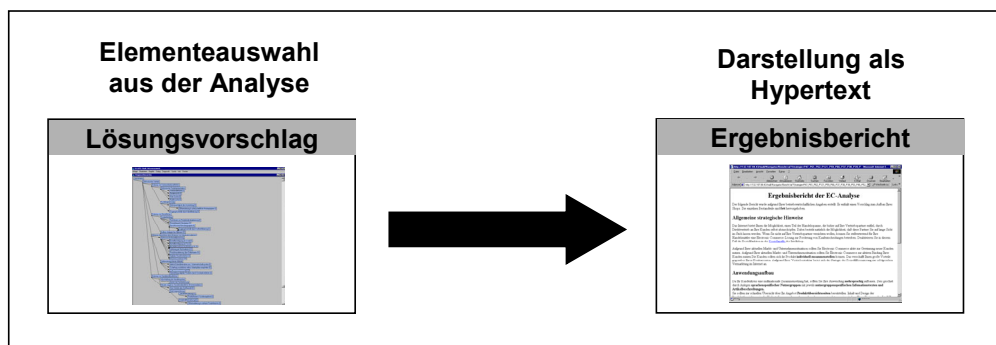


Abbildung 6-5: Generierung des Ergebnisberichts

Akzeptiert der Benutzer das Ergebnis, werden die vorgeschlagenen Lösungselemente mit ihren Gewichtungen in das Verzeichnis *Benutzte_Komponenten* übernommen. Diese Informationen genügen, um den Ergebnisbericht jederzeit reproduzieren zu können. Er ist damit vollständig dokumentiert und für andere Systeme in strukturierter Form verfügbar. Der Leitfaden kann für die weiteren Phasen der Adaption auf die ausgewählten Elemente reduziert werden.

6.2 Konzeption des Prozesseditors

Im nächsten Schritt des Adaptionleitfadens müssen die ausgewählten Softwarekomponenten zu Prozessen zusammengefügt werden. Im ODYSSEUS-Konzept geschieht dies durch Zuordnung betriebswirtschaftlicher Profile.

Zur Modellierung von Prozessen werden im Rahmen der Systementwicklung typischerweise grafische Modellierungswerkzeuge eingesetzt. Die Anforderungen bei der Abbildung von Prozessen für die digitale Geschäftsabwicklung sind allerdings besonders hoch, denn diese Prozesse müssen später weitestgehend automatisiert ablaufen, ohne dass manuelle Eingriffsmöglichkeiten bestehen. Das Problem im Rahmen der Adaption der Standardanwendungssoftwarebibliothek besteht nicht nur darin, alle Prozesse betriebswirtschaftlich korrekt, sondern auch entsprechend den Zielvorgaben des Konzeptes der ersten Phase des Adaptionleitfadens und unter bestmöglicher Nutzung des Lösungsvorrates der Softwarebibliothek zu formulieren. Hier zeigt sich ein wesentlicher Vorteil des PENELOPE-Konzeptes gegenüber rein grafischen Verfahren zur Unterstützung der Prozessmodellierung. Er liegt in der unmittelbaren Nutzung der Referenzmodelle der Standardanwendungssoftwarebibliothek. Bereits in Kapitel 3 wurde deutlich, dass technische Referenzmodelle zwar bei der Lösung schwieriger technischer Probleme eine wesentliche Hilfe darstellen, durch Nutzung betriebswirtschaftlicher Referenzmodelle aber ein noch höherer Produktivitätsvorteil entsteht.

Im Zusammenspiel mit einer Softwarebibliothek bieten betriebswirtschaftliche Referenzmodelle die Möglichkeit, verschiedene Alternativen der Implementierung einer konkreten Zielsetzung zu visualisieren. Aktive Modelle verfügen über Interaktionsmechanismen, um ein Ausgangsmodell individuell anzupassen. Im PENELOPE-Konzept werden deshalb Referenzmodelle zur Unterstützung der aktiven Modellierung von Prozessen eingesetzt [VOGE98, S. 195-213].

Bei der Entwicklung eines Werkzeuges, das die aktive Prozessmodellierung bei der Adaption von Standardanwendungssoftware wirksam unterstützt, sollten folgende Ziele verfolgt werden [VOGE98, S. 117f.]:

- *Kundenindividuelle, aktive Modellierung*

Ein geeignetes Referenzmodell wird in einem grafischen Editor mit Interaktionsmechanismen an den manipulierbaren Elementen dargestellt, an denen der Benutzer interaktiv Anpassungen vornehmen kann. Voraussetzung ist allerdings, dass das Referenzmodell als aktives Referenzmodell vollständig in die wissensbasierten Komponenten des Adaptionswerkzeugs integriert ist, damit die Integrität des Modells und die Konformität mit dem Ergebnis der wissensbasierten Analyse überprüft werden können.

- *Visualisierung getroffener Entscheidungen*

Im grafischen Editor müssen komplexe Zusammenhänge visuell veranschaulicht und getroffene Entscheidungen sofort überprüft werden können. Dazu sind Handlungsalternativen des Benutzers und Konsequenzen von Veränderungen unmittelbar grafisch darzustellen.

- *Demonstration des Funktionsumfangs der Standardanwendungssoftware*

Der grafische Editor muss ein Referenzmodell darstellen, das ein Abbild der Funktionalität der Standardsoftware repräsentiert. Daraus lassen sich Rückschlüsse auf den Funktionsumfang und Ideen für individuelle Implementierungsalternativen ableiten.

- *Veranschaulichung des organisatorischen Potentials der Standardsoftware*

Die Visualisierung alternativer Prozessvarianten muss möglich sein, damit deutlich wird, welche Freiheitsgrade die Standardsoftware zur Abbildung eines Prozesses bietet. Dadurch werden Implementierungsalternativen in Betracht gezogen, die ohne Werkzeugunterstützung möglicherweise übersehen werden.

- *Identifikation von Lücken/Brüchen*

Die grafische Darstellung soll den Abdeckungsgrad der Standardanwendungssoftware transparent machen und Rückschlüsse auf Lücken in der Implementierung erlauben. So werden Hinweise auf eventuell mögliche Nachbesserungen in der

Konzeption sichtbar. Die Entscheidung über die Eignung und Vollständigkeit der Lösung muss dem Anwender überlassen bleiben.

- *Aufzeigen von Defiziten der Standardsoftware*

Die visuelle Darstellung im Prozessmodell soll nicht nur die Potentiale, sondern auch die Grenzen der Standardanwendungssoftware aufzeigen. Sie sind Ansatzpunkte für mögliche Ergänzungsentwicklungen.

Bisher existieren für die Implementierung elektronischer Märkte noch keine allgemein anerkannten Referenzmodelle, wie dies für die Abbildung klassischer betriebswirtschaftlicher Abläufe schon seit Jahren der Fall ist. In einer umfassenden Untersuchung der relevanten Prozesse bei der digitalen Geschäftsabwicklung konnte aber gezeigt werden, dass sich auch die Abläufe auf elektronischen Märkten auf Kernprozesse zurückführen lassen, die allein durch betriebswirtschaftliche Anforderungen determiniert werden [THOM96a]. Über das wissensbasierte Adaptionswerkzeug können die Auswahl und die Anpassung eines geeigneten betriebswirtschaftlichen Modells aus dem Lösungsvorrat der Standardanwendungssoftwarebibliothek unmittelbar unterstützt werden. Aufgrund der wissensbasierten Auswahl ist damit die Vollständigkeit und betriebswirtschaftliche Korrektheit der Funktionsauswahl sichergestellt. Ist die Lösungsmenge der Wissensbasis in Form der in 6.1.2.2 entwickelten Lösungsheterarchie aufgebaut, liefert die Auswahl der Lösungselemente über die Rückwärtsverkettung gewichtete Informationen über wesentliche Ziele des Projekts. Ordnet man den Elementen der übergeordneten Ebene der Lösungsheterarchie geeignete Modelle zu, kann folglich ein vorkonfigurierter Modellvorschlag ermittelt werden, der die ausgewählten Elemente der Lösung enthält (siehe Abbildung 6-6). So wurde in der aktuellen Version des IntelliShop-Systems dem ersten Element der Lösungsheterarchie (vgl. Abbildung 6-3, S. 128) ein Prozessmodell für den elektronischen Vertrieb zugeordnet, das in einem Editor interaktiv angepasst werden kann (siehe 7.3.2.2). Weitere Modelle für die verschiedenen Zielsetzungen bzw. Anwendungsgebiete können entweder induktiv durch Analyse betriebswirtschaftlicher Abläufe oder deduktiv durch Abstraktion der Modelle am Markt verfügbarer Standardsoftwarelösungen ermittelt werden.

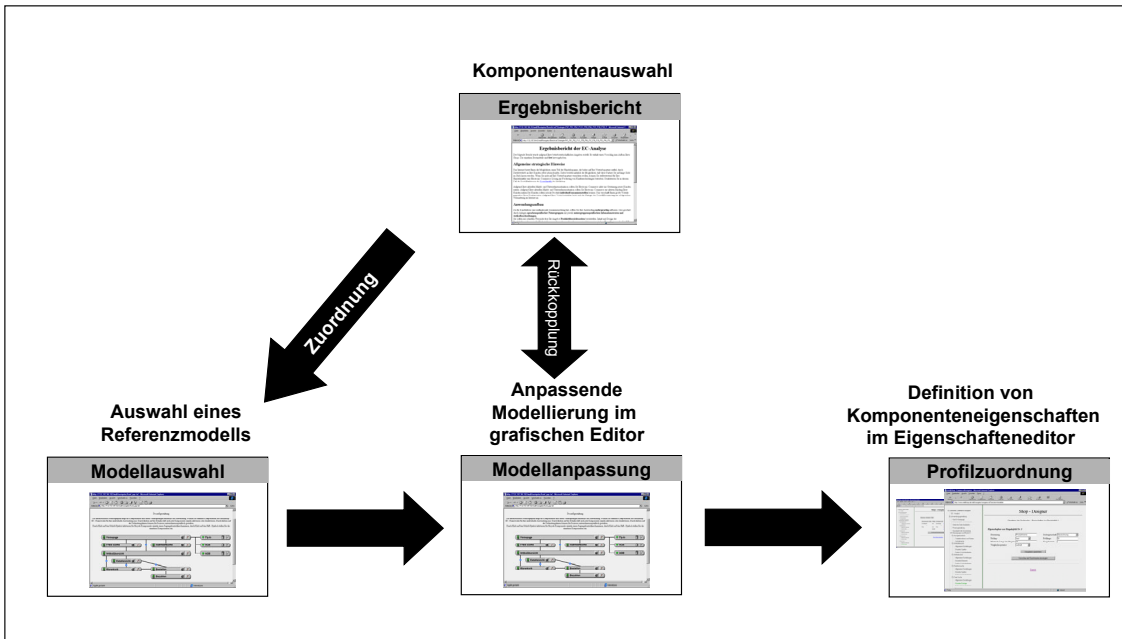


Abbildung 6-6: Integration und Anwendung der Werkzeuge zur Prozessmodellierung

Um dem Anwender eine Übersicht über die Potentiale der Softwarebibliothek und mögliche Prozessvarianten aufzuzeigen, wird das vorgeschlagene Modell so konfiguriert, dass Elemente, die sich während der Analyse als nicht zwingend erforderlich erwiesen haben, ebenfalls angezeigt werden. Sie sind mit einem gesonderten Statuskennzeichen zu versehen.

Damit der Modellvorschlag interaktiv modelliert werden kann, muss ein ständiger Abgleich zwischen den Anpassungen in einem grafischen Prozesseditor (siehe Abbildung 7-7, S. 155) und den Ergebnissen der Analyse stattfinden. Diese Rückkopplung zum wissensbasierten Analysewerkzeug ist erforderlich, um die Integrität der Gesamtlösung sicherzustellen. Die notwendige Integration kann über einen direkten Zugriff des Prozesseditors auf die Tabelle *Benutzte_Komponenten* erfolgen (siehe Abbildung 5-1). Dieses Verzeichnis enthält auch Informationen darüber, welche Elemente des Konzeptes zwingend erforderlich und welche optional wählbar sind. Die Zuordnung von betriebswirtschaftlichen Profilen zu den Komponenten des Referenzmodells erfolgt mit Hilfe komponentenspezifischer Eigenschafteneditoren (siehe Abbildung 7-9 und Abbildung 7-10, S. 159-160). Sie entsprechen in ihrer Funktion den Eigenschafteneditoren objektorientierter Entwicklungsumgebungen.

Die erfassten Werte werden in den Struktur- und Elementerelationen der jeweiligen Komponente (siehe 5.3.2) in der Parameterverwaltung ablegt.

6.3 Verfahren zum Design der Benutzerschnittstelle

Das Ergebnis der bis hierher beschriebenen Phasen des Adaptionprozesses ist ein vollständig funktionsfähiges System, das allerdings noch keine optisch ausgestaltete Benutzerschnittstelle besitzt. Da das Design der Benutzerschnittstelle stets in einem kreativen Prozess geschieht, ist es nur schwer möglich, diese Aufgabe in eine stringente Methodik zu fassen. Auf der definierten Datenstruktur der Softwarebibliothek können jedoch Werkzeuge aufgesetzt werden, die bereits gewonnene Informationen aus den vorherigen Adaptionphasen nutzen und den Designprozess in einem einfachen Verfahren unterstützen (siehe Abbildung 6-7). Es gliedert den Designprozess in den Entwurf der Anwendungsarchitektur (engl. Site-Design) und das Design der Inhalte (engl. Content-Design). Dies entspricht der in der Praxis bewährten und in der Literatur zum Web-Design (z. B. [SIEG98, S. 15-24] und [BAUM98, S. 17-20]) häufig empfohlenen Vorgehensweise.

Entwurf der Anwendungsarchitektur

Ziel des Entwurfs der Anwendungsarchitektur im Rahmen des Designs der Benutzerschnittstelle ist die Definition klarer und für den Anwender unmittelbar einsichtiger Strukturen, die eine schnelle Übersicht, einfache Orientierung und komfortable Navigation im gesamten Angebot des Marktsystems ermöglichen. Wichtige Elemente der Anwendungsarchitektur sind ein einheitliches Design und übergreifende Navigationselemente. Als Vorgabe bieten sich die Richtlinien für das Corporate Design des Anbieterunternehmens an. Zur Unterstützung der Navigation haben sich Navigationsleisten etabliert, die in separaten Bildschirmabschnitten Verknüpfungen für einen schnellen Zugriff auf die verschiedenen Angebotsbereiche und wichtige Funktionen enthalten.

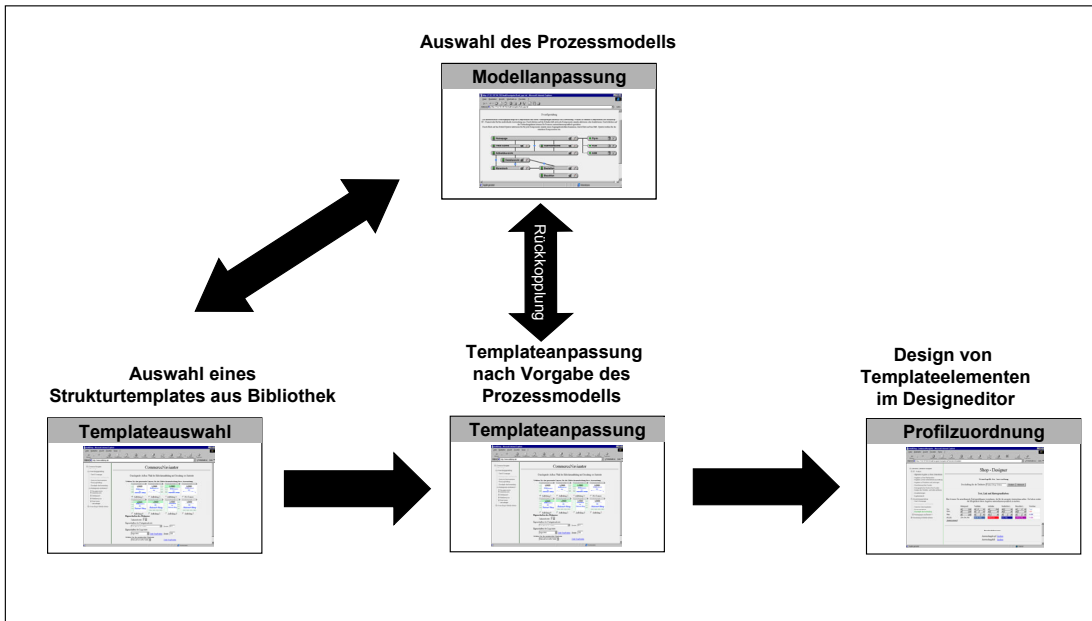


Abbildung 6-7: Werkzeugbasierte Unterstützung des Anwendungsdesigns

Die Informationen für den Entwurf der Anwendungsarchitektur können aus den Ergebnissen der wissensbasierten Analyse und der darauf aufbauenden Prozessmodellierung bezogen werden. Die wissensbasierte Analyse gibt den funktionalen Rahmen und eine grobe Strukturierung der Anwendung in Nutzergruppen und Angebotsbereiche vor. Bei der Prozessmodellierung werden aufgrund dieser Vorgaben Prozessketten gebildet. Jeder Beginn einer Prozesskette markiert einen möglichen Einstiegspunkt in die Anwendung, der folglich auch als Verknüpfung in der Navigationsleiste enthalten sein sollte. Weitere Einstiegspunkte sind zentrale Anwendungskomponenten, die persistent Informationen über den aktuellen Status des jeweiligen Nutzers speichern (z. B. Warenkorb, Guthabenverwaltung, u. Ä.). Aus diesen Informationen kann ein Template für die Bildschirmdarstellung automatisch ausgewählt und vorkonfiguriert werden. Auch dieser Vorschlag muss interaktiv angepasst werden können. Dazu sind interaktive Editoren bereitzustellen, die direkt auf die Ergebnisse der Prozessmodellierung zugreifen. Auf diese Weise findet indirekt eine Rückkopplung zu den Ergebnissen der Analyse statt.

Definition von Inhaltsstrukturen

In einem elektronischen Marktsystem ist es sinnvoll, Informationen überwiegend transaktionsorientiert aufgrund von Benutzereingaben aufzubereiten und darzustellen. Im Gegensatz zum Design statischer Seiten im WWW lässt sich die Darstellung dynamischer Informationen nicht pixelgenau definieren. Es können lediglich Strukturen festgelegt werden, die erst auf Anfrage eines Nutzers mit Inhalten gefüllt werden. Nur so ist eine weitgehend automatisierte, interaktive Informationsverarbeitung möglich.

Die im vorherigen Kapitel konzipierte Standardanwendungssoftware ist gezielt auf dynamische Informationsdarstellung ausgerichtet. In der Parameterverwaltung können auf Anwendungs-, Bereichs-, Nutzergruppen- und Komponentenebene Designparameter spezifiziert werden. Darüber hinaus sind in den Elementerelationen der Komponenten Datenfelder für HTML-Code vorgesehen, durch die sich die einzelnen Interaktionselemente der Benutzerschnittstelle flexibel formatieren und anordnen lassen. Mit diesen Mitteln lassen sich die Gestaltungsmöglichkeiten des HTML-Standards nahezu uneingeschränkt ausschöpfen, ohne dass dabei die Funktionsfähigkeit der Komponenten beeinträchtigt wird. Durch den hierarchischen Aufbau der Parameterverwaltung kann jeder ausgewählten Komponente nach Bedarf ein bereichsspezifisches und zu jedem Bereich ein nutzergruppenspezifisches Profil zugeordnet werden.

6.4 Unterstützung der Systemeinführung

Nach Abschluss des Designs der Benutzerschnittstelle kann das System eingeführt werden. Die Einführung umfasst die technische und organisatorische Integration des Systems in das Unternehmen und seine vollständige Inbetriebnahme. Wichtige Aufgaben der Systemeinführung im Allgemeinen sind:

- Übernahme auf den Betriebsrechner,
- Anpassung personeller Ressourcen und Qualifikationen sowie der technischen Ausstattung,
- Import von Produkt- und Mediendaten,
- Inbetriebnahme des Systems.

Der Aufwand für Installation und Betrieb eines eigenen Servers ist für KMU, die nicht bereits über Betriebspersonal für Server-Systeme verfügen, meist zu hoch. Die im vorigen Kapitel entwickelte Standardanwendungssoftwarebibliothek bietet dagegen einem professionellen Dienstleister ideale Möglichkeiten, um auf einer Plattform für eine größere Anzahl von KMU Online-Angebote unterschiedlichster Art zu betreiben, die diese mit Hilfe eines integrierten Anforderungsnavigators selbstständig pflegen und ausbauen (siehe 7.3).

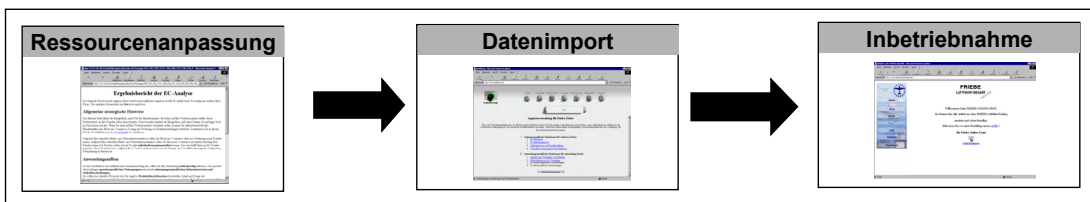


Abbildung 6-8: Ablauf der Systemeinführung

Der Import und die Pflege der Daten müssen über entsprechende Funktionen im Back-Office-Bereich der Standardanwendungssoftware erfolgen.

Organisatorische Maßnahmen im Rahmen der Systemeinführung können durch ein softwarebasiertes Werkzeug nicht unmittelbar unterstützt werden. Hinweise darauf, welche Maßnahmen zur erfolgreichen Einführung des Systems entscheidend sind, können dem Ergebnisbericht aus der wissensbasierten Analyse entnommen werden.

6.5 Integration der Erfolgskontrolle

Erfolgskontrollen im laufenden Betrieb sind sinnvollerweise Bestandteil jedes langfristig angelegten Projekts. Im Rahmen der Adaption nach dem ODYSSEUS-Konzept kann die Erfolgskontrolle als Mittel zur Identifikation von Lücken oder als Anhaltspunkt für die Durchführung eines neuen Adaptionsschrittes herangezogen werden.

Aus der wissensbasierten Analyse sind die Ziele des Projekts bekannt und in einem strukturierten Analysebericht abgelegt. Durch Auswertung von Daten aus dem laufenden Anwendungsbetrieb kann im Vergleich mit den dokumentierten Zielen eine sehr exakte Erfolgskontrolle erfolgen (siehe Abbildung 6-1, S. 108).

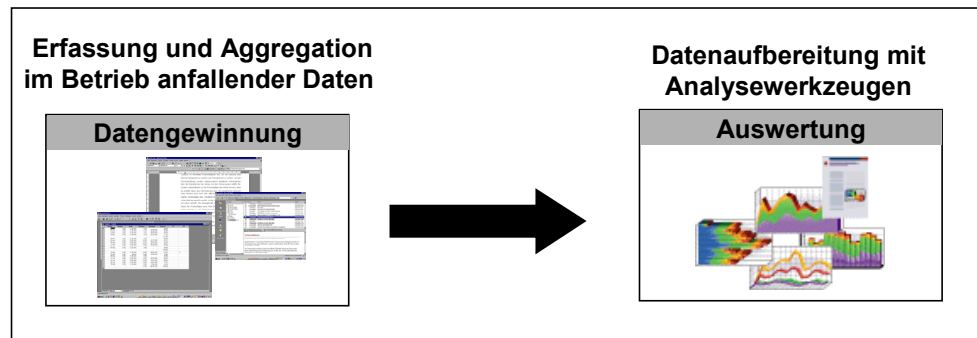


Abbildung 6-9: Durchführung der Erfolgskontrolle

Quellen zur Gewinnung von Informationen über die Akzeptanz des Angebots auf einem elektronischen Markt sind Protokolldateien, Nutzereingaben und E-Mails. Aus ihnen können mit sehr unterschiedlichem Aufwand Informationen verschiedenster Qualität für die spätere Auswertung gewonnen werden.

Protokolldateien

Innerhalb einer Client-Server-Architektur führen die einzelnen Serversysteme üblicherweise Protokolldateien mit, um die Nutzung ihrer Dienste transparent zu machen und Transaktionen zu sichern. Je nach Servereinstellung werden vollautomatisch detaillierte Informationen über die Interaktionen der Nutzer mit dem Serversystem erfasst. Besonders aufschlussreich ist die Protokolldatei des WWW-Servers, denn sie enthält neben den Informationen über die ausgeführten Aktionen eines Nutzers auch noch sehr differenzierte Angaben darüber, welche Technologien bzw. Produkte für den Zugang zum elektronischen Markt genutzt werden. In den meisten Fällen ist sich der Nutzer dessen kaum bewusst. Der Auswertung dieser Dateien wird deshalb vielfach eine hohe Bedeutung zugemessen. Eine ganze Reihe marktgängiger Produkte unterstützt die Auswertung von Webserver-Protokollen nach verschiedenen Größen (z. B. Anzahl der Seitenzugriffe, Anzahl der Zugriffe auf eine Anwendung, Anzahl der identifizierten Nutzer). Vereinzelt wurden sogar umfassende Kennzahlensysteme entwickelt, die diese Größen zueinander in Beziehung setzen, um differenzierte Aussagen zu gewinnen [LOHS02, S. 77; BRUH97, S. 109-127]. Die Aussagekraft dieser Kennzahlen ist allerdings begrenzt, da Webserver-Protokolle aus verschiedensten technischen Gründen (z. B. Zwischenspeicherung von Daten auf Proxi-Servern und in Browsern, wechselnde Nutzeradressen in überzeichneten Adresspools u. Ä.) [KNÜP00, S. 253] nur bedingt Informationen über die reelle

Nutzung des Angebots liefern. Aufgrund von Protokollauswertungen lassen sich nur signifikante Veränderungen der Nutzung erkennen. Über die Ursachen von Problemen, wie mangelnde Akzeptanz durch die Zielgruppe oder eine geringe Anzahl bis zum Abschluss ausgeführter Transaktionen, lassen sich aus Serverprotokollen kaum Informationen gewinnen [SCHM99b, S. 59].

Aussagekräftiger ist die Verfolgung der Wege einzelner Nutzer durch Ermittlung sog. Clickstreams [GLUC00]. Vollständige Clickstreams können aber nur ermittelt werden, wenn jede Komponente der Web-Anwendung über Session-Control-Funktionen verfügt. Viele Standardanwendungssoftwarelösungen bieten diese Funktion nur in definierten Bereichen zur Transaktionsabwicklung. Ein als Softwarebibliothek konzipiertes Werkzeug muss Session-Control-Funktionen in allen Komponenten enthalten, damit diese universell einsetzbar sind.

E-Mail

Nach der immer noch verbreiteten Internet-Konvention (Netikette) sollte den Nutzern jeder Internet-Anwendung Gelegenheit gegeben werden, das Angebot zu kommentieren, was aus Gründen der Einfachheit meist über Mail-Funktionen geschieht. Dieses Medium ist durch seinen informellen Charakter vielfach eine Quelle von wertvollen Anregungen und Hinweisen für die weitere Entwicklung des Angebots. Da die auf diesem Weg gewonnenen Informationen nur unstrukturiert vorliegen, kann die Bearbeitung allerdings kaum automatisiert erfolgen.

Nutzereingaben

Über elektronische Formulare können gezielt Informationen von den Nutzern erfragt werden. Diese Angaben besitzen einen hohen Wert, da sie strukturiert vorliegen. Bei vorher erfolgter Nutzerauthentifizierung sind sie sogar kundenbezogen. In der Summe können die gezielte Abfrage und Auswertung von Nutzereingaben sehr genaue Angaben über die Präferenzen und die demografische Struktur einzelner Nutzergruppen eines elektronischen Marktes liefern, die Grundlage für die gezielte Ausrichtung des Online-Angebots an den Wünschen der Kunden ist. Die Auswertung dieser Daten ist aufgrund der Vielfältigkeit möglicher Fragestellungen kaum mit einem einzigen Werkzeug sinnvoll möglich. Üblicherweise werden dazu gängige Standardsoftwarepakete (z. B. Tabellenkalkulationsprogramme und Lösungen für

statistische Auswertungen) eingesetzt, die sich flexibel zur Auswertung von Datenmengen mit überschaubarer Komplexität einsetzen lassen.

Transaktionsdaten

Transaktionsdaten müssen zwangsläufig strukturiert erfasst werden und beziehen sich stets auf einen Geschäftsvorfall. Ihre Auswertung für Marketing- und Vertriebszwecke etwa durch Data Mining ist lange vor der Verbreitung elektronischer Märkte ein wichtiges Thema der Wirtschaftsinformatik gewesen und erlebt im Zuge der wachsenden Akzeptanz einen neuerlichen Anschub. Transaktionsdatenauswertungen liefern detaillierte, oft sogar personenbezogene Informationen über das Nutzerverhalten und über den realen wirtschaftlichen Erfolg des Produktangebots. Neue Marketingstrategien wie Customer Relationship Management (CRM) nutzen Data-Mining-Systeme als zentrale Komponente ihrer elektronischen Vertriebsplattform. Auch die Auswertung von Nutzereingaben ist eine wichtige Quelle für CRM.

Tabelle 6-3: Zusammenfassung der Auswertungsmöglichkeiten von Datenquellen und Bewertung der Ergebnisse für die Erfolgskontrolle

	Möglichkeiten der Auswertung	Bewertung des Auswertungsergebnisses
Protokolldateien	Marktgängige Auswertungswerkzeuge ermöglichen Aggregation der Protokollinformationen nach Häufigkeiten.	Nur relative Änderungen der Nutzung erkennbar. Auswertung von Clickstreams liefert Informationen über die Akzeptanz der Anwendungsarchitektur.
E-Mail	Automatisierte Auswertung nicht möglich.	Auswertung ist sehr aufwändig, liefert aber vereinzelt entscheidende Informationen.
Nutzereingaben	Durch gezielte Datenauswertung können Informationen über die Zielgruppen gewonnen werden. Die Auswertung von Nutzereingaben wird kaum durch standardisierte Verfahren unterstützt.	Auswertung ist aufwändig, liefert aber bei gezielten Aktionen detaillierte Informationen über Kundenwünsche.
Transaktionsdaten	Auswertung kann durch Anwendung von Data-Mining-Verfahren unterstützt werden.	Auswertung liefert differenzierte Informationen über den tatsächlichen Erfolg des Angebots.

Es wird deutlich, dass zur Auswertung aller Daten, die während des Betriebes eines elektronischen Marktes anfallen, mehrere Werkzeuge eingesetzt werden müssen. Eine automatische Aufbereitung der Ergebnisse ist nur für standardisierte Fragestellungen möglich. Zur Erfolgskontrolle der jeweils letzten Anpassungen des Marktsystems an neue Anforderungen sind sie dennoch wichtige Entscheidungsgrundlagen.

7 Realisierung des IntelliShop-Systems

In diesem Kapitel wird die Realisierung des IntelliShop-Systems beschrieben. Der IntelliShop wurde am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg entwickelt. Dabei wurden die beschriebenen Konzepte in einem vollständig integrierten Werkzeug umgesetzt.

Mit Blick auf die Zielgruppe wurde der Schwerpunkt auf die Realisierung von Komponenten und Adaptionswerkzeugen zum Aufbau von Shop-Systemen gelegt. Anforderungen aus Projekten mit den ersten Pilotanwendern führten zur Entwicklung weiterer Komponenten, z. B. für den Bereich der Virtuellen Gemeinschaften. Zur Zeit ist das IntelliShop-System in der Version 2.02 verfügbar. Während mit der ersten Version vor allem die Realisierbarkeit der Konzepte nachgewiesen werden sollte, wurde die zweite Version für den Einsatz in Anwenderprojekten entwickelt. Zu den wesentlichen Verbesserungen der aktuellen Version gegenüber dem ersten Prototyp zählen u. a. eine deutliche Ausweitung des Komponentenvorrats der Standardanwendungssoftware, eine Erweiterung und verbesserte Integration der Wissensbasis des Expertensystems sowie eine stärkere Endbenutzerorientierung der Benutzerschnittstelle des Anforderungsnavigators und der Administrationsfunktionen im Back-Office-Bereich des Systems.

Im ersten Abschnitt dieses Kapitels wird zunächst näher auf die technischen Grundlagen des IntelliShop-Systems eingegangen. Die weitere Darstellung folgt im Wesentlichen dem Ablauf bei der Erstellung eines Online-Shop-Systems aus Anwendersicht.

7.1 Systemarchitektur

Das IntelliShop-System ist als Mall-System realisiert, das auf einem zentralen Server von einem Service-Provider betrieben wird. So können mehrere Anbieter auf einer gemeinsamen Plattform parallel Anwendungen entwickeln. Technische Probleme werden im Bedarfsfall vom Service-Provider gelöst, sodass auf Seiten der Anbieter kein Know-how über den Betrieb eines Internet-Systems benötigt wird. Die einheit-

liche Struktur aller Anwendungen macht den direkten Zugriff auf die Angebote jedes Anbieters über eine zentrale Suchfunktion auf der Leitseite der Mall möglich. Dort finden sich weitere Funktionen zum Aufbau eines Systems und der Zugang zu den Verwaltungsfunktionen.

Eine wesentliche Eigenschaft des IntelliShop-Systems ist die vollständige Integration aller Teilsysteme.

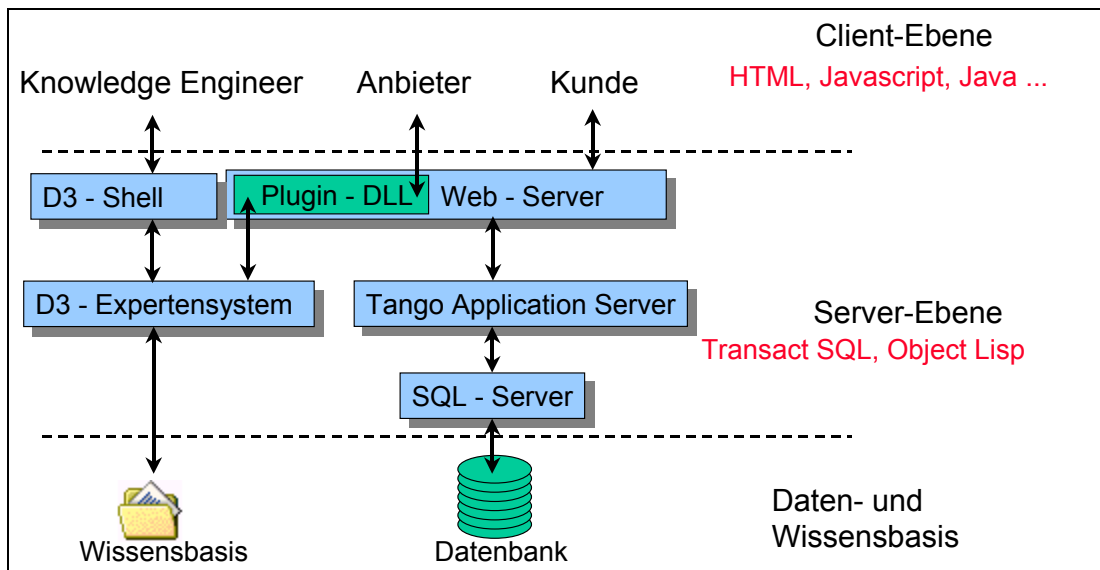


Abbildung 7-1: Systemarchitektur des IntelliShop-Systems

Die Realisierung erfolgte unter Verwendung einer Reihe verschiedener Technologien und Werkzeuge auf Basis des Betriebssystems WINDOWS-NT. Abbildung 7-1 zeigt die Architektur des Systems sowie die eingesetzten Technologien und technischen Komponenten.

Das System ist nach einer Three-Tier-Architektur aufgebaut. Für die Anbieter und Nutzer wurden in der Mall jeweils getrennte Anwendungsbereiche mit eigenen Sicherheitsmechanismen geschaffen. Der Anwendungsbereich der Anbieter ist mit einem eigenen Passwortschutz versehen. Er schützt den Zugang jedes Anbieters zum Anforderungsnavigator und den Administrationsfunktionen für die Verwaltung der Geschäftsdaten. Der Zugriff auf die Komponenten der Softwarebibliothek ist nicht grundsätzlich beschränkt. Die Zugangskontrollverfahren werden innerhalb jeder Anwendung individuell durch Einsatz von speziellen Komponenten realisiert.

Die Anwendungslogik des Anforderungsnavigators, des Back-Office-Bereichs und des Frameworks der Standardanwendungssoftwarebibliothek wurden mit dem Produkt Tango Enterprise der Firma Pervasive Software entwickelt. Dieses Werkzeug basiert auf einem leistungsfähigen Application Server (siehe 3.3.2.2) und bietet eine sehr gute Unterstützung bei der Realisierung datenbankbasierter Webanwendungen. Weitere Vorteile sind die hohe Plattformunabhängigkeit und die vielfältigen Möglichkeiten, externe Systeme zu integrieren. Versionen von Tango Enterprise sind für MAC OS, MS-WINDOWS-NT bzw. MS-WINDOWS 98 verfügbar. Im Lieferumfang sind eine Entwicklungsumgebung und ein Application Server enthalten. Die Entwicklungsumgebung integriert leistungsfähige Editoren für die Programmierung HTML-basierter Benutzerschnittstellen, die die Formulierung von Datenbankabfrage und Entwicklung von Verarbeitungsprozessen visuell unterstützen. Über eine offene Schnittstelle können zudem beliebige, auch in anderen Rechnern befindliche Java-Klassen, WINDOWS-DLLs, Unix- oder Apple-Scripte aufgerufen und die Ergebnisse weiterverarbeitet werden. Der Application Server verfügt über eine standardisierte ODBC-Datenbankschnittstelle und über weitere proprietäre Schnittstellen zu verschiedenen Datenbankverwaltungssystemen führender Hersteller. Alle mit Tango Enterprise entwickelten Softwarekomponenten werden vom Application Server zur Laufzeit interpretiert, sodass eine plattform- und datenbankunabhängige Entwicklung möglich ist. Über den Webserver wird der Aufruf der einzelnen Programmmodule und Komponenten an den Tango Application Server weitergeleitet. Dieser führt die Programmteile aus und steuert den Zugriff auf das Datenbankverwaltungssystem.

Als Datenbankverwaltungssystem wurde der SQL-SERVER der Firma Microsoft eingesetzt. Dieser verwaltet die Parameter, Standardanwendungssoftwarebibliothek und die Geschäftsdaten, die im laufenden Betrieb des Systems anfallen.

Zur Umsetzung der Wissensbasis wurde die Expertensystem-Shell D3 eingesetzt. Modifikationen an den Inhalten der Wissensbasis des IntelliShops können nur mit Hilfe der Wissenserwerbskomponente von D3 vorgenommen werden. Diese ist nur auf dem Rechner zugänglich, auf dem das Expertensystem installiert ist. Für kommende Versionen von D3 ist ein webbasiertes Front-End für den Zugriff auf die Wissenserwerbskomponente über das Internet geplant.

7.2 Aufbau und Integration des Expertensystems

Die Expertensystem-Shell D3 wurde im Rahmen der Forschungsaufgaben am Lehrstuhl für Künstliche Intelligenz und Angewandte Informatik der Universität Würzburg entwickelt und ist mittlerweile auch als kommerzielles Produkt verfügbar. D3 ist ein problemspezifisches Werkzeug zur Erstellung von Diagnosesystemen. Dieses System bietet alle wichtigen Problemlösungsmethoden für Klassifikationsprobleme (siehe Tabelle 6-1). Regeln können sowohl in Entscheidungstabellen als auch in Form von Entscheidungsbäumen abgebildet werden.

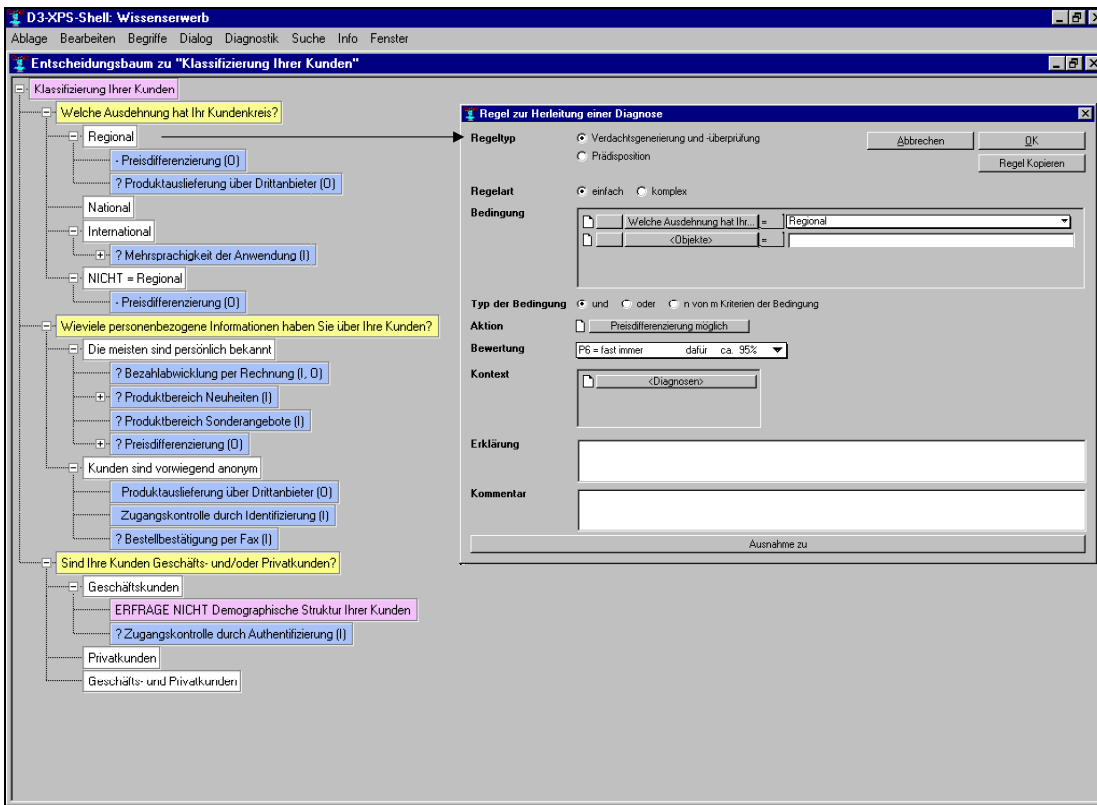


Abbildung 7-2: Entscheidungsbaum zur Analyse der Kundenstruktur

Intern werden die Inhalte der Wissensbasis in einem Objektsystem repräsentiert. Basis des D3-Systems ist die objektorientierte Programmiersprache OBJECT LISP. Frageklassen, Fragen, Merkmale und Lösungselemente eines Klassifikationsproblems werden als grafische Elemente in interaktiven Editoren eingegeben und mit Hilfe von Regeleditoren zu einem Regelwerk verknüpft.

Ursprünglich für Apple Macintosh-Systeme entwickelt, ist die Benutzungsoberfläche sehr einfach bedienbar. Die verschiedenen Objekttypen sind durch unterschiedliche Farben gekennzeichnet. So können Regelwerke mit hoher Komplexität in Entscheidungsbäumen anschaulich abgebildet und auf verschiedenen Abstraktionsebenen dargestellt werden. Abbildung 7-2 zeigt die oberen Ebenen des Entscheidungsbaumes zur Analyse der Kundenstruktur und den Editor zur Bearbeitung von Regeln zur Ableitung von Lösungselementen aus den ermittelten Merkmalen.

D3 besitzt eine Schnittstelle, über die das Expertensystem in Fremdsysteme eingebettet werden kann [PUPP98, S. 178]. Sie macht alle Objekte der Wissensbasis von außen zugänglich. Im IntelliShop-System wird diese Schnittstelle genutzt, um D3 mit den anderen Systemkomponenten des IntelliShops zu integrieren.

7.3 Anwendungsentwicklung mit dem CommerceNavigator

Der Anforderungsnavigator des IntelliShop-Systems wird als CommerceNavigator bezeichnet. Er bildet den in Kapitel 6 beschriebenen Adaptionleitfaden vollständig ab und integriert die dort angeführten Werkzeuge in einer homogenen Umgebung. Alle Funktionen des CommerceNavigators sind über das WWW mit einem marktüblichen Browser zu nutzen.

Die Benutzungsoberfläche des CommerceNavigators ist in zwei Bildschirmabschnitte geteilt (siehe Abbildung 7-3). Im linken Abschnitt befindet sich ein Strukturbaum, der alle Aktivitäten des Adaptionleitfadens aufführt. Er ist in Form eines Java-Applets implementiert. Optik, Funktionsweise und Bedienung sind analog zu der bekannter Strukturbäume aufgebaut, wie sie beispielsweise im Explorer des WINDOWS-NT-Systems verwendet werden. Der Strukturbaum gliedert die Aktivitäten des Adaptionleitfadens in mehrere Ebenen. Die Aktivitäten einer Ebene lassen sich nach Bedarf ein- und ausblenden, um den Strukturbaum den jeweils durchzuführenden Aufgaben anzupassen. Die erste Ebene des Strukturbaums entspricht den Phasen des Adaptionleitfadens. Bei Klick auf einen Eintrag der untergeordneten Ebenen werden die benötigten Werkzeuge ausgewählt und im rechten Abschnitt des Bildschirms dargestellt.

Durch Abarbeitung des Strukturbaums von oben nach unten wird der Anwender Schritt für Schritt von der Analyse und Konzeption bis zur vollständigen Realisierung einer eigenen Anwendung geführt. Er ist dabei nicht an eine streng sequentielle Vorgehensweise gebunden, denn durch die vollständige Integration aller Werkzeuge ist eine Überarbeitung durchgeführter Anpassungen jederzeit möglich.

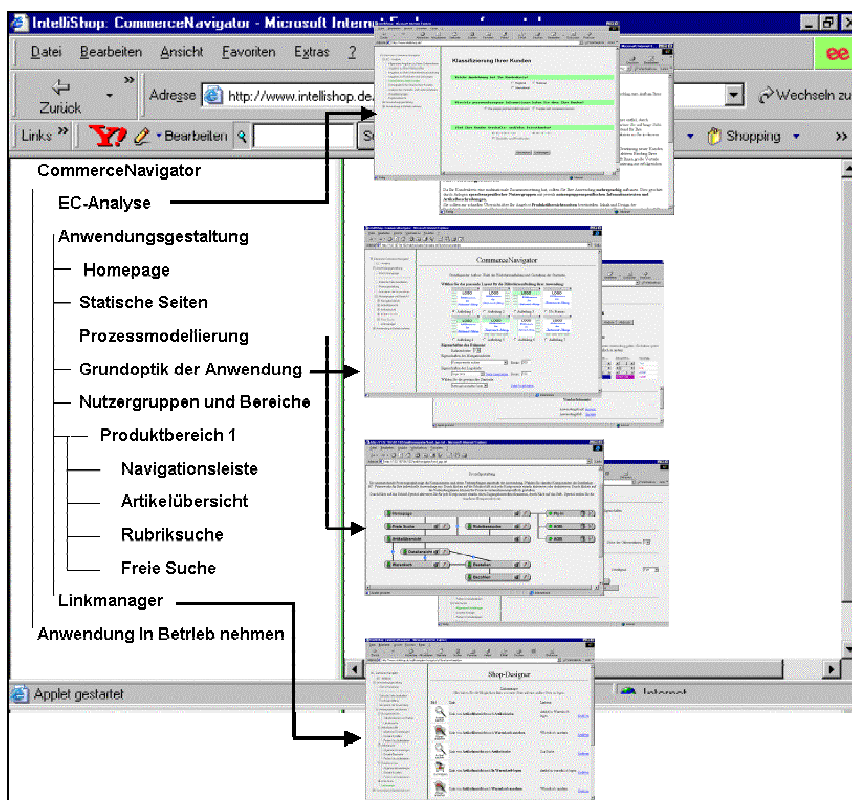


Abbildung 7-3: Aufbau des CommerceNavigators

Um die Funktionsweise der einzelnen Werkzeuge im CommerceNavigator zu veranschaulichen, werden sie in der Reihenfolge ihrer Anwendung bei der Entwicklung eines neuen Online-Shops vorgestellt.

7.3.1 EC-Analyse

Am Beginn jedes Projekts steht die EC-Analyse. Sie entspricht der wissensbasierten Analyse und Konzeption im Adaptionleitfad. Hier wird der Fragenkatalog der Wissensbasis schrittweise abgearbeitet. Öffnet ein Anwender den CommerceNavi-

gator, werden über die Schnittstelle zum Expertensystem alle Frageklassen der Wissensbasis ausgelesen und im Strukturbaum als Hyperlinks unterhalb der EC-Analyse angezeigt.

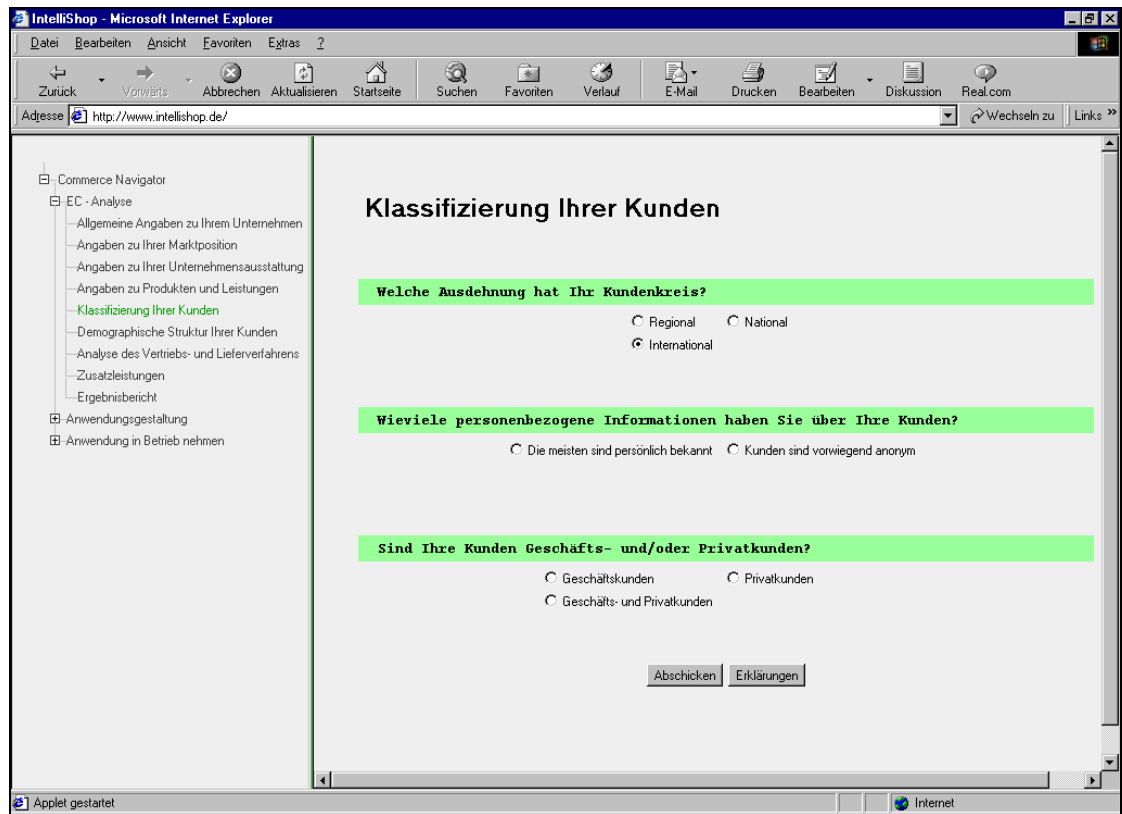


Abbildung 7-4: Analyse der Kundenstruktur in der EC-Analyse

In Abhängigkeit von den Angaben des Benutzers werden zu jeder Frageklasse die relevanten Fragestellungen ausgelesen und im rechten Bildschirmabschnitt dargestellt. Dabei ist es auch möglich, durch Aufruf einer Frageklasse aus dem Strukturbaum Antworten zu revidieren oder Fragen zu übergehen. Das Expertensystem reagiert in solchen Fällen dynamisch auf die Eingaben des Anwenders.

Für die Qualität der Angaben und die spätere Akzeptanz des Analyseergebnisses ist es wichtig, dass der Anwender unmittelbar bei der Abarbeitung des Fragenkatalogs die Bedeutung der Fragestellungen verstehen und auch mögliche Folgen seiner Antworten nachvollziehen kann. Deshalb ist zu jeder Frageklasse ein erklärender Text vorhanden, der die Fragestellung und mögliche Konsequenzen der Antwort erläutert. Die einzelnen Erklärungstexte sind als Hypertext in HTML verfasst, sodass komplexe

Zusammenhänge durch multimedial aufbereitete Informationen und Verknüpfung mit weiteren Quellen im Netz erklärt werden können. Im Normalfall wird die Folge der Fragestellung durch die Dialogsteuerung des D3-Systems determiniert. Der Benutzer hat aber auch die Möglichkeit, durch Auswahl einer Frageklasse im Strukturbaum Fragen zu einem bereits behandelten Problembereich erneut zu beantworten, was u. U. zu anderen Ergebnissen des Expertensystems führt.

Nach der Auswertung der Antworten gibt das Expertensystem eine Menge bestätigter Diagnosen aus. Sie entsprechen der Auswahl von Zielen, Maßnahmen und Softwarekomponenten, die für die Realisierung des Online-Shops empfohlen werden. Über die Schnittstelle des Expertensystems werden die Diagnosen ausgelesen und in Form einer Zeichenkette kodiert an den Application Server übergeben. So steht die Ergebnisdiagnosemenge allen anderen Werkzeugen zur Verfügung.

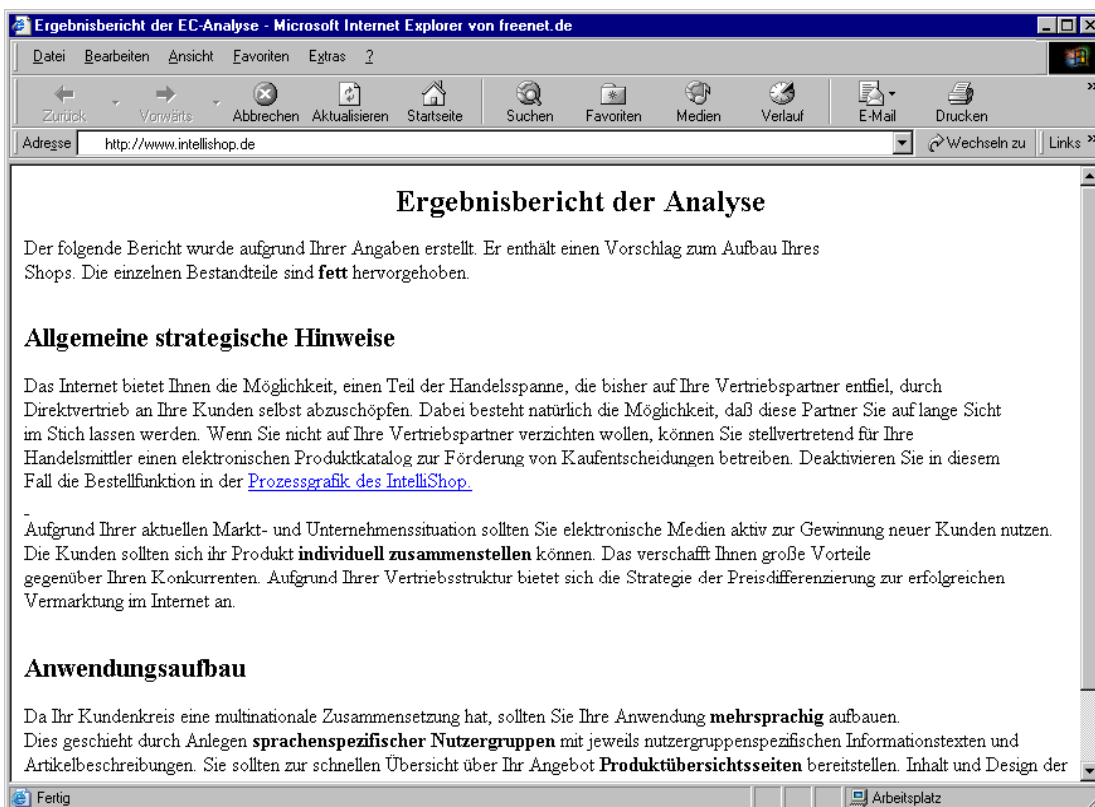


Abbildung 7-5: Ergebnisbericht der EC-Analyse

Aus dem kodierten Analyseergebnis wird am Ende jeder EC-Analyse der Ergebnisbericht erzeugt. Zur besseren Darstellung wurde für das IntelliShop-System eine eigene

Erklärungskomponente entwickelt, die das Analyseergebnis einliest und einen nach Themengebieten strukturierten Ergebnisbericht erzeugt. Dieser liefert Vorschläge für strategische Projektziele sowie ihre technische bzw. organisatorische Umsetzung (siehe Abbildung 7-5).

Weiterführende Arbeiten zeigen, dass sich der Ergebnisbericht durch Generierung eines XML-Dokuments aus den Ergebniselementen und durch Aufbereitung mit Style Sheets dynamisch an die Anforderungen unterschiedlicher Anwendergruppen im betrieblichen Einsatz anpassen lässt [SCHÖ00].

Akzeptiert der Anwender das Ergebnis der EC-Analyse, werden die Ergebniselemente in der relationalen Datenbank gespeichert und eine Verzeichnisstruktur zur Ablage der multimedialen Inhalte einer Anwendung im Dateisystem des IntelliShop-Servers angelegt. Alle Elemente, die Bestandteile des Zielsystems beschreiben, werden den entsprechenden Softwarekomponenten zugeordnet. Diese werden in das Verzeichnis *Benutzte_Komponenten* übernommen.

Mit der Auswahl der Komponenten ist der Funktionsumfang des Zielsystems individuell an die Anforderungen des Anwenders angepasst und in der Softwarebibliothek vollständig beschrieben.

7.3.2 Anwendungsgestaltung

Die Aktivitäten der Prozessmodellierung und des Designs der Benutzerschnittstelle sind im CommerceNavigator unter Anwendungsgestaltung zusammengefasst. Nach dem Adaptionleitfaden aus Kapitel 6 wird im Anschluss an die EC-Analyse ein geeignetes Prozessmodell und ein Strukturtemplate zur Definition der Bildschirmaufteilung ausgewählt und die entsprechenden Parameter in der Datenbank voreingestellt. Danach ist in einer rudimentären Version ein bereits voll funktionsfähiges System vorhanden, an dem jede weitere Anpassung unmittelbar überprüft werden kann.

Alle Werkzeuge, die zur Anwendungsgestaltung eingesetzt werden, haben jederzeit Zugriff auf die bereits eingestellten Adaptiondaten, sodass sie in beliebiger Reihenfolge eingesetzt werden können und jede weitere Anpassung durch den Anwender unmittelbar am funktionsfähigen System ausprobiert werden kann.

7.3.2.1 Anpassung der Anwendungsarchitektur

Der erste Schritt der Anwendungsgestaltung ist die genaue Definition der Anwendungsarchitektur. Sie geschieht durch Auswahl eines Strukturtemplates, das die Struktur der Benutzungsoberfläche auf dem Bildschirm definiert. Jedes Template wird durch eine eindeutige Nummer identifiziert, die in die Tabelle *Homepage* der Parameterdatenbank eingetragen wird.

Die Komponente *Homepage* liest diesen Parameter und die Referenz auf die zugehörigen Inhalte der Frames aus und erzeugt das entsprechende Frame-Set. Durch Menüauswahl können die Inhalte der Frames angepasst werden.

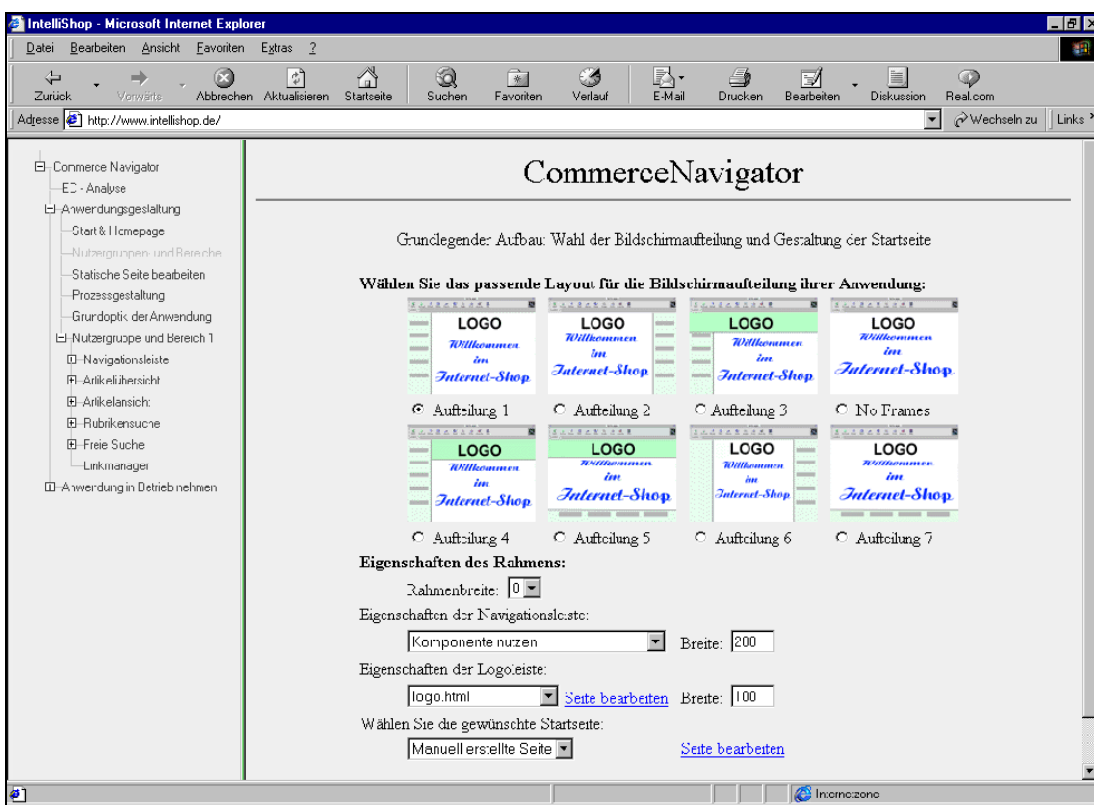


Abbildung 7-6: Auswahl eines Strukturtemplates

Hier ist auch die Möglichkeit gegeben, statt einer der Komponenten aus der Softwarebibliothek in einem Frame eine HTML-Seite aufzurufen, die vom Anwender selbst erstellt wurde. Über einen Hyperlink neben den Menüs zur Auswahl der Frame-Inhalte lässt sich der HTML-Editor des IntelliShops öffnen. Analog kann mit

diesem Editor HTML-Code bearbeitet und an allen dafür vorgesehenen Stellen in das System integriert werden.

7.3.2.2 Anpassende Modellierung im Prozesseditor

Im Anschluss an die erste Definition übergreifender Anwendungsstrukturen werden mit dem Prozesseditor die Prozesse innerhalb der Anwendung definiert. Der Prozesseditor ist als interaktives Java-Applet programmiert und bietet zahlreiche Funktionen für die grafische Prozessmodellierung. Die einzelnen Komponenten des vorgeschlagenen Referenzmodells werden als Symbole dargestellt. Mitteilungen, die zwischen den Komponenten ausgetauscht werden, sind durch Verbindungslinien symbolisiert.

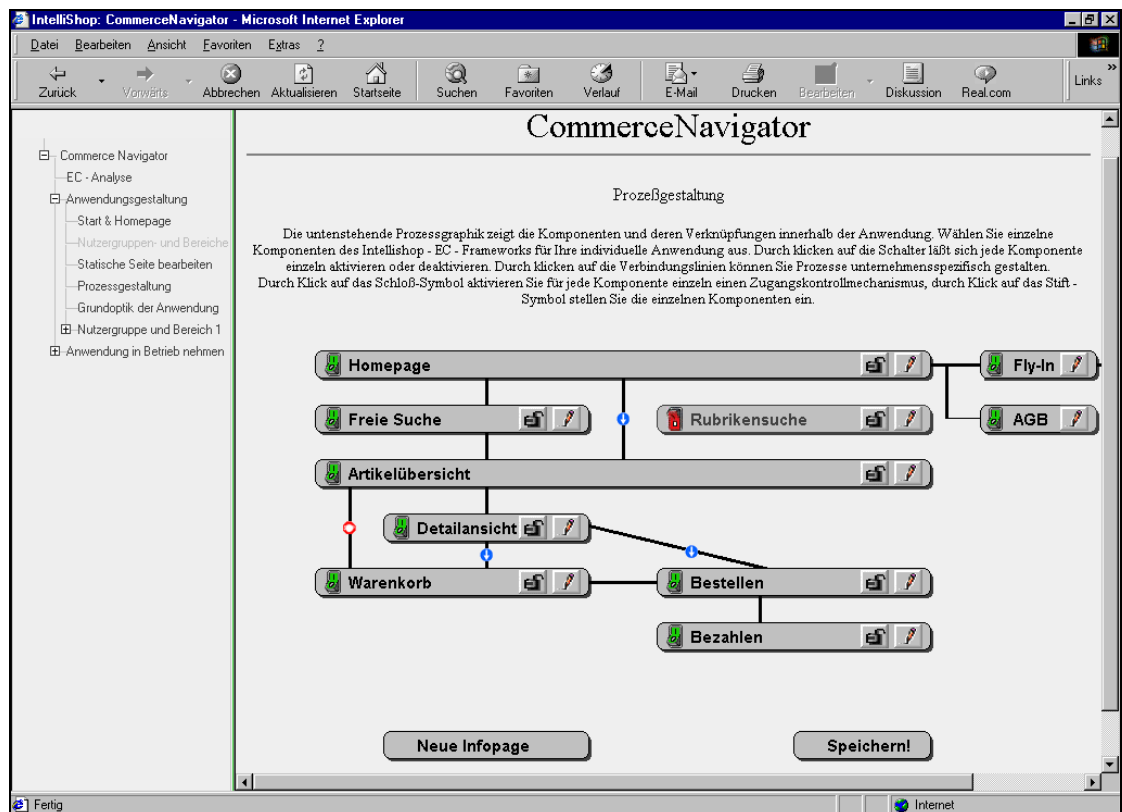


Abbildung 7-7: Prozesseditor

Der Modellvorschlag wird aufgrund der EC-Analyse ermittelt. Er enthält alle Softwarekomponenten, die als Teil des Lösungsvorschlages durch das Expertensystem ermittelt wurden. Mit der Funktion *Neue Infopage* des Prozesseditors können zusätzlich statische HTML-Seiten in das Modell integriert werden. Durch Anklicken

des Schaltersymbols können Komponenten im Modell aktiviert bzw. deaktiviert werden, wobei sich nur Komponenten deaktivieren lassen, die zuvor in der Konzeption nicht als zwingend erforderliches Element eingestuft wurden. Wird eine Komponente deaktiviert, ist der von ihr ausgeführte Teilprozess nicht mehr Bestandteil der Prozesskette. Durch das rote Schaltersymbol als inaktive Komponente gekennzeichnet, bleibt sie Bestandteil des Modells. In der Parameterverwaltung der Datenbank wird der inaktive Zustand einer Komponente lediglich durch Setzen des Wertes in der jeweiligen Strukturrelation vermerkt. Alle bereits eingestellten Parameter zur Konfiguration der Komponente bleiben erhalten. Bei dem Versuch, eine zwingend erforderliche Komponente zu deaktivieren, wird über die Schnittstelle zum Expertensystem aus der Erklärungskomponente des D3-Systems der Text zur Begründung, die im jeweiligen Fall zur Ableitung der Diagnose geführt hat [PUPP00, S. 35], ausgelesen und als kurze Meldung angezeigt. Durch Anklicken des Stift-Symbols gelangt der Anwender direkt zu dem Eigenschafteneditor einer Komponente.

Auch Verbindungen zwischen den einzelnen Komponenten lassen sich in ähnlicher Weise deaktivieren und wieder aktivieren. Dazu dient ein interaktives Symbol auf jeder Verbindungslinie, das anzeigt, ob noch eine Kommunikationsverbindung zwischen zwei Komponenten besteht (gekennzeichnet durch das Pfeil-Symbol) oder unterbunden wurde (gekennzeichnet durch das Halt-Symbol). Auch bei der Manipulation der Verbindungen zwischen den Komponenten wird die formale Richtigkeit des Modells durch ein internes Regelwerk überwacht, das sicherstellt, dass jede aktive Komponente mindestens auf eine Weise genutzt werden kann.

Die visuelle Darstellung des grafischen Prozesseditors liefert dem Anwender ein anschauliches Bild vom Aufbau seiner Anwendung. Die Interaktionsmechanismen des Prozesseditors erlauben ihm, interaktiv das Prozessmodell seinen individuellen Anforderungen anzupassen. Durch den ständigen Abgleich mit dem Ergebnis der expertensystembasierten Konzeption mit dem Prozesseditor ist sichergestellt, dass jede Anpassung stets zu einer betriebswirtschaftlich sinnvollen und formal richtigen Lösung führt. Statische HTML zur multimedialen Informationsvermittlung können mit marktüblichen Editoren entwickelt und neben den dialogorientierten Komponenten des Systems integriert werden.

7.3.2.3 Detailanpassung des Anwendungsdesigns

In den nächsten Schritten der Adaption werden die Elemente der Anwendung im Detail angepasst. Analog zu der hierarchischen Beschreibung in der Parameterverwaltung müssen zunächst die Grundeinstellungen der Anwendung selbst festgelegt werden. Wie alle grafischen Elemente werden sie im IntelliShop-System mit dem Shop-Designer definiert.

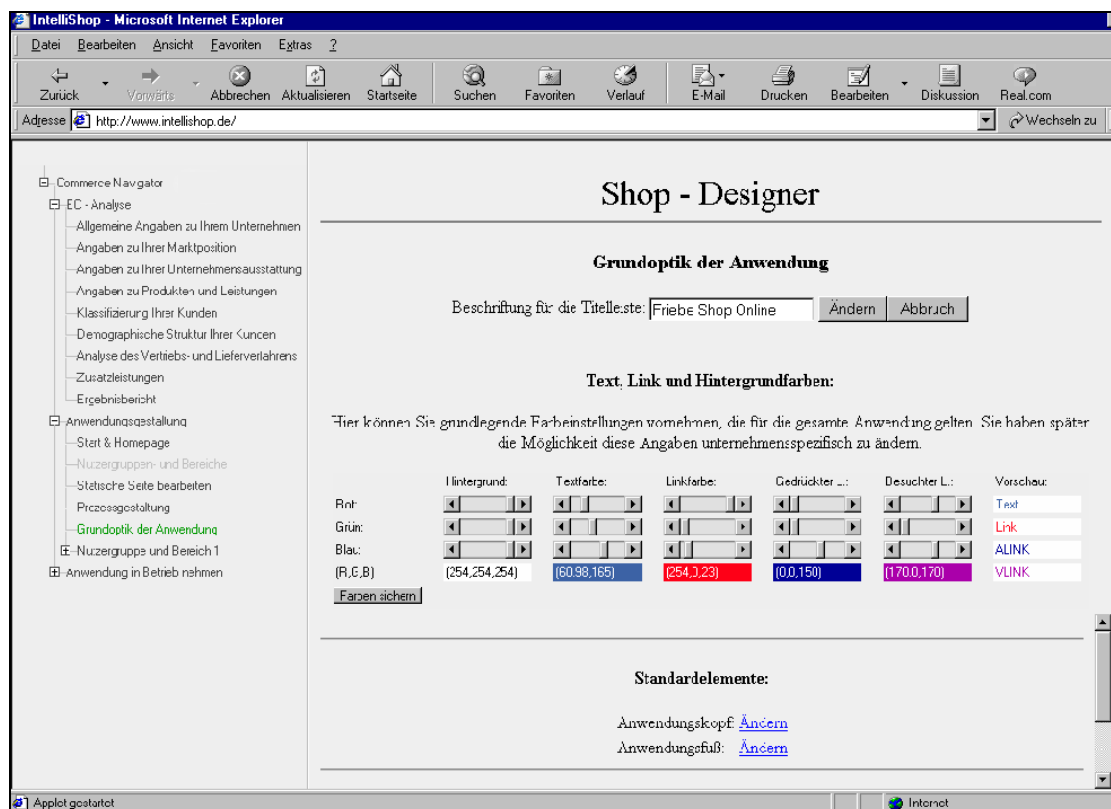


Abbildung 7-8: Definition des Corporative Designs der Anwendung

Dort können alle Parameter, die auf das Design der Anwendung Einfluss haben, eingestellt und vordefinierte grafische Elemente in das System integriert werden. Jede Anpassung wird im Editor sofort visualisiert und kann in einem kreativen Prozess schrittweise verbessert werden. Alle Parameterwerte werden bei der Übernahme des Ergebnisses der EC-Analyse mit vordefinierten Werten belegt, die in der Definition des Datenbankschemas der relationalen Datenbank festgelegt sind. Farben werden analog zur Definition von Farbwerten im HTML-Standard in der Datenbank als RGB-Werte abgelegt. Im Shop-Designer werden diese Werte wieder in Farben

umgesetzt und lassen sich dort über Schieberegler interaktiv anpassen. Komplexere, individuell entwickelte Designelemente können mit dem integrierten HTML-Editor des IntelliShop-Systems direkt bearbeitet werden.

7.3.2.4 Anpassung einzelner Komponenten

Im Strukturbaum des CommerceNavigators sind alle Komponenten verzeichnet, die in der EC-Analyse ausgewählt wurden. Dabei ist ein Mehrfacheinsatz derselben Komponente in unterschiedlicher Parametrisierung möglich. Wird z. B. empfohlen, die Anwendung in mehrere Nutzergruppen zu strukturieren, können dieselben Komponenten mehrfach in jeweils unterschiedlichen Nutzergruppen aufgeführt sein. Die Eigenschaften der eingesetzten Komponenten werden mit Hilfe komponentenspezifischer Editoren definiert. Analog zu der Trennung der Komponentenparameter in eine Struktur- und eine Elementerelation werden im IntelliShop-System die Eigenschaften einer Komponente in zwei Schritten angepasst. Der Aufbau und die Funktionsweise der Eigenschafteneditoren werden am Beispiel der Komponente *Produktsuche* erläutert. Die entsprechenden Struktur- und Elementerelationen wurden in Kapitel 5 beispielhaft für die parametrisierbare Abbildung von Komponenten in der Parameterverwaltung dargestellt.

Mit dem Eigenschafteneditor in Abbildung 7-9 lassen sich alle Parameter der Komponentenkopfrelation einstellen. Soweit die Parameter nur vordefinierte Werte annehmen dürfen, werden diese durch Pull-Down-Menüs vorgegeben, um Eingabefehler von vorneherein auszuschließen. In der Komponentenstrukturrelation werden u. a. Einstellungen definiert, die Auswirkungen auf die Darstellung aller Elemente haben (z. B. tabellarische Darstellung, Anzeige von Feldbenennungen). Durch die Zusammenfassung von Eigenschaften, die für alle Elemente einer Komponente identisch sein sollten, wird die Zahl der einzustellenden und vom System bei jedem Zugriff auf die Komponente zu verarbeitenden Parameter deutlich reduziert.

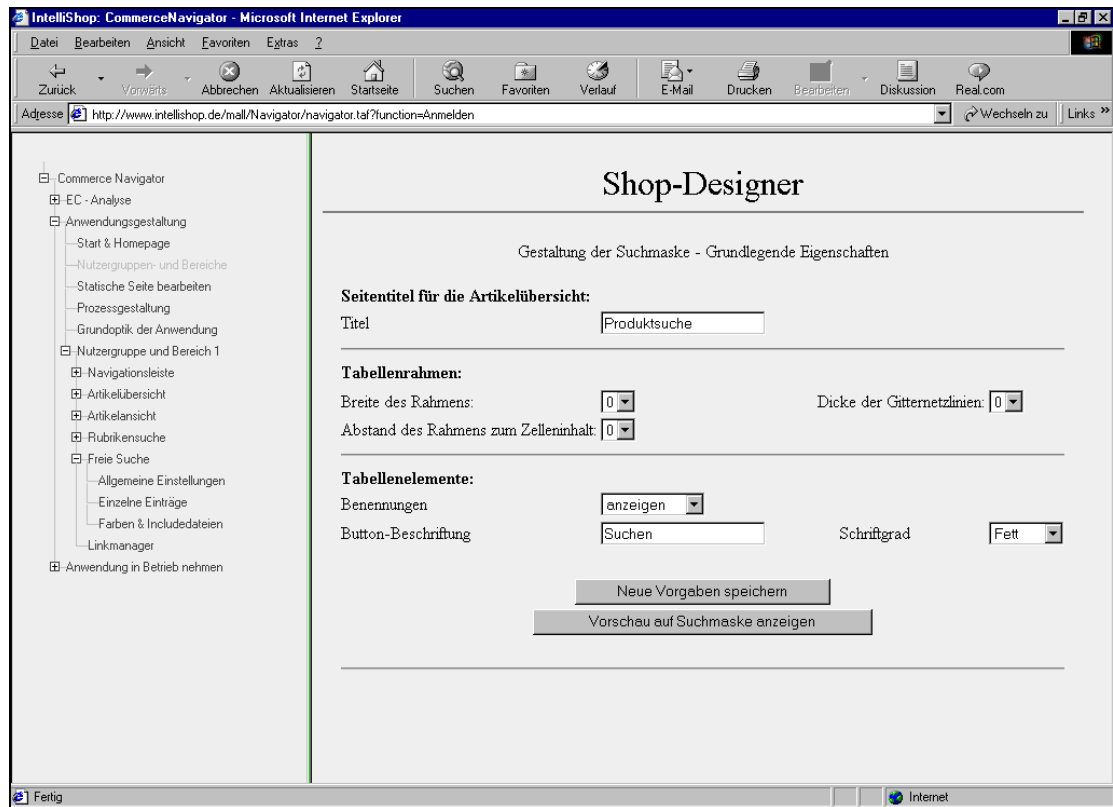


Abbildung 7-9: Parametrisierung der Komponentenstrukturrelation am Beispiel der Produktsuche

Zur Anpassung von Farbwerten und zur Integration von individuell entwickeltem HTML-Code wird der Shop-Designer bzw. der HTML-Editor des IntelliShop-Systems eingesetzt. Dabei werden alle Angaben automatisch übernommen, die auf Anwendungsebene bereits festgelegt wurden. Sie können bei Bedarf durch Parametrisierung einzelner Komponenten überschrieben werden. Diese einfache Abbildung von Vererbung von Eigenschaften in einer Hierarchie führt nur zu einer weiteren Reduktion des Parametrisierungsaufwandes und stellt automatisch sicher, dass definierte Designrichtlinien durchgängig verwendet werden. Kurzfristige Modifikationen dieser Richtlinien, wie sie in der Praxis besonders bei Erstentwicklungen häufig vorkommen, werden automatisch innerhalb der gesamten Anwendung umgesetzt.

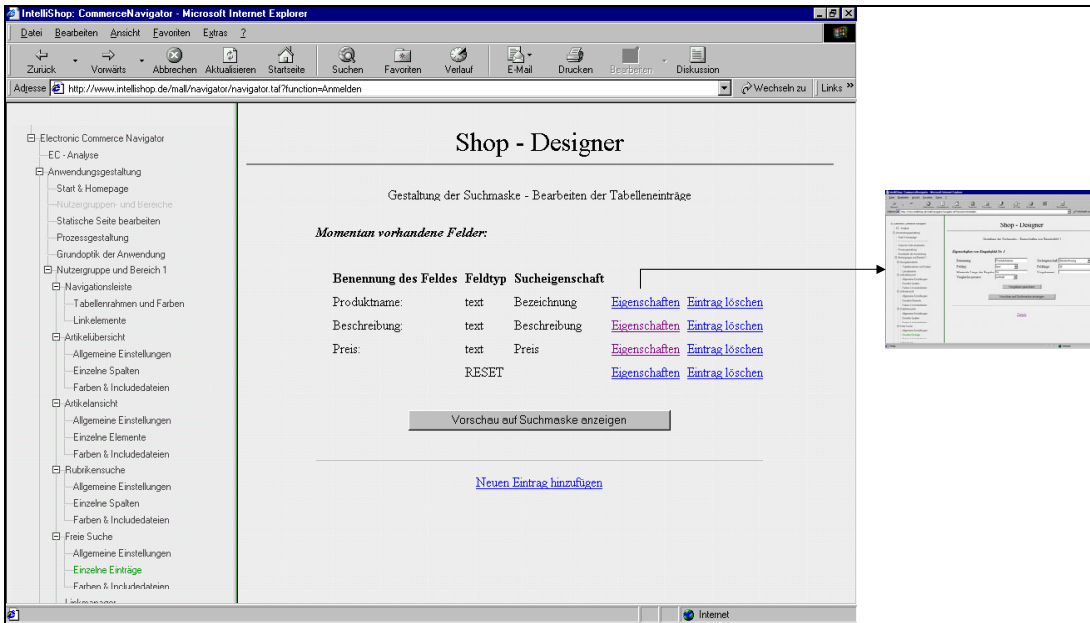


Abbildung 7-10: Parametrisierung der Elementerelation am Beispiel der Produktsuche

Die einzelnen Formularelemente der Eingabemaske werden in einem zweiten Editor bearbeitet. Dieser Editor setzt unmittelbar auf der Elementerelation der Komponente *Produktsuche* auf. Abbildung 7-10 zeigt, dass aufgrund der Datenstruktur der Parameterverwaltung eine beliebige Erweiterung des Suchformulars in einem sehr einfachen zweistufigen Verfahren möglich wird.

7.3.2.5 Verlinken von Komponenten

Einer der wichtigsten Vorteile des Web ist die Möglichkeit, über Hyperlinks Inhalte beliebig miteinander verknüpfen zu können. Allerdings werden Hypertextstrukturen schon bei geringem Umfang schnell unübersichtlich. Kommt es im Laufe des Projektes zu umfangreichen Modifikationen des Gesamtsystems, kann dadurch ein erheblicher Aufwand bei der Anpassung der Hypertextstruktur entstehen. Im IntelliShop-System kann die Verwaltung der Hypertextstruktur durch ein einfaches Werkzeug wirkungsvoll unterstützt werden.

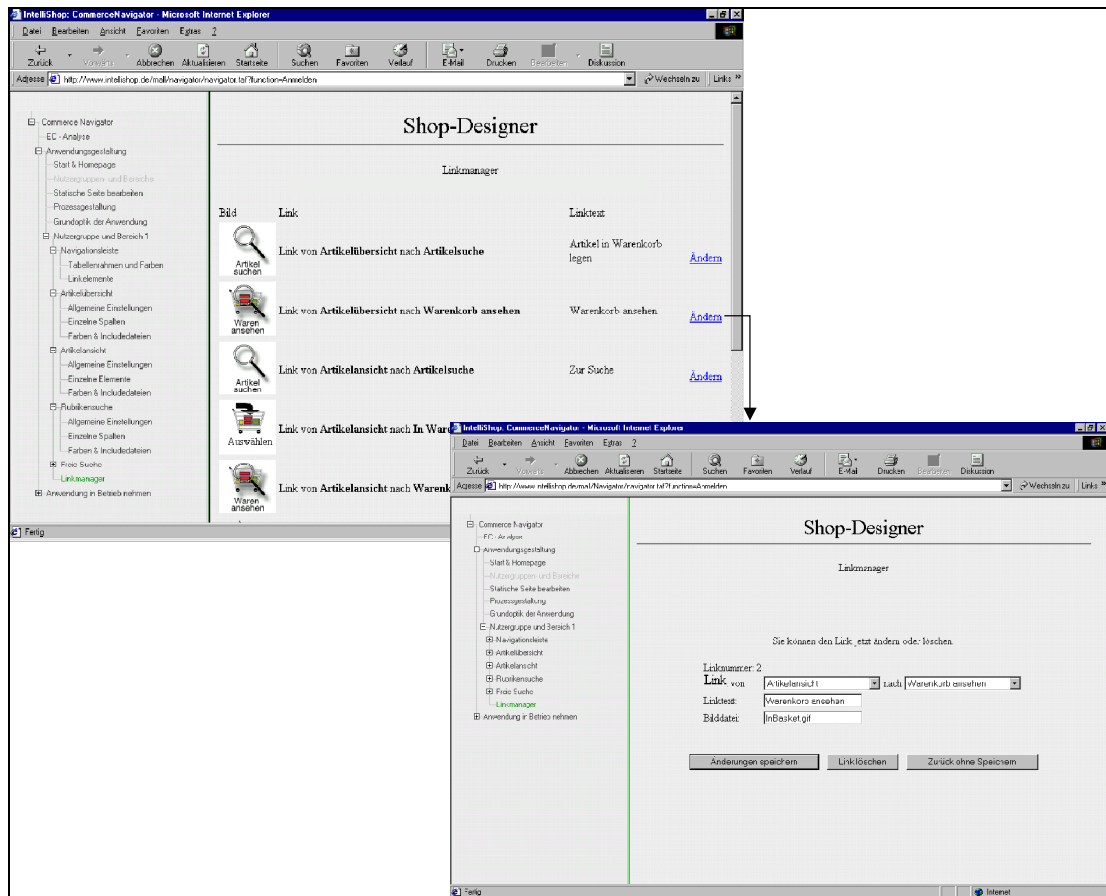


Abbildung 7-11: Linkmanager zur Verwaltung von Hyperlinks zwischen Komponenten

Der Linkmanager erzeugt eine Übersicht aller Hyperlinks zwischen den Komponenten einer Anwendung. Die Informationen darüber, welche Komponenten miteinander verknüpft sind, werden in der Tabelle *Links* der Parameterverwaltung abgelegt, wobei dort nur die Komponenten-IDs aufgeführt sind. Die Bezeichnungen der Komponenten werden zur Anzeige im Linkmanager durch eine Abfrage auf die Tabellen *Benutzte_Komponenten* und *Vorhandene_Komponenten* ermittelt. Wird eine Komponente aufgrund einer Änderung im Prozessmodell nicht mehr eingesetzt, werden im Linkmanager alle Verknüpfungen zu ihr deaktiviert.

In dem Formular, das im Linkmanager zum Anlegen bzw. Ändern eines Links verwendet wird, ist die Auswahl der verknüpfbaren Komponenten von vornherein auf jene beschränkt, die in der Tabelle *Benutzte_Komponenten* verzeichnet sind. Fehler bei der Verwaltung von Hyperlinks sind so weitgehend ausgeschlossen.

7.3.3 Inbetriebnahme

Die Inbetriebnahme fasst im IntelliShop-System alle Aktivitäten der Datenübernahme und Registrierung der erstellten Anwendung zusammen. Die Registrierung eines Domain-Namens im Internet lässt sich ohne direkte Anbindung an die Domain-Verwaltung eines Netzbetreibers nur begrenzt unterstützen. Im IntelliShop-System wird deshalb lediglich die zu registrierende Anwendung beschrieben und auf der Leitseite der IntelliShop-Mall veröffentlicht.

Eine weitere wichtige Aufgabe bei der Inbetriebnahme jedes Online-Systems ist die Registrierung und anschließende Vermarktung eines zum Inhalt der Anwendung passenden Domain-Namens. Von den Institutionen, die Domain-Namen verwalten, wird bisher noch kein vollautomatisches Verfahren zur Anmeldung, Prüfung, Freischaltung und Abrechnung einer Domain-Registrierung angeboten, sodass vom IntelliShop-System lediglich Hinweise darauf gegeben werden können, was bei der Domain-Registrierung zu beachten ist. Daneben wird mit einem Hyperlink unmittelbar auf den Anbieterzugang des Bereichs *Verwaltung* verwiesen. In der Produktverwaltung des IntelliShops befinden sich alle für die Inbetriebnahme wichtigen Funktionen zum Import von Massendaten sowie Masken zur Anlage produktspezifischer Datensätze und Integration multimedialer Informationsquellen (siehe Abbildung 7-12).

7.4 Anwendungsbetrieb und Erfolgskontrolle

Alle Aktivitäten des Anwendungsbetriebes werden im IntelliShop-System im Bereich *Verwaltung* durchgeführt. Dieser ist vom CommerceNavigator getrennt und mit einem separaten Passwortschutz versehen, denn in vielen Fällen werden die im Betrieb der Anwendung anfallenden Aufgaben von Mitarbeitern mit speziellen Fachkenntnissen durchgeführt.

Im IntelliShop-System hat ein Anbieter grundsätzlich die Möglichkeit, beliebig viele Anwendungen zu betreiben. Dies wird genutzt, um mehrere Anwendungen für unterschiedliche Zielgruppen aufzubauen, oder um nach der Einführung eines Systems mit weiteren Optionen des IntelliShop-Systems zu experimentieren, während die erste in

Betrieb bleibt. Durch die einheitliche Datenstruktur aller Anwendungen ist eine Datenübernahme von der Produktivianwendung in das neue System (Staging) sehr einfach möglich.

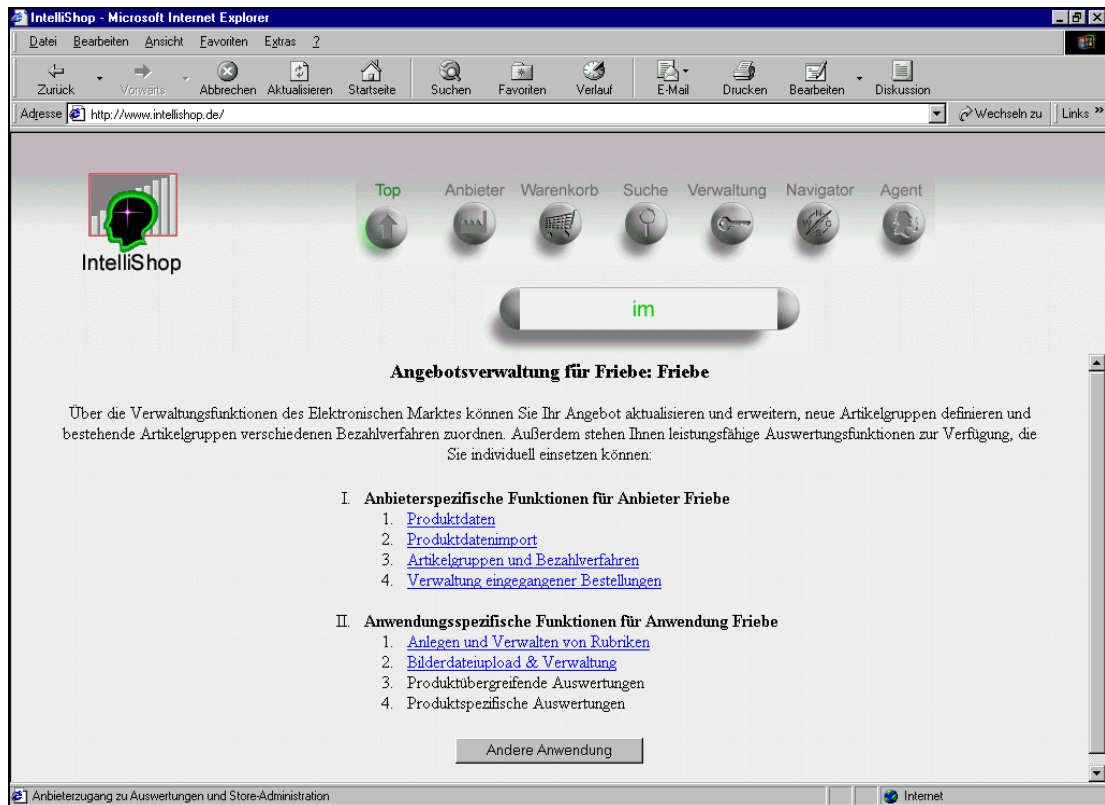


Abbildung 7-12: Produktverwaltung im IntelliShop

Das Passwort des Verwalters verschafft Zugang zu allen Anwendungen eines Anbieters. Innerhalb des Verwaltungsbereichs befinden sich die wesentlichen Funktionen zur Administration eines Online-Shops. Das IntelliShop-System besitzt leistungsfähige Funktionen für Import und Bearbeitung von Produktdaten. Eine besondere Eigenschaft des IntelliShop-Systems, die sich aus dem Aufbau der Datenstruktur ergibt, ist die Möglichkeit, zu einem Artikel beliebig viele Darstellungsformen zu definieren. Die Tabellen, die diese Daten speichern, sind mit der Tabelle *Nutzergruppen* verknüpft, sodass eine nutzergruppenspezifische Artikeldarstellung möglich ist.

Neben der Pflege von Produktdaten bietet der Verwaltungsbereich auch Funktionen zur Bearbeitung und Auswertung von Transaktionsdaten. Aufgrund seines Aufbaus

als Mall-System und der einheitlich generischen Datenstruktur über alle Anwendungen hinweg bietet IntelliShop die Möglichkeit, Abfragen über das gesamte Angebot aller Anbieter zu formulieren. In Verbindung damit ist auch die Auswertung der Adaptionparameter möglich, die während der Anwendungsentwicklung im CommerceNavigator erfasst werden. Sie liefern Informationen darüber, welche Verfahren, Technologien und Gestaltungselemente die Anbieter im IntelliShop-System längerfristig mit Erfolg einsetzen. Die Auswertungsergebnisse können zur Überprüfung und Verbesserung der Empfehlungen der Wissensbasis herangezogen werden.

8 Erfahrungsbericht und Ausblick

Die vorliegende Arbeit ist die Darstellung der Ergebnisse mehrjähriger Forschungsarbeit und eines in diesem Zusammenhang entwickelten und in der Praxis erprobten Systems. Vor dem Hintergrund, dass die Entwicklung im Bereich des Electronic Commerce einerseits ständig fortschreitet, sich aber die Einschätzung der Potentiale und Anforderungen in der jüngsten Vergangenheit sehr geändert haben, soll abschließend eine Wertung der erreichten Ziele und eine Einordnung der Ergebnisse sowie eine Bewertung der neueren Entwicklungen am Markt seit der Umsetzung der beschriebenen Konzepte erfolgen.

8.1 Kritische Würdigung der Ziele

Das wesentliche angestrebte Ziel der Arbeit war die Entwicklung eines Werkzeugs, das KMU in die Lage versetzt, selbstständig Lösungen für elektronische Märkte zu entwickeln, die ihren individuellen Anforderungen entsprechen. Ergebnis ist das IntelliShop-System. Es wird in der aktuellen Version seit mehreren Jahren von mittelständischen Unternehmen eingesetzt. Die Firma Friebe Luftfahrtbedarf GmbH betreibt seit 1998 einen Online-Shop auf Basis des IntelliShop-Systems zum Vertrieb von Zubehör und Accessoires für Piloten. Das Produktangebot umfasst rund 3.000 Artikel. Daneben wurde der IntelliShop im Rahmen der Aufgabenstellung des MECK (Mainfränkisches Electronic Commerce Kompetenzzentrum) in einer Reihe von Projekten eingesetzt. Damit konnte die Praxistauglichkeit des Ansatzes nachgewiesen werden.

In den durchgeführten Projekten treten im Vergleich zu typischen Online-Shop-Entwicklungen betriebswirtschaftliche Fragestellungen gegenüber technischen Problemen in den Vordergrund. Es zeigte sich, dass der Ergebnisbericht der EC-Analyse und die daraus resultierenden unmittelbaren Einstellungen im System wichtige Funktionen erfüllen. Sie trägt wesentlich zur Motivation gerade technisch unerfahrener Projektpartner bei. Diese vermerken positiv, dass die individuellen Rahmenbedingungen ihres Betriebes ganz offensichtlich vom IntelliShop-System berück-

sichtigt und sofort in eine funktionierende Lösung umgesetzt werden. Schon die Erläuterungen zu den einzelnen Fragestellungen in der Analyse und die Darstellung des Ergebnisses im Analysebericht fördern wesentlich das Verständnis für die Problemstellungen und Potentiale im Umfeld elektronischer Märkte und führen zu realistischen Erwartungen an das Projekt. Die vollständige Integration und der endbenutzergerechte Aufbau der Adaptionswerkzeuge erlauben es, Ergebnisse spontan an einem funktionierenden System auszuprobieren und wenn nötig zu revidieren bzw. zu verbessern. Dadurch wird die Hemmschwelle für den Einstieg der Anbieter in die Nutzung der elektronischen Geschäftsabwicklung erheblich gesenkt. Dies führt sehr schnell zu einer aktiven Teilnahme der Auftraggeber am Projektverlauf. Nach kurzer Einweisung sind sie in der Lage, mit Hilfe des CommerceNavigators einen Großteil der Entwicklung selbst zu übernehmen. Dabei wird die sonst übliche langwierige und abstrakte Systemkonzeption durch einfaches Ausprobieren ersetzt. In allen Projekten konnte so die Erstentwicklung der Systeme bereits nach wenigen Tagen fertiggestellt werden. Dadurch ist die Lösung gerade für die Zielgruppe der KMU besonders geeignet und wirtschaftlich sehr attraktiv. Die durchgeführten Projekte zeigten jedoch auch, dass für einen noch breiteren Einsatz des IntelliShop-Systems ein umfangreicher Vertriebs-, Wartungs- und Kundenbetreuungsaufwand notwendig wäre, der im Rahmen eines Forschungsprojektes nicht zu leisten ist.

Aus der Entwicklung und der Anwendung des IntelliShop-Systems konnten auch zahlreiche neue Erkenntnisse für die Bewertung und Auswahl von Werkzeugen zum Aufbau elektronischer Märkte gewonnen werden. Gleichzeitig wurde das System erfolgreich in Beratungsprojekten eingesetzt, um Lösungsideen und -alternativen schnell zu erarbeiten und am funktionsfähigen Prototyp sofort zu testen. Umgekehrt ließen sich aber auch neu gewonnene Erkenntnisse aus Beratungsprojekten relativ einfach in der Wissensbasis des IntelliShop-Systems strukturiert ablegen und über den CommerceNavigator im Internet bereitstellen. Aus technischer Sicht konnten darüber hinaus wertvolle Erfahrungen bezüglich der Konzeption und Entwicklung generischer Standardanwendungssoftwarelösungen und neuer Adaptionswerkzeuge gewonnen werden.

8.2 Einordnung der Ergebnisse

Der vorgestellte Ansatz ist keine universelle Lösung für alle Aufgabenstellungen im Umfeld elektronischer Märkte. Der Einsatz auch des IntelliShops stellt keine unmittelbare Konkurrenz zur Individualentwicklung elektronischer Marktsysteme dar. Wo es um die Realisierung neuartiger Ideen oder die Nutzung technischer Innovationen geht, ist eine Eigenentwicklung oftmals trotz des höheren Aufwandes und größerer Risiken der richtige Ansatz, um die gewünschten Resultate zu erzielen. Unabdingbare Voraussetzung für den Erfolg eines solchen Projekts ist die Auswahl und der Einsatz geeigneter Methoden und Werkzeuge. Die relevanten Vor- und Nachteile sowohl der verfügbaren Werkzeuge als auch der existierenden methodischen Ansätze wurden im Rahmen dieser Arbeit aufgezeigt. Dabei wurden deutliche Vorteile der Verwendung von Standardsoftware sichtbar.

Die hier entwickelte Lösung schafft die Verbindung der Vorteile funktionsorientierter und betriebswirtschaftlicher Standardsoftwarelösungen über das wissensbasierte Einführungswerkzeug, das individuell auf Anforderungen reagiert. Dies macht sie im Vergleich zu den immer noch weitgehend technisch orientierten marktgängigen Lösungen deutlich flexibler und führt darüber hinaus zur Senkung von Beratungskosten.

Weitergehende Entwicklungsansätze ergeben sich aus den Potentialen der Wissensbasis und der Adaptionfähigkeit der Lösung. So lassen sich aus dem längerfristigen Betrieb mehrerer Anwendungen nicht nur Informationen über die Nutzung im Sinne der üblichen Erfolgskontrolle gewinnen. Da alle Analyseergebnisse im System abgelegt sind, ist eine umfassende Verifikation der Empfehlungen des Expertensystems im Nachhinein möglich. Hier wäre zu untersuchen, inwieweit die Überprüfung und Bewertung von Regeln zur gezielten Verbesserung der Wissensbasis methodisch unterstützt werden können. Auf der Grundlage von Erfahrungswerten aus der Einführung und dem Betrieb einer größeren Anzahl an Lösungen könnten Kennzahlen gewonnen werden, die eine Bewertung des Lösungsvorschlags nach Einführungs- und Umsetzungsaufwand ermöglichen.

Durch die Adaptionfähigkeit im laufenden Betrieb bietet das IntelliShop-System überdies grundsätzlich die Möglichkeit, Lösungen dynamisch an spontane Kundenanforderungen anzupassen. Dies wurde bereits im Rahmen der Realisierung eines ein-

fachen Agentensystems genutzt [KNÜP98, S. 25]. Der Agent durchsucht das Angebot aller Anbieter im IntelliShop, um kurzfristig Sortimente nach den angegebenen Präferenzen des Nutzers zusammenzustellen. Sowohl die Angabe der Präferenzen als auch die Präsentation der Produktauswahl lassen sich durch eine Vielzahl an Parametereinstellungen variieren. Weitere interessante Entwicklungen in diese Richtung sollten durch Nutzung der Inferenzmechanismen der Wissensbasis möglich sein.

8.3 Bewertung aktueller Entwicklungen

Nachdem sich die zunächst teilweise sehr hohen Erwartungen vor allem in den Medien ins Gegenteil verkehrten, sind zur Zeit in der Fachwelt zahlreiche Aktivitäten gerade im Bereich der Marktsysteme zu erkennen. Dabei steht vor dem Hintergrund negativer Erfahrungen jetzt der Kosten-Nutzen-Aspekt stärker im Vordergrund. Die Bedeutung einer eigenen Electronic Commerce-Strategie und die Notwendigkeit, Geschäftsprozesse an neue Anforderungen anzupassen und in gewachsene Unternehmensstrukturen zu integrieren, wird vielfach erkannt. Großunternehmen bewältigen diese Anforderungen in aufwändigen Projekten, häufig unter Einsatz umfangreicher und flexibel anpassbarer Standardsoftwarepakete. Hier bestätigt sich der in Kapitel 3 aufgezeigte Trend zum Zusammenwachsen von funktionsorientierter und betriebswirtschaftlicher Standardsoftware.

Im Segment der weniger umfangreichen Lösungen hat sich eine Reihe namhafter OpenSource-Lösungen etabliert. Technische Basis ist vielfach PHP, eine Technologie, die auch von vielen Service Providern als Plattform angeboten wird. Die frei verfügbaren Ressourcen im Internet in diesem Bereich sind nahezu unerschöpflich. Module zur Datenintegration auf Basis von XML finden sich ebenso wie fertige Scripten zur unternehmensübergreifenden Integration von Web Services. Damit hat sich für KMU eine kostengünstige Technologie etabliert. Um diese nutzen zu können, ist dennoch vielfach Hilfe von Außen nötig.

Anhang

Das D3-System erlaubt eine anschauliche grafische Darstellung der wichtigsten Elemente einer Wissensbasis. Eine Darstellung komplexer Regeln ist nur kontextsensitiv möglich. Die nachfolgenden Abbildungen zeigen den Fragenkatalog und die Entscheidungsbäume zur Ableitung von Diagnosen aus der Wissensbasis des Intelli-Shop-Systems. Die Diagnosehierarchie ist bereits in Abbildung 6-3, S. 128, dargestellt.

A1 Grafische Darstellung des Fragenkatalogs

Der Fragenkatalog ist in Frageklassen und die ihnen untergeordneten Fragen gegliedert.

A1.1 Frageklassenhierarchie

In der Frageklassenhierarchie werden alle Frageklassen Oberklassen zugeordnet. Die Oberklassen entsprechen den fünf Stufen des Fragenkatalogs. Jede Unterklasse repräsentiert ein Template mit Fragen zu einem Themenkomplex, der auf dieser Stufe untersucht wird.

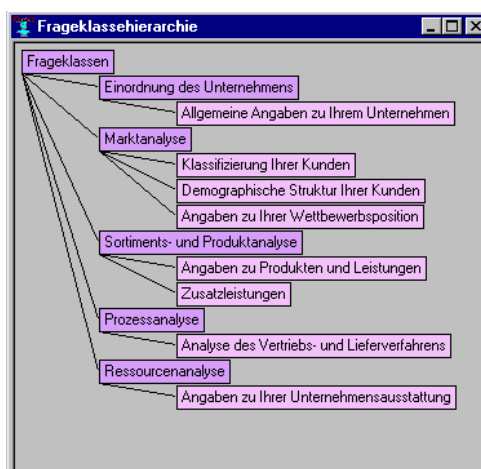


Abbildung A-1: Frageklassenhierarchie

A1.1 Fragehierarchie

In der Fragehierarchie werden alle Fragen ihren Frageklassen zugeordnet dargestellt.

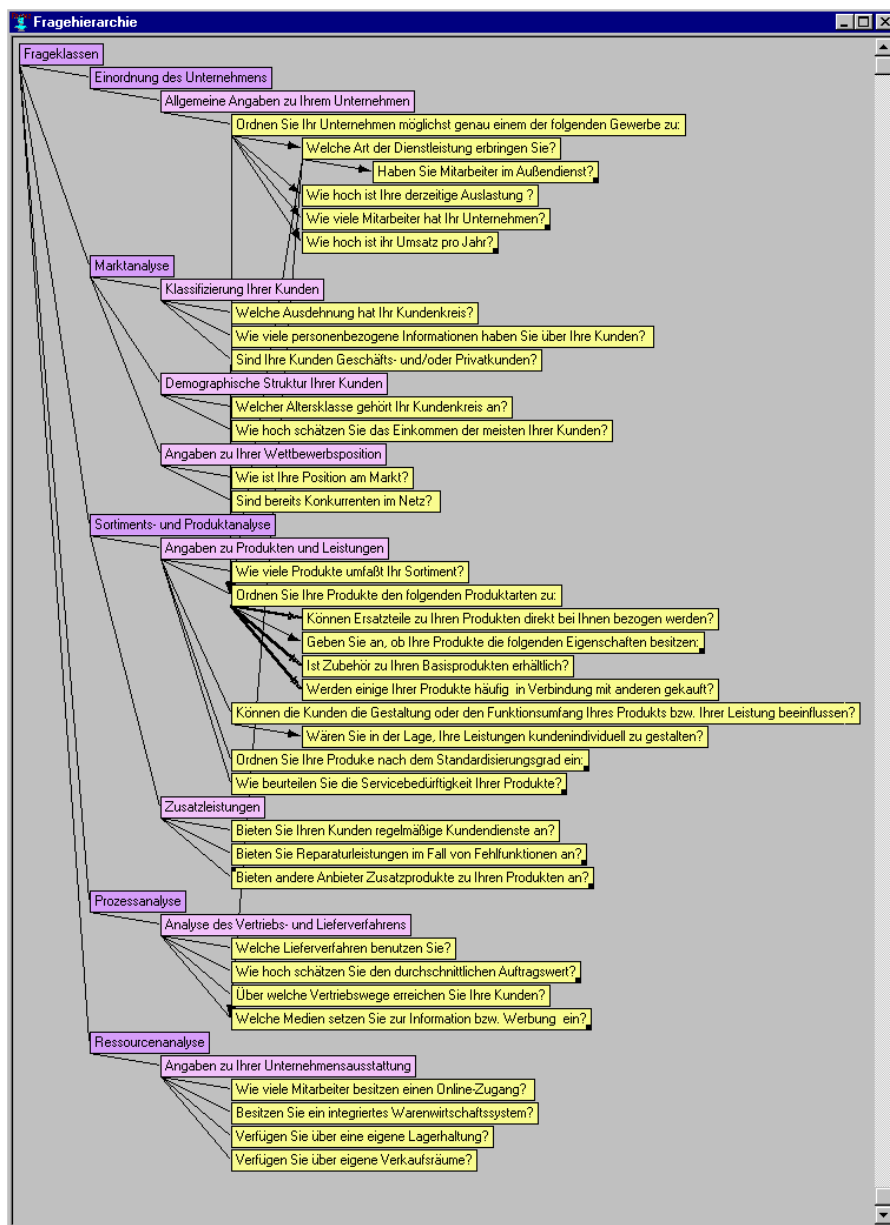


Abbildung A-2: Frageklassenhierarchie

Verknüpfungen machen Beziehungen zwischen Fragen deutlich. Folgefragen werden durch Verknüpfungen ihren übergeordneten Fragen unmittelbar untergeordnet. Jeder dieser Verknüpfungen kann eine Regel zugeordnet werden.

A2 Beispiel für einen wissensbasierten Dialog

Um die Arbeitsweise des wissensbasierten Systems zu veranschaulichen, sollen der Inferenzprozess zur Ableitung eines Analyseergebnisses und die Konsequenzen für die Parametrisierung der Softwarebibliothek anhand eines Teilausschnitts aus einer Analyse veranschaulicht werden. Um den Ablauf einzelner Hintergrundprozesse zeigen zu können, erfolgt die Darstellung direkt in der Expertensystem-Shell des D3-Systems. Dort muss die Beantwortung jeder Frage einzeln bestätigt werden (siehe Abbildung A-3) und wird sofort ausgewertet.

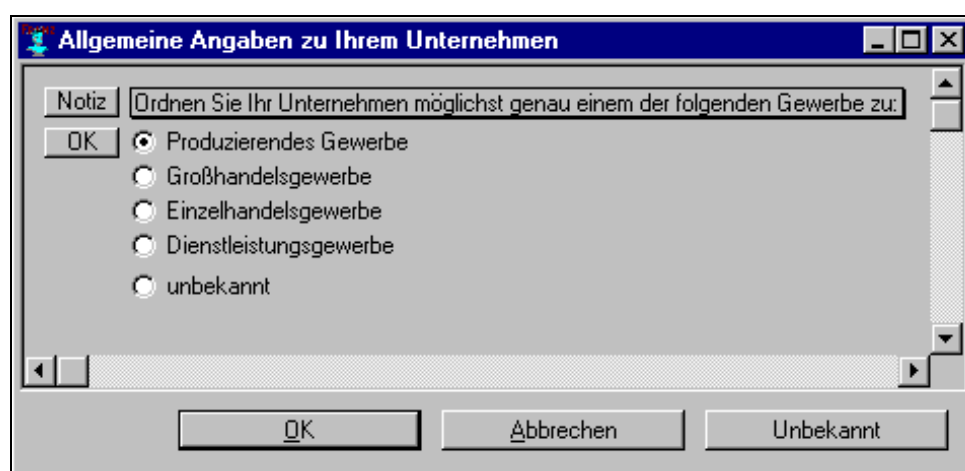


Abbildung A-3: Frageklasse „Allgemeine Angaben zu Ihrem Unternehmen“ im D3-System

So können auch die Auswirkungen der Antworten auf einzelne Fragen verdeutlicht werden, und eine anschauliche Darstellung von Zwischenergebnissen anhand von Begründungsgraphen ist möglich. Unter der vereinfachten Benutzungsoberfläche der EC-Analyse im CommerceNavigator des Intellishop-Systems kann jeweils immer nur eine Frageklasse als Ganzes bearbeitet werden, um die Konzentration des Anwenders auf den jeweiligen Themenbereich zu sichern (siehe Abbildung 7-4, S. 151).

Nachfolgend wird der Inferenzprozess am Beispiel der Analyse eines kleinen Produktionsunternehmens dargestellt. Der in Abbildung 7-5, S. 152 dargestellte Ergebnisbericht ist dagegen das Resultat der Analyse eines Handelsunternehmens. So kann deutlich gemacht werden, wie das System auf unterschiedliche Angaben reagiert.

Am Beginn der Abarbeitung des Fragenkatalogs stehen bei jeder Analyse zunächst die Fragen zur Einordnung des Unternehmens (siehe Abbildung 6-2, S. 116). Wird das Unternehmen im produzierenden Gewerbe eingestuft, führt dies in der Folge zur Bewertung der Unternehmensgröße nicht nach dem Umsatz, wie dies für ein Handelsunternehmen sinnvoll wäre, sondern nach der Mitarbeiterzahl. In diesem Zusammenhang wird gleich die Produktionsauslastung mit erfragt (siehe Abbildung A-4).

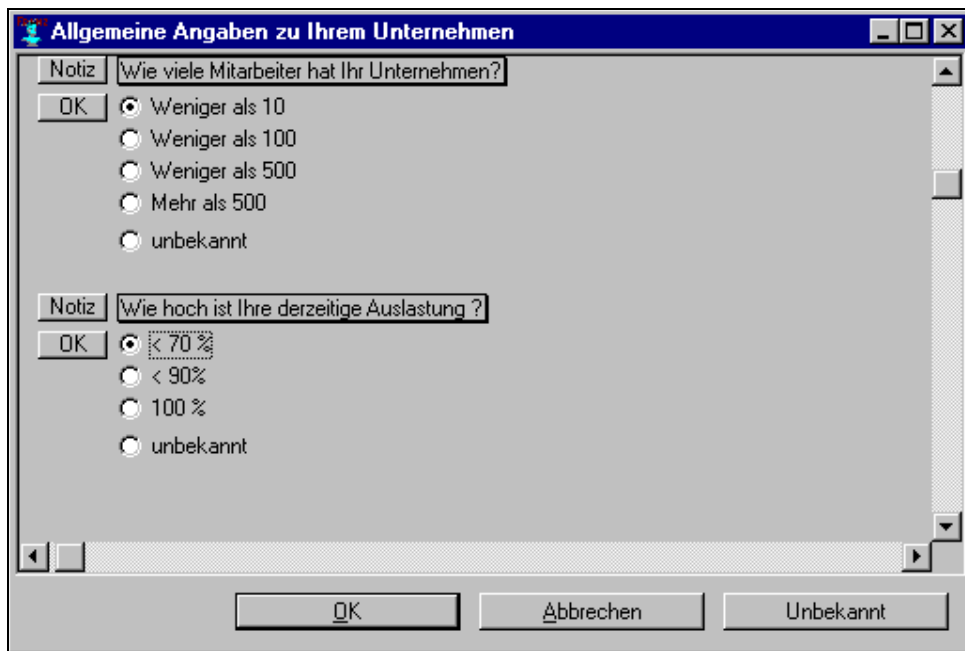


Abbildung A-4: Folgefragen zur Unternehmenseinordnung

Die Angabe der relativ niedrigen Auslastung führt zur Ausführung einer Regel im Expertensystem mit der Empfehlung, elektronische Märkte zur Kundengewinnung einzusetzen. Als Feindiagnosen werden dazu der Aufbau einer mehrsprachigen Anwendung und die Implementierung von Zugangskontrollen durch das einfachere Verfahren der Identifizierung empfohlen (siehe Abbildung A-10, S. 178). Würde ein Handelsunternehmen betrachtet, wäre die Vermeidung negativer Handelsmittlereffekte die erste Strategieempfehlung (siehe Abbildung 7-5, S. 152).

Der Verlauf des Inferenzprozesses kann in D3 über den Begründungsgraph näher veranschaulicht werden (siehe Abbildung A-5, S. 173). Der Begründungsgraph wird vom System automatisch erzeugt und zeigt den Stand der Ableitung aller bestätigten Diagnosen aus den relevanten Antworten zu einem beliebigen Zeitpunkt. Als bestätigt werden Diagnosen mit einer Wahrscheinlichkeit von mehr als 80 Prozent

eingestuft, was einer Wertungsstufe von mindestens P5 im D3-System entspricht [PUPP99, S. 34]. Diese Wertigkeit hat die Empfehlung, die Anwendung mehrsprachig aufzubauen, zu diesem Zeitpunkt der Analyse noch nicht erreicht. Sie hat bis hierher lediglich eine Gewichtung von P1 (vergleiche dazu Abbildung A-8).

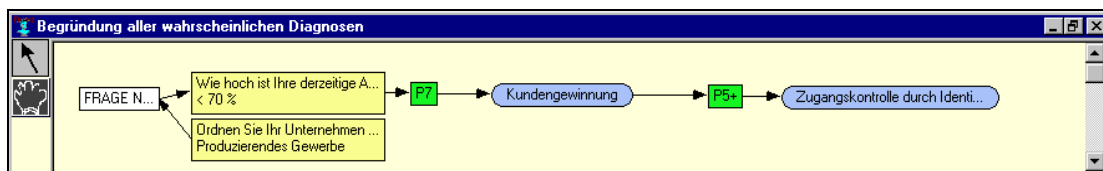


Abbildung A-5: Begründungsgraph nach der Unternehmenseinordnung

Erst im Rahmen der Analyse der Kundenstruktur wird deutlich, dass die geplante Anwendung dringend mehrsprachig aufzubauen ist.

Klassifizierung Ihrer Kunden

Notiz: Welche Ausdehnung hat Ihr Kundenkreis?
 Regional
 National
 International
 unbekannt

Notiz: Wie viele personenbezogene Informationen haben Sie über Ihre Kunden?
 Die meisten sind persönlich bekannt
 Kunden sind vorwiegend anonym
 unbekannt

Notiz: Sind Ihre Kunden Geschäfts- und/oder Privatkunden?
 Geschäftskunden
 Privatkunden
 Geschäfts- und Privatkunden
 unbekannt

OK Abbrechen Unbekannt

Abbildung A-6: Template zur Analyse der Kundenstruktur, Stufe I

Die Empfehlung erhält aus dieser Angabe, dass der Kundenkreis international ist, ein zusätzliches Gewicht von P6 (entspricht 95 Prozent) (siehe Abbildung A-7, S. 174). Als Feindiagnose folgt sofort, dass die Anwendung zur Abbildung der verschiedenen

Sprachen in mehrere Nutzergruppen zu strukturieren ist. Als Konsequenz der Angabe, dass die Kunden dem Anbieter nicht direkt bekannt sind und es im Internet leicht zu Missbrauch kommen kann, wird zudem vorgeschlagen, das Lieferrisiko auf Drittanbieter zu verlagern.

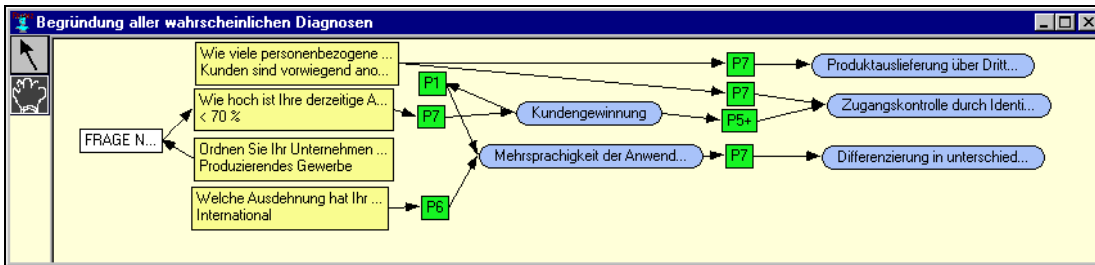


Abbildung A-7: Begründungsgraph nach der Analyse der Kundenstruktur, Stufe I

Die Bearbeitung der anschließenden Frageklassen und die weitere Anwendung der Inferenzstrategien erfolgt analog.

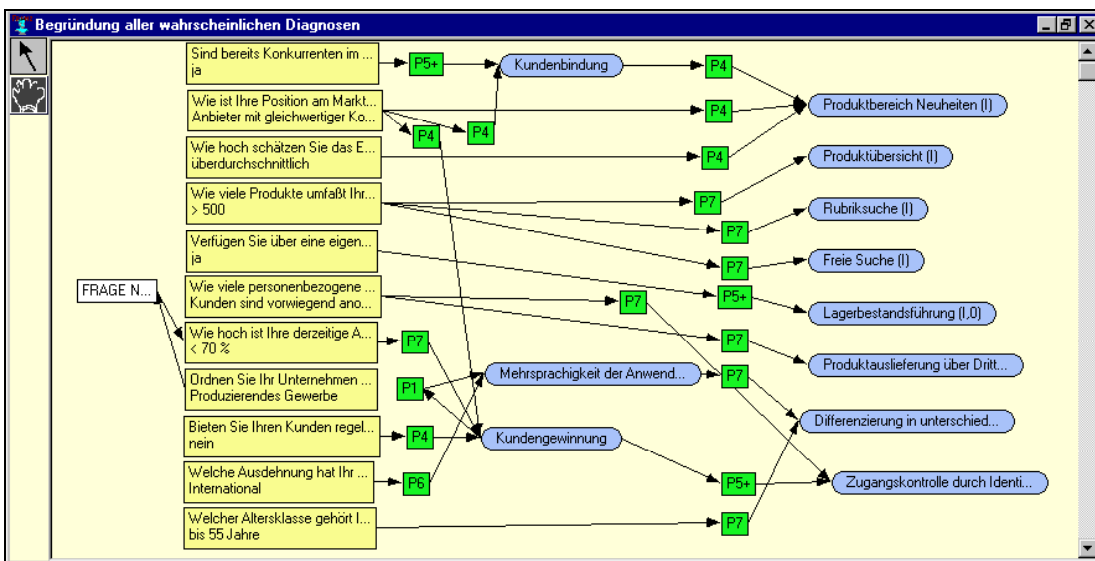


Abbildung A-8: Abschließender Begründungsgraph

Im Beispiel wurden vorrangig Angaben gewählt, die zu einem im gegebenen Rahmen gut darstellbaren Ergebnis führen (siehe Abbildung A-8). So folgt aus der Angabe in der Wettbewerbsanalyse, dass bereits Konkurrenten im Internet zu finden sind, die Empfehlung, durch eine EC-Anwendung die Bindung von Kunden zu forcieren und in diesem Zusammenhang einen speziellen Bereich zur besonders werbewirksamen

Darstellung jeweils aktueller Produktneuheiten einzurichten. Auf diese Weise sollen Kunden zum wiederholten Kauf angeregt werden. Angaben zum Produktspektrum führen zur Auswahl mehrerer Anwendungselemente, die im Zusammenhang mit einem relativ umfangreichen Sortiment unerlässlich sind.

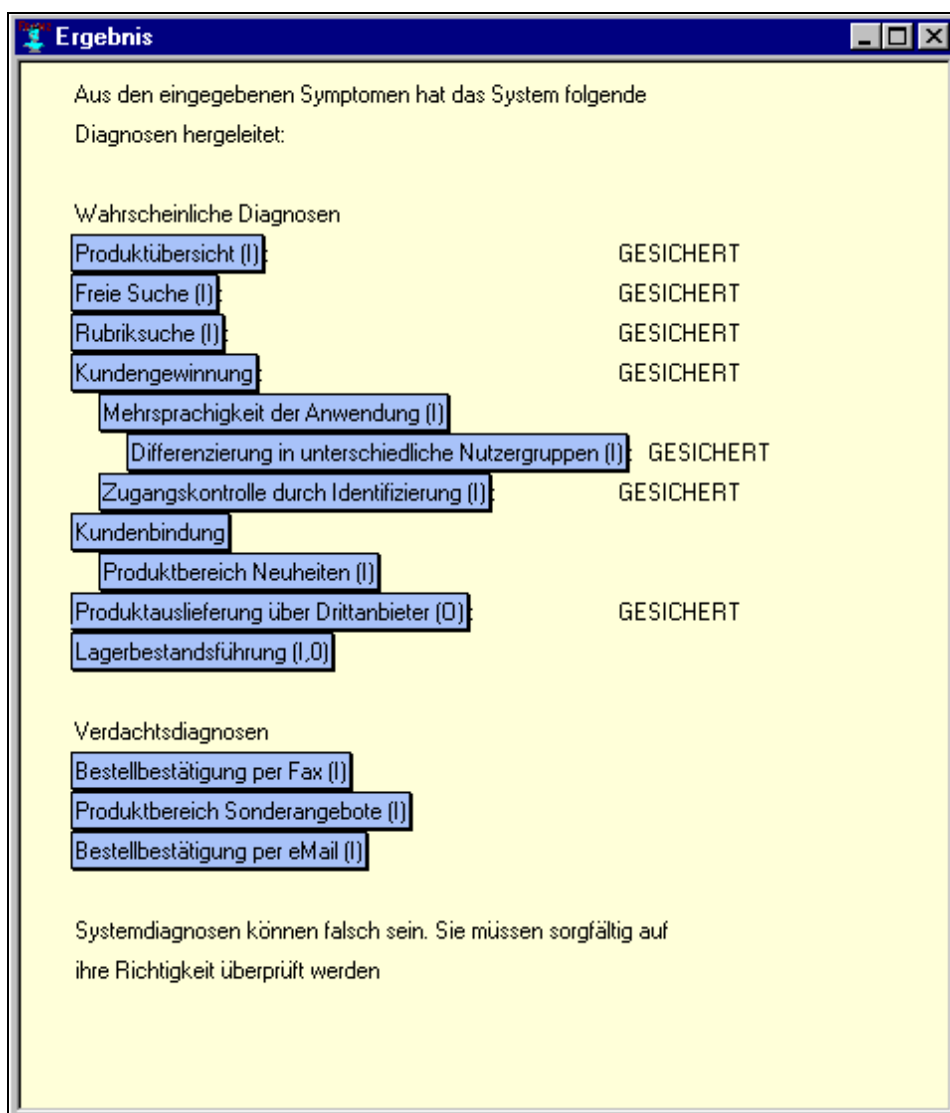


Abbildung A-9: Ergebnis des Inferenzprozesses

Ergebnis des Inferenzprozesses ist die Auswahl gesicherter Diagnosen und sog. Verdachtsdiagnosen (siehe Abbildung A-9).

Gesicherte Diagnosen werden als notwendig zur Umsetzung des Ergebnisses erachtet. Verdachtsdiagnosen entsprechen lediglich Empfehlungen, die abgewählt werden können.

Das Ergebnis wird in Form einer Zeichenkette kodiert an eine URL angehängt, die das Programmmodul zur Erzeugung des Ergebnisberichts aufruft. In diesem Script ist zu jeder möglichen Diagnose ein Textbaustein hinterlegt. Durch Auswertung der URL wird ein Ergebnisbericht in der in Abbildung 7-5, S. 152, dargestellten Form erzeugt. Die Textbausteine können beliebig lang und in HTML formatiert sein, um den Sinn eines Teilvorschlags näher zu begründen. Zusammenhänge zwischen mehreren Ergebnisdiagnosen, wie sie der Begründungsgraph zeigt, sind bisher nicht darstellbar.

Wird der Ergebnisbericht akzeptiert, werden die ausgelesenen Diagnosen an ein weiteres Programmmodul übergeben. Dieser sog. Anwendungsgenerator erzeugt eine neue Anwendungsnummer und fügt entsprechende Einträge in die Tabelle *Anwendung* der Parameterverwaltung, Abbildung 5-5, S. 98, ein. Für alle Komponenten, die ein Element des Anwendungsaufbaus im Ergebnisbericht repräsentieren, wird ein Eintrag in der Tabelle *Benutzte_Komponenten* vorgenommen. Dort wird auch angegeben, ob eine Komponente optional ist oder nicht. Für alle Komponenten, die am Beginn einer Prozesskette stehen können, wird ein Eintrag in der Tabelle *Navigationsleisteneinträge* vorgenommen. Daneben werden zahlreiche weitere Grundeinträge, z. B. für zwei Nutzergruppen mit je einem Bereich oder auch die Basiswerte für die benötigten Komponenten *Suche*, *Rubrikübersicht*, *Produktübersicht* und *Detailansicht*, angelegt. Sie werden durch Eintrag der jeweils nachfolgenden Komponente als Folgefunktion in den Tabellen miteinander verknüpft.

Das Resultat ist eine noch völlig schmucklose, aber bereits voll funktionsfähige Anwendung, die mit dem Prozesseditor und dem Shop-Designer (siehe 7.3.2) interaktiv weiter ausgearbeitet werden kann.

A3 Entscheidungs­bäume zur Ableitung von Diagnosen nach Frageklassen

Entscheidungs­bäume zeigen die Herleitung von Diagnosen in Abhängigkeit von Antwortmöglichkeiten auf die einzelnen Fragen einer Frageklasse. Herleitungen ohne Kennzeichen führen zu einer positiven Gewichtung der Diagnose von 100 Prozent. Das Fragezeichen deutet eine Gewichtung kleiner als 100 Prozent an, das Minus­Zeichen deutet auf negative Gewichtungen hin. In Abhängigkeit von Regeln können einzelne Fragen oder ganze Frageklassen von der Merkmalerhebung ausgenommen werden.

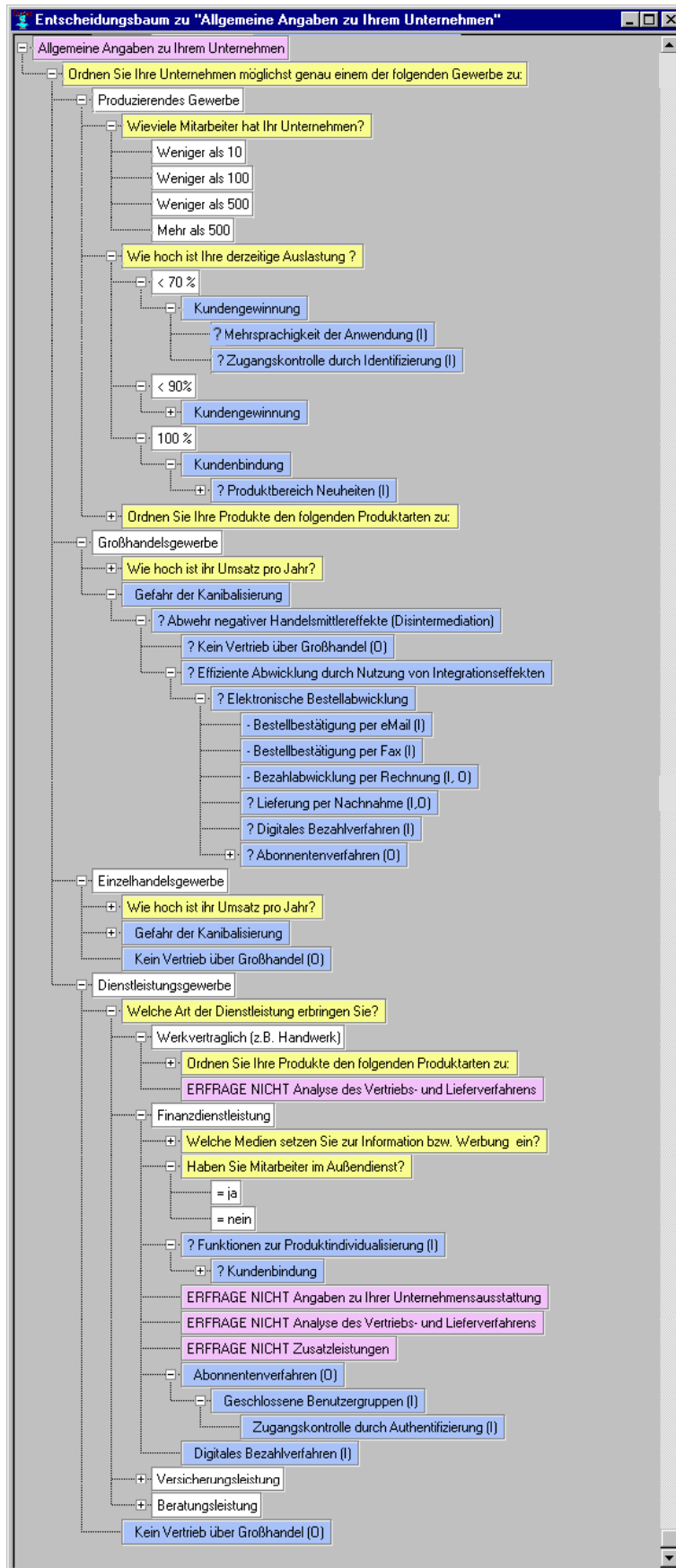


Abbildung A-10: Entscheidungsbaum zur Einordnung des Unternehmens

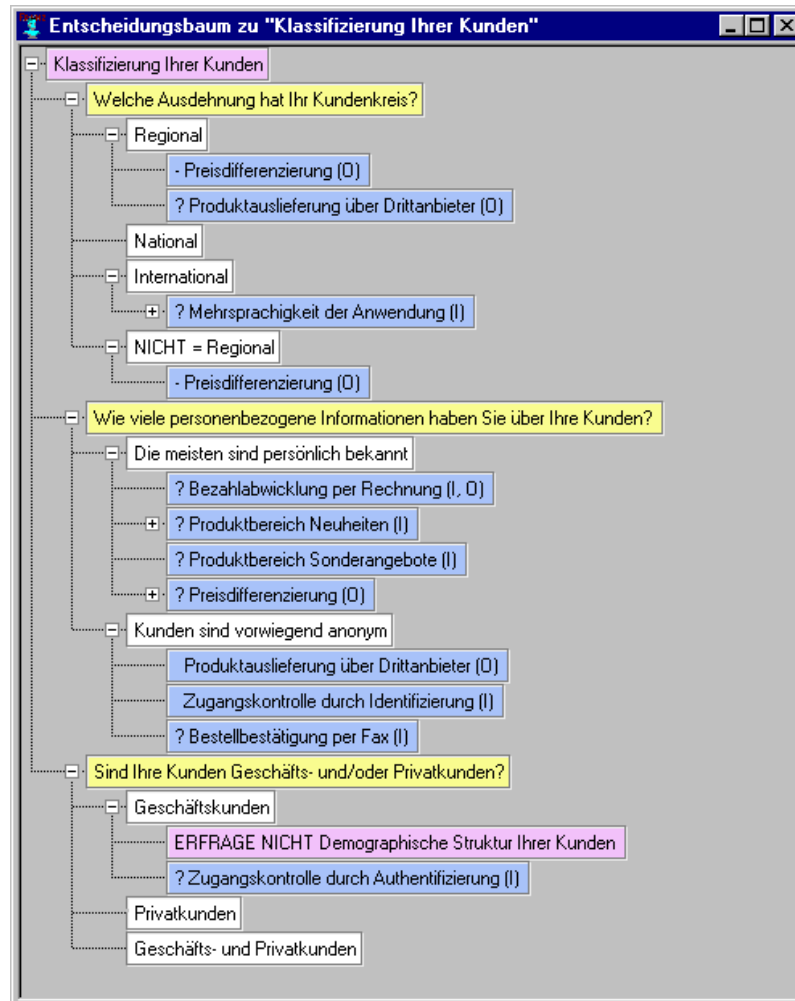


Abbildung A-11: Entscheidungsbaum zur Klassifizierung der Kunden

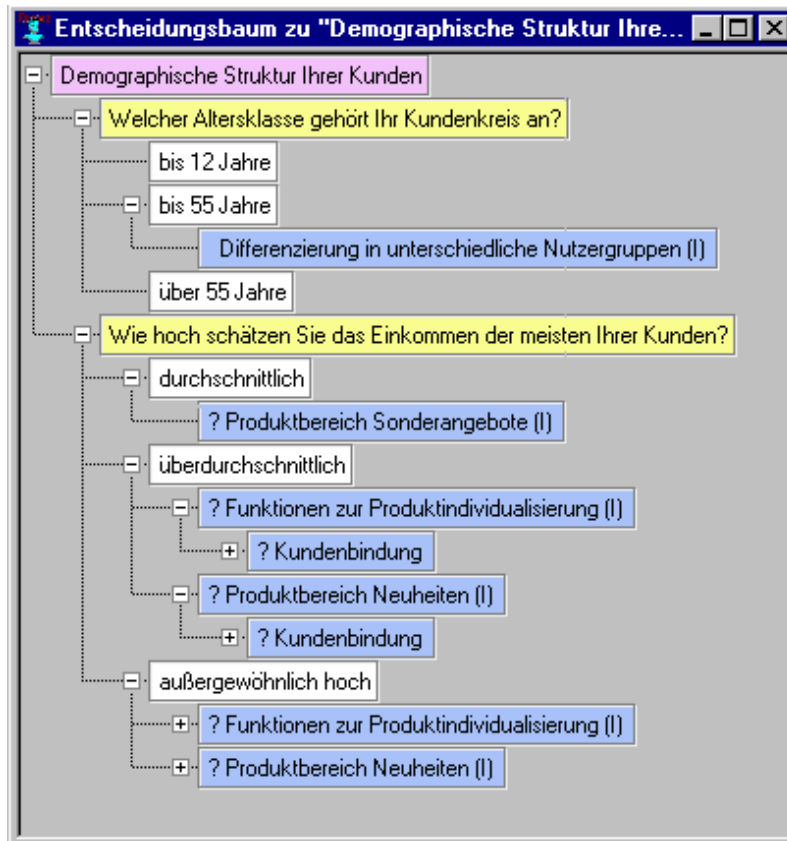


Abbildung A-12: Entscheidungsbaum zur demographischen Analyse

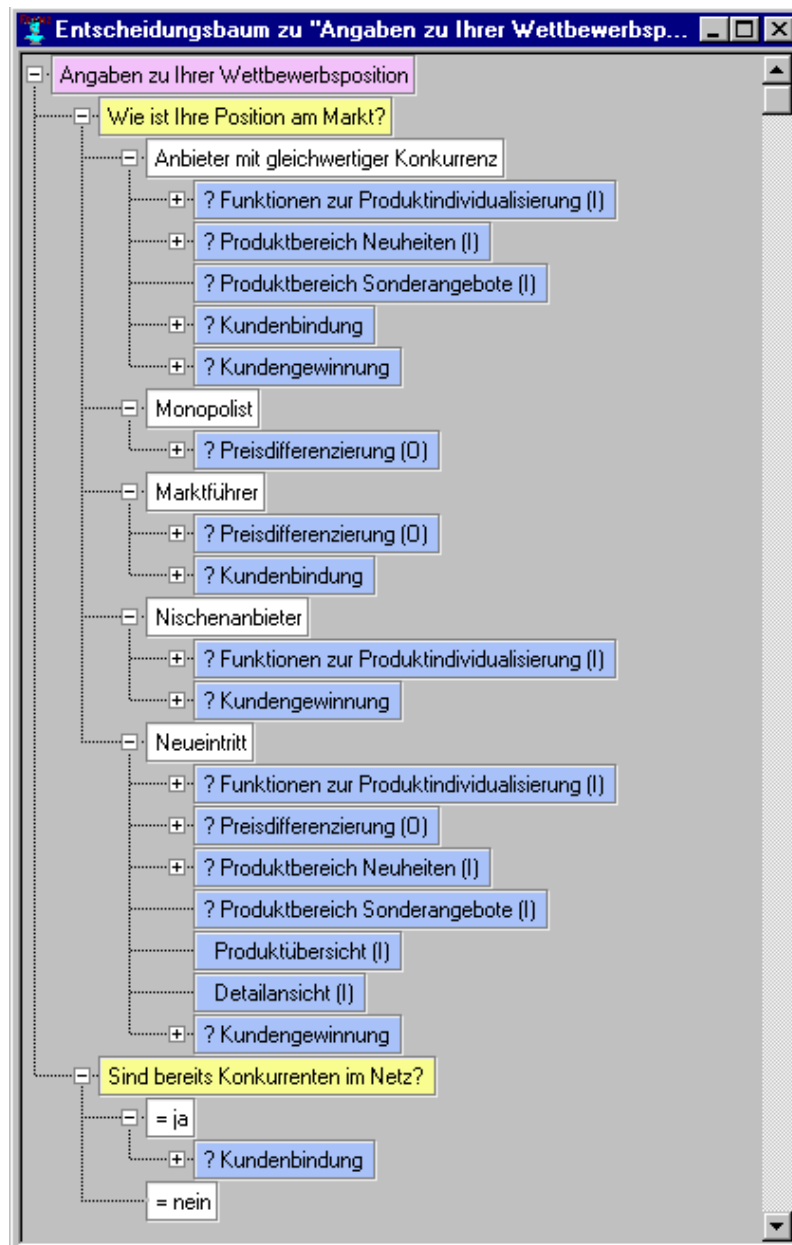


Abbildung A-13: Entscheidungsbaum zur Wettbewerbsanalyse

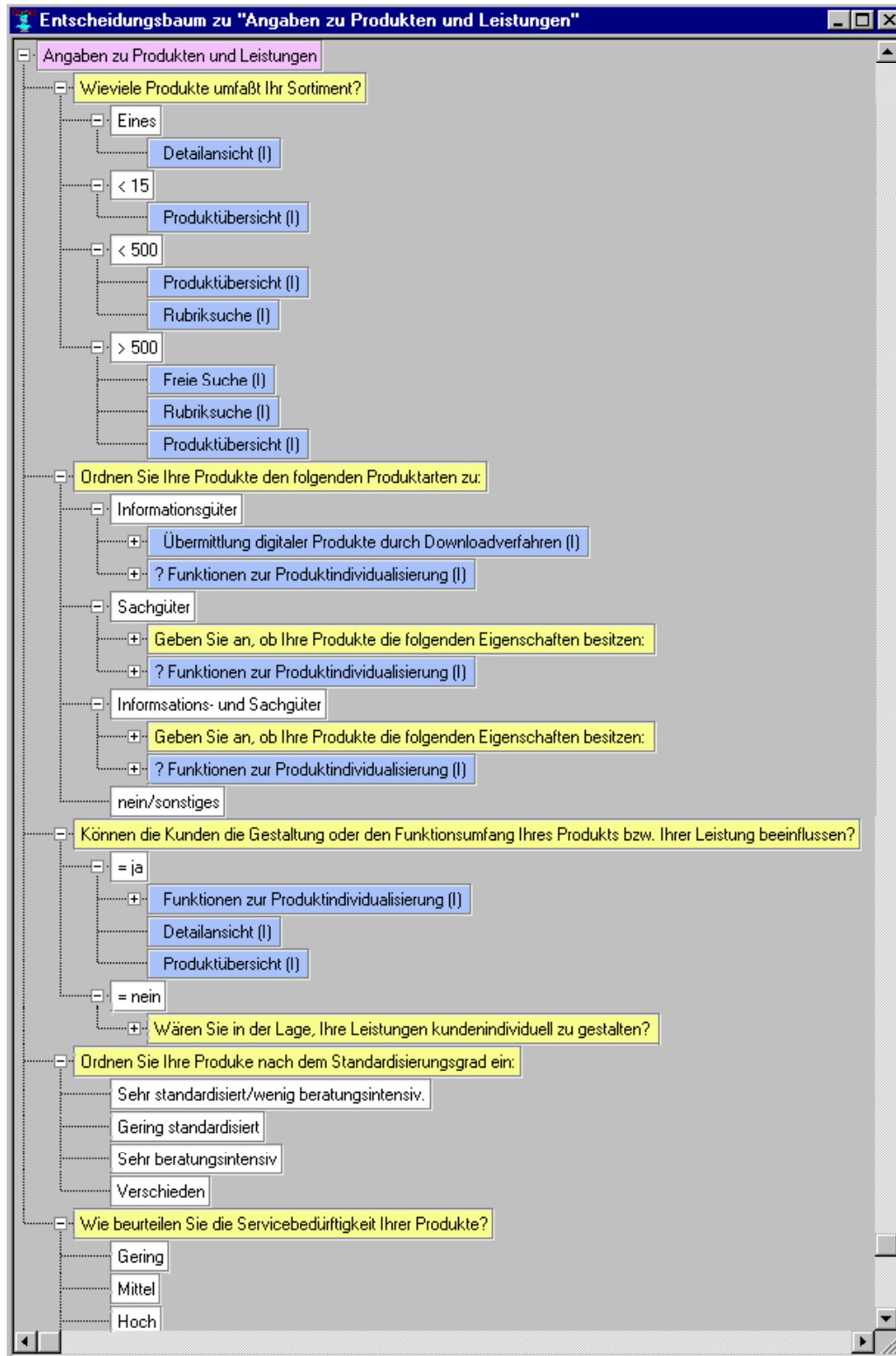


Abbildung A-14: Entscheidungsbaum zur Produktanalyse, Teil I

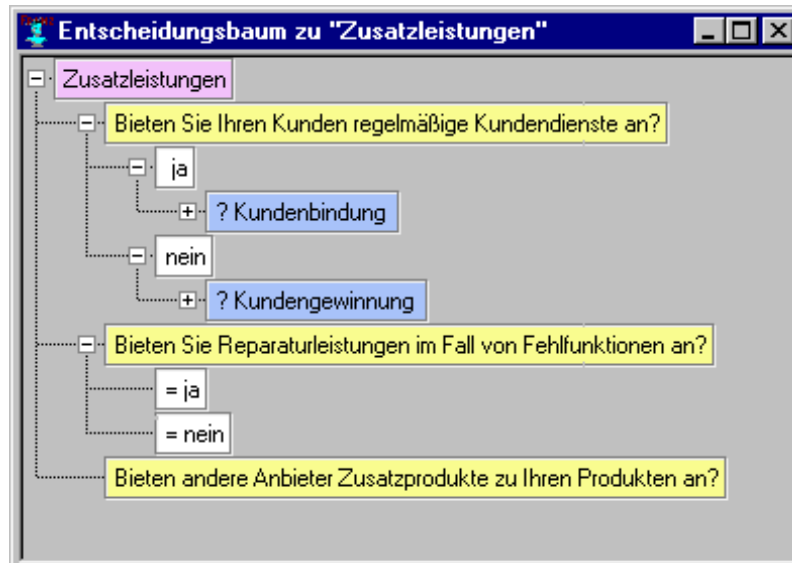


Abbildung A-15: Entscheidungsbaum zur Produktanalyse, Teil II

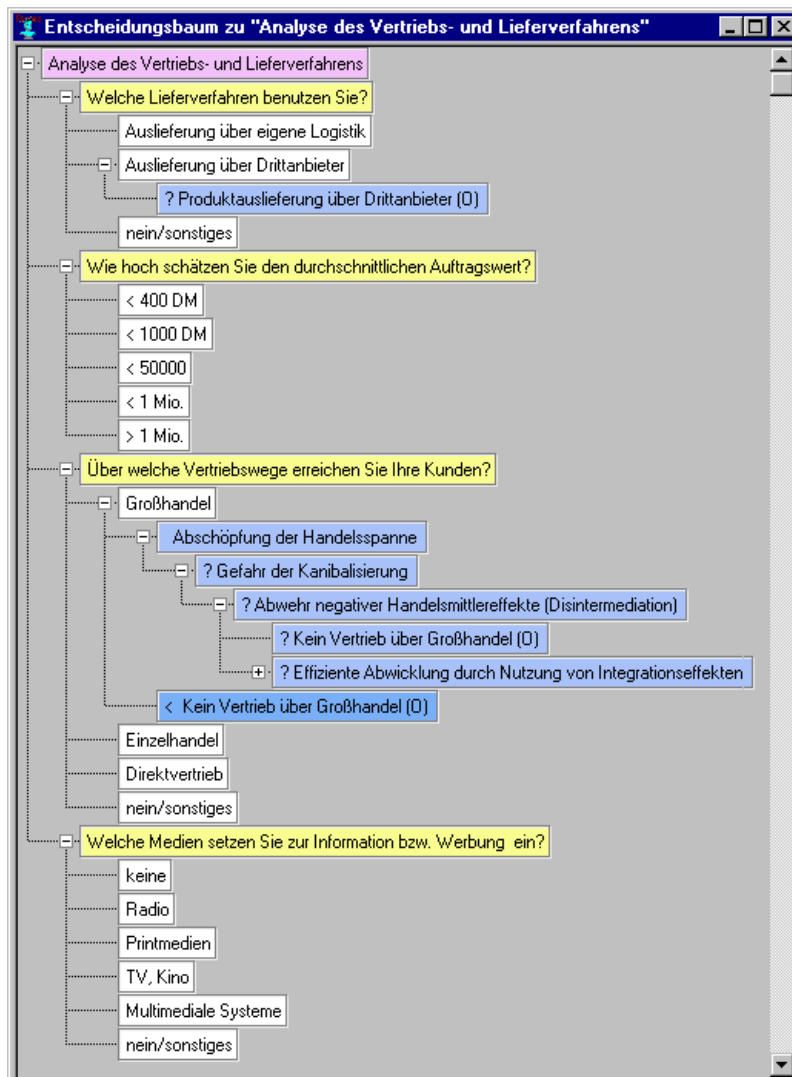


Abbildung A-16: Entscheidungsbaum zur Prozessanalyse

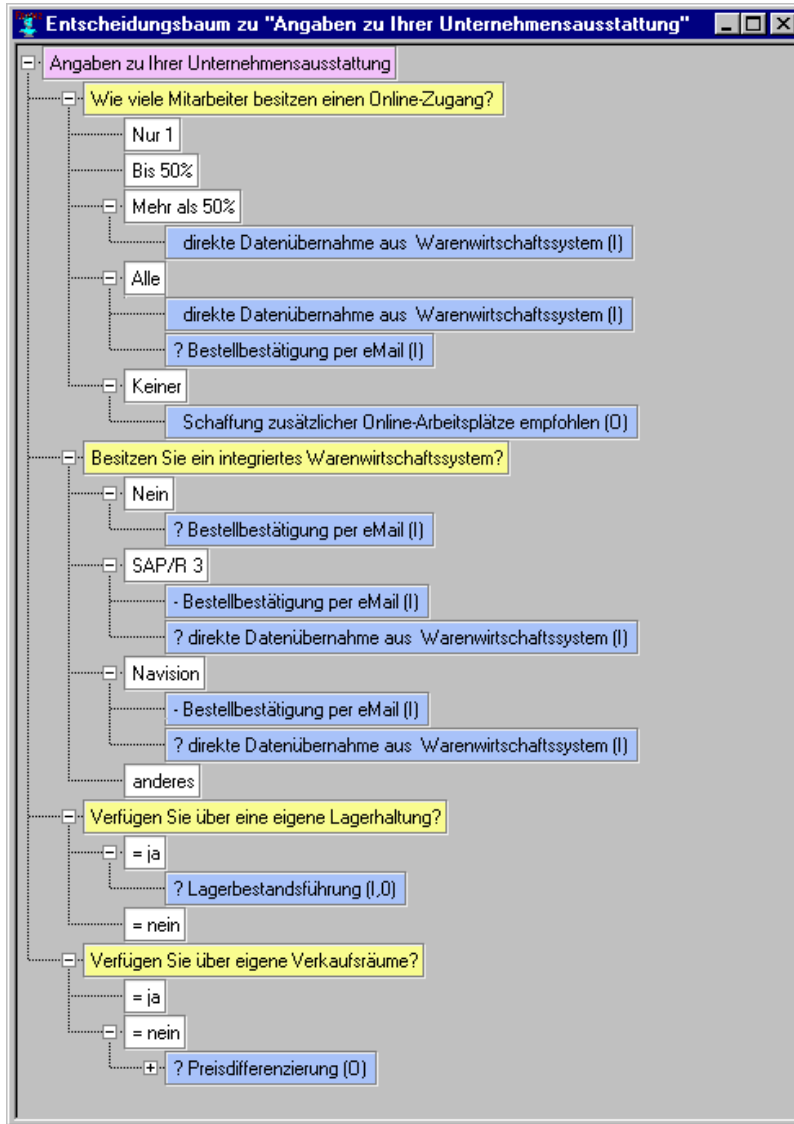


Abbildung A-17: Entscheidungsbaum zur Ressourcenanalyse

Literaturverzeichnis

- [ACKE99] Acker, O.: Evaluierung von Internet-Shop-Lösungen für Kleinunternehmen. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik. Würzburg 1999.
- [APPL01] Ohne Autor: Inside WebObjects; Developing WebObjects Applications with Direct to Web. Apple Computer Inc. 2001.
- [BALZ01] Balzert, Helmut: Lehrbuch der Software-Technik. 2. Auflage. Spektrum Akademischer Verlag. Berlin 2000.
- [BASL89] Basler, H.: Grundbegriffe der Wahrscheinlichkeitsrechnung und Statistischen Methodenlehre. 10. Auflage. Physika Verlag. Heidelberg 1989.
- [BAUM98] Baumgardt, M.: Web Design kreativ! Springer. Berlin 1998.
- [BECK01] Becker, J.: Referenzmodell. In: P. Mertens et al.(Hrsg.): Lexikon der Wirtschaftsinformatik. Springer Verlag. Berlin usw. 2001. S. 399-400.
- [BENJ95] Benjamin, R.; Wigand, R.: Electronic Markets and Virtual Value Chains on the Information Superhighway. In: Sloan Management Review, 1995. S. 62-71.
- [BERG00] Bergner, Klaus et al.: Putting the Parts Together – Concepts, Description Techniques and Development Process for Componentware. Institut für Informatik. Report. Technische Universität München 2000.
- [BOEH88] Boehm, B.: A Spiral Model of Software Development and Enhancement. In: IEEE Computer 5/88. S. 61-67.
- [BOGA95] Bogaschewsky, R.: Vertikale Kooperationen - Erklärungsansätze der Transaktionskostentheorie und des Beziehungsmarketing, in: KAAS, K. P. (Hrsg.): Kontrakte, Geschäftsbeziehungen, Netzwerke - Marketing und Neue Institutionenökonomik, Zeitschrift für betriebswirtschaftliche Forschung, Sonderheft Nr. 35, S. 159-177. Düsseldorf, Frankfurt am Main 1995.
- [BOOC97] Booch, G.: Objektorientierte Analyse und Design: Mit praktischen Anwendungsbeispielen. Addison-Wesley. Unveränderter Nachdruck Addison Wesley. Bonn usw.1997.

- [BORC99] Borchers, D.: Traumgemeinschaften allerorten. In: Computerwoche Spezial 3/1999.
- [BRUH97] Bruhn, M.: Multimedia-Kommunikation; Systematische Planung und Umsetzung eines interaktiven Marketinginstruments. Beck. München 1997.
- [CANT98] Cantor, M.: Object-Oriented Project Management with UML. Wiley Computer Publishing. New York 1998.
- [COAS37] Coase, R.: The Nature of the Firm. In: *Economia* Nr. 4, 1937. S. 386-405.
- [CONA00] Conallen, J.: Building Web Applications with UML. Addison Wesley. Bonn usw. 2000.
- [DEUT98] Deutsch, M.: Electronic Commerce; Zwischenbetriebliche Geschäftsprozesse und neue Marktzugänge realisieren. Vieweg. Wiesbaden 1998.
- [DÖRF99] Dörffeldt, T.: Erfolgreicher PC-Direktvertrieb im Internet. Das Beispiel Dell Computer. In: Hermanns, A.; Sauter, M. (Hrsg.): Management-Handbuch Electronic Commerce. Grundlagen, Strategien, Praxisbeispiele. Vahlen. München 1999. S. 405-409.
- [DT99] Ohne Autor: Deutsche Telekom bietet preiswerten Einstieg in E-Commerce. Pressemitteilung vom 27.08.99.
<http://www.telekom.de/dtag/presse/artikel/0,1018,x309,00.html>.
- [EICK99] Eicker, S.; Nietch, M.: Standards zum objektorientierten Paradigma. In: *Wirtschaftsinformatik* 41 (1999) 4. S. 358-370. Vieweg. Wiesbaden 1999.
- [EU00] Richtlinie des Europäischen Parlamentes und des Rates über bestimmte rechtliche Aspekte des Dienstes der Informationsgesellschaft, insbesondere des elektronischen Geschäftsverkehrs im Binnenmarkt („Richtlinie über den elektronischen Geschäftsverkehr“) vom 8. Juni 2000. Amtsblatt der Europäischen Gemeinschaft 2000.
- [EU00a] Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Koordinierung der Verfahren zur Vergabe öffentlicher Lieferaufträge, Dienstleistungsaufträge und Bauaufträge (Dok. KOM (2000) 275 endg. v. 10. Mai 2000.
- [EYGO01] Consumer Trends in Online Shopping. Global online Retail Report. Ernst & Young 2001.

- [FITT99] Fittkau, S.; Maaß, H.: Electronic Commerce aus der Konsumentenperspektive – Profile und Bedürfnisse der Internet-Nutzer. In: Hermanns, A.; Sauter, M. (Hrsg.): Management-Handbuch für Electronic Commerce. Grundlagen, Strategien, Praxisbeispiele. Vahlen. München 1999. S. 75-83.
- [FLAS02] Ohne Autor: Application Server Comparison Matrix. <http://www.flashline.com/components/appservermatrix.jsp#resources>. Informationsabfrage vom 20.10.2002.
- [FRIS01] Frischmuth, J.; Karrlein, W.: Strategien und Geschäftsmodelle im E-Business. In: Frischmuth, J. et al. (Hrsg.): Strategien und Prozesse für neue Geschäftsmodelle. Praxisleitfaden für E- und Mobile Business. Springer. Berlin usw. 2001. S. 11-37.
- [GEOR01] Georgy, U.: Konkurrenzanalyse mittels kostenloser Wirtschaftsinformationen im Internet. Information, Wissenschaft und Praxis, Nr. 4/2001, S. 197-200.
- [HAGE97] Hagel III, J.; Armstrong, A.: Net Gain – Profit im Netz: Märkte erobern mit virtuellen Communities. Gabler. Wiesbaden 1997.
- [HANS92] Hansen, R.: Wirtschaftsinformatik I. 6. Auflage. Gustav Fischer Verlag. Stuttgart usw. 1992.
- [HANK90] Hanker, J.: Die strategische Bedeutung der Informatik für Organisationen. Industrieökonomische Grundlagen des strategischen Informatikmanagements. Teubner. Stuttgart 1990.
- [HAUN00] Haun, M.: Wissensbasierte Systeme; Eine praxisorientierte Einführung. Expert Verlag. Rennigen-Malmsheim 2000.
- [HERM99] Hermanns, A.; Sauter, M.: Electronic Commerce – Grundlagen, Potentiale, Marktteilnehmer und Transaktionen. In: Hermanns, A.; Sauter, M. (Hrsg.): Management-Handbuch des Electronic Commerce, S.13-29. Vahlen. München 1999.
- [HESS00] Heß, R.: Konzeption und Realisierung der Produktindividualisierung im Online-Shop einer Standardsoftware. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik. Würzburg 2000.
- [HIMB94] Humberger, A.: Der Elektronische Markt als Koordinationssystem. Dissertation, Hochschule St. Gallen 1994.
- [HUFG94] Hufgard, A.: Betriebswirtschaftliche Softwarebibliotheken und Adaption. Vahlen. München 1994.
- [IBM97] VisualAge for Java – Getting Started for OS/2 and for Windows, version 1.0. IBM Corp.1997.

- [INTE00] Ohne Autor: Handbuch Site Administration Intershop 4. Intershop Communications 2000.
- [INTE00a] Ohne Autor: Inside INTERSHOP enfinity. Intershop Communications 2000.
- [INTE00b] Ohne Autor: Benutzerhandbuch Intershop 4. Intershop Communications 2000.
- [INTE01] Developer Guide: Programming with enfinity Cartridge API. Intershop Communications 2001.
- [KATZ85] Katz, M. L.; Shapiro, C: Network Externalities, Competition and Compatibility. In: American Economic Review, 75/1985. S. 424-440.
- [KNÜP00] Knüpfer, W.: Management von EC-Projekten mit UML. In: Thome, R.; Schinzer, H. (Hrsg.): Electronic Commerce. Anwendungsbereiche und Potentiale der digitalen Geschäftsabwicklung. 2. Auflage. Vahlen. München 2000. S. 229-253.
- [KOTL95] Kotler, P.; Bliemel, F.: Marketing Management; Analyse, Planung, Umsetzung und Steuerung. 8. Auflage, Schäffer-Poeschel Verlag. Stuttgart 1995.
- [KRÄH94] Krähmann, N.: Ökonomische Gestaltungsanforderungen für die Entwicklung elektronischer Märkte. Dissertation, Hochschule St. Gallen 1994.
- [KRÄK92] Kräkel, M.: Auktionstheorie und interne Organisation. Nbf, Berlin 1992.
- [KREL61] Krelle, W.: Preistheorie. Mohr. Tübingen 1961.
- [KRUC00] Kruchten, P.: The Rational Unified Process; an Introduction; second edition. Addison Wesley. Bonn usw. 2000.
- [KURB92] Kurbel, K.: Entwicklung und Einsatz von Expertensystemen. Eine anwendungsorientierte Einführung in wissensbasierte Systeme. 2. Auflage. Springer Verlag. Berlin usw. 1992.
- [LANG94] Langenohl, T.: Systemarchitekturen elektronischer Märkte. Hochschule St. Gallen 1994.
- [LANK00] Lankau, R.: Webdesign und -publishing; Projektmanagement für Websites. Carl Hanser Verlag. München 2000.
- [LANK00a] Lankau, R.: Webdesign und -publishing. Grundlagen und Designtechniken. 2. aktualisierte Auflage. 2000.

-
- [LIEB94] Liebowitz, S. J.; Margolis, S.: Network Externality: An uncommon Tragedy. In: Journal of Economic Perspectives. Vol. 8, No. 2, 1994. S. 133-150.
- [LINK99] Linke, D.-M.; Zimmermann, H.-D.: Integrierte Standardanwendungen für Electronic Commerce – Anforderungen und Evaluationskriterien. In: Management-Handbuch Electronic Commerce. Vahlen. München 1999.
- [LOHS02] Lohse C. et al.: Electronic Commerce – Ein Leitfadens für die Anwendung in kleinen und mittleren Unternehmen. Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IOA). Fraunhofer IRB Verlag 2002.
- [MALO87] Malone, T. et al.: Electronic Markets and Electronic Hierarchies. In: Communications of the AMC, 1987 (6). S. 484-497.
- [MALO89] Malone, T.; Yates, J.; Benjamin, R.: The Logic of Electronic Markets. In: Harvard Business Review, 1989 (3). S. 166-172.
- [MART98] Martin, U.: 16. Schritt-für-Schritt-Anleitung zur Entwicklung von Wissensbasen in D3 gemäß dem Formalisierungsmuster „heuristische Entscheidungsbäume“. Dokumentation. Universität Würzburg 1998.
- [MEFF82] Meffert, H.: Marketing – Einführung in die Absatzwirtschaft. 6. Auflage. Wiesbaden 1982.
- [MERT91] Mertens, P.; Wedel, T.; Hartinger, M.: Management by Parameters? In: Zfb 61 (1991) 5/6. S. 569-588.
- [MERT99] Mertens, P.; Bodendorf, F.: Programmierte Einführung in die Betriebswirtschaftslehre. Institutionenlehre. 10. Auflage. Gabler. Wiesbaden 1999.
- [MOUG97] Mougayar, W.: Opening Digital Markets: Battle Plans and Business Strategies for Internet Commerce. New York 1997.
- [MÜLL99] Müller-Ettrich, G.: Objektorientierte Prozessmodelle; UML eingesetzt mit OOIC, V-Modell, Objectory. Addison Wesley. Bonn usw. 1999.
- [MÜLL03] Müller, H.: Gestaltungsanforderungen Internettechnologie-basierter Marktplätze aus Sicht der industriellen Beschaffung; Entwurf eines entscheidungsorientierten objektzentrierten Analyseschemas. Dissertation an der Universität Würzburg. Würzburg 2003.
- [NIER97] Nierstrasz, O.; Lumpe, M.: Komponenten, Komponentenframeworks und Gluing. In HMD 197/1997. S. 8-23.

- [NIST93] National Institute of Standards and Technology (Hrsg.):
Announcing the Standard for Integration Definition for Information
Modelling (IDEF1X). Federal Information Standard Processing
Publication 1993. <http://www.edef.com/Downloads/pdf/Idef1x.pdf>.
- [O.A.97] Ohne Autor: Component Software Glossary. Object Services and
Consulting Inc. 1997.
<http://www.objs.com/survey/ComponentwareGlossary.htm>.
- [O.A.02] Ohne Autor: Konkurrenzanalyse – So funktioniert die
Wettbewerberbeobachtung unter Zuhilfenahme des Internet.
<http://www.onlinemarketer.de/know-how/hintergrund/konkurrenzbeobachtung.htm>. Informationsabfrage
vom 26.10.2002.
- [OMG01] Ohne Autor: Unified Modelling Specification; Version 1.4.
<ftp://ftp.omg.org/pub/docs/formal/01-09-67.pdf>.
Informationsabfrage vom 23.09.2001.
- [PICO98] Picot, A.; Reichwald, R.; Wigand, R.: Die grenzenlose
Unternehmung; Information, Organisation und Management.
3. Auflage. Gabler. Wiesbaden 1998.
- [PICO01] Picot, A.; Reichwald, R.; Wigand, R.: Die grenzenlose
Unternehmung; Information, Organisation und Management.
4. Auflage. Gabler. Wiesbaden 2001.
- [POON99] Poon, S.; Swatman, P.: An explorative study of small business
Internet commerce issues. In: Information & Management 35
(1999) 1. S. 9-18.
- [PUPP90] Puppe, F.: Problemlösungsmethoden in Expertensystemen.
Springer. Berlin usw. 1990.
- [PUPP96] Puppe, F. et al.: Wissensbasierte Diagnose- und
Informationssysteme; mit Anwendungen des Expertensystem-
Shell-Baukastens D3. Springer, Berlin usw. 1996.
- [PUPP99] Puppe, F. et al.: Wissensbasierte Diagnose- und
Informationssysteme mit dem Shell-Baukasten D3. Handbuch.
Würzburg 1999.
- [QUAT00] Quatrani, T.: Visual Modeling with Rational Rose 2000 and UML.
Addison Wesley. Bonn usw. 2000.
- [RATI99] Ohne Autor: Building Web Solutions with the Rational Unified
Process: Unifying the Creative Design Process and the Software
Engineering Process. A Rational Software & Context Integration
white paper. Rational Software Corp. 1999.

- [REYN96] Reynolds, K.: Going, Going, Gone!
<http://www.agorics.com/auctions/auction1.html>. Erstellungsdatum 1996.
- [ROHR97] Rohrbach, P.: Interaktives Teleshopping; Elektronisches Einkaufen auf dem Information Highway. Deutscher Universitätsverlag. Wiesbaden 1997.
- [ROSE01] Rosenblum, D. S.: WREN- An Environment for Component-Based Development. Proceedings of the 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium of the Foundations of Software Engineering FSE-9. Software engineering Notes 26, 5 (2001). S. 207-217.
- [ROYC70] Royce, W.: Managing the Development of Large Software Systems: Concepts and Techniques. Proc. IEEE WESTCON, 1-9. Los Angeles 1970.
- [RUTK97] Rutkowski, Tony: Internet Trends: WWW-Prefixed Hosts. In: www.genmagic.com/Internet/Trends/slide-6.html. Informationsabfrage vom 08.09.1997.
- [SARK95] Sarkar, M. et al.: Intermediaries and Cybermediaries: A Continuing Role for Mediating Players in the Electronic Marketplace. Journal of Computer Mediated Communication, Special Issue on Electronic Commerce, 1 (1995) 3.
- [SCAL01] Scallan, T.: CORBA Primer White Paper.
<http://www.omg.org/news/whitepapers/index.htm#CORBA/IIOP>. Informationsabfrage vom 03.10.01.
- [SCHE90] Scheer, A. W.: EDV-orientierte Betriebswirtschaftslehre. 4. Auflage. Berlin 1990.
- [SCHI98] Schinzer H.;Knüpfner W: Einführungswerkzeuge zur Adaption elektronischer Marktplätze. In: (Wilke, W. et al. (Hrsg.): KI 98 Workshop on Intelligent Systems and Electronic Commerce, S. 19-30. Kaiserslautern 1998.
- [SCHM93] Schmid, B.: Elektronische Märkte. In: Wirtschaftsinformatik 5/93, S. 465-480. Vieweg 1993.
- [SCHM95] Schmid, B. et al.: Electronic Mall: Banking and Shopping in globalen Netzen. Teubner. Stuttgart 1995.
- [SCHM99] Schmid, B.: Elektronische Märkte – Merkmale, Organisation und Potentiale. In: Management-Handbuch Electronic Commerce. Vahlen. München. 1999.
- [SCHM99a] Schmid, B. et al.: Ein Referenzmodell für Gemeinschaften und Medien – Case Study Amazon.com. Net Academy 1999.
<http://www.netacademy.org>.

- [SCHM99b] Schmitt, Eric et al.: Measuring Web Success. The Forester Research 11/99. Forester Research 1999.
- [SCHO98] Schoder, D. et al.: Electronic Commerce Enquête 1997/98. Empirische Untersuchung zum betriebswirtschaftlichen Nutzen von Electronic Commerce für Unternehmen im deutschsprachigen Raum. Executive Research Report. Stuttgart 1998.
- [SCHÖ00] Schön, W.: Interaktion mit einem Expertensystem auf Basis von XML. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik. Würzburg 2000.
- [SCHW97] Schwickert, A.: Web Site Engineering – Modelltheoretische und methodische Erfahrungen aus der Praxis. In: HMD 196/1997. S. 22-35.
- [SIEG97] Siegel, D.: Secrets of Successful Web Sites; Project Management on the World Wide Web. Hayden Books. London 1997.
- [SIEG98] Siegel, D.: Web Site Design: Killer Web Sites der 3. Generation. 2. überarbeitete Auflage. Markt und Technik, Buch und Softwareverlag. Haar bei München 1998.
- [STEF95] Stefik, M.: Introduction to Knowledge Systems. Morgan Kaufman Publishers. San Mateo 1995.
- [STRA02] Strauß, R., Schoder, D.: eReality; Das e-business-Bausteinkonzept. Strategien und Erfolgsfaktoren für das e-business-Management. F.A.Z.-Institut für Management-, Markt- und Medieninformationen. Frankfurt am Main 2002.
- [STRI97] Strinzinger, A.: Komponentenbasierte Softwareentwicklung: Konzepte und Techniken für das Modellieren in Smalltalk. Addison Wesley. Bonn usw. 1997.
- [SZYP99] Szypersky, C.: Component Software: Beyond Object-Oriented Programming. Addison Wesley. Indianapolis 1999.
- [THOM92] Thome, R.: R/3 ist für die Kunden die einzige BWL-Richtlinie. In: Computerwoche 19 (1992) 44. S. 43.
- [THOM96] Thome, R.; Hufgard, A.: Continuous System Engineering; Entdeckung der Standardsoftware als Organisator. Vogel Verlag. Würzburg 1996.
- [THOM96a] Thome, R.: Geschäftsprozessanalyse für Telecommerce. Studie im Auftrag der Deutschen Telekom AG 1996.

- [THOM00] Thome, R.; Schinzer, H. (Hrsg.): Electronic Commerce: Anwendungsbereiche und Potentiale der digitalen Geschäftsabwicklung. 2. überarbeitete Auflage. Vahlen. München. 2000.
- [THOM00a] Thome, R. et al.: Kategorisierung von eC-Geschäftsprozessen zur Identifikation geeigneter eC-Komponenten für die organisierte Integration von Standardanwendungssoftware. FORWIN – Bayerischer Forschungsverbund Wirtschaftsinformatik. Würzburg 2000.
- [TIMM98] Timmers, P.: Business Models for Electronic Markets. In: International Journal of Electronic Markets Vol. 8 No. 2 1998. S. 3-8.
- [VOGE98] Vogelsang, E.: Geschäftsprozessorientierte Adaptionstrategie für betriebswirtschaftliche Softwarebibliotheken; Prozess-Ebenen-Analyse für Ergänzungsentwicklung, Lückenidentifikation und organisatorische Problemlösung. Dissertation an der Universität Würzburg. Würzburg 1998.
- [WARD99] Ward, S.; Kroll, P.: Building Web Solutions with the Rational Unified Process: Unifying the Creative Design Process and the Software Engineering Process. White Paper. Rational Software Cooperation 1999.
- [WERB02] Werbach, K.: Bare Bones Guide to HTML. <http://www.werbach.com/barebones/index.html>. Informationsabfrage vom 20.10.2002.
- [WÖHE93] Wöhe, G.: Einführung in die allgemeine Betriebswirtschaftslehre. 16. Auflage. Vahlen. München 1993.
- [WOLF91] Wolfstetter, E.: Auctions. In: Discussion Papers on Political Economy, Nr. 65 (1991).
- [WOLF96] Wolf, S.; Rebel, T.: Automatisierung von Funktionen und Geschäftsprozessen im Internet – Erfahrungen eines Mittelständlers. In: Wirtschaftsinformatik 38, (1996) 6. S. 579-585.
- [ZBOR96] Zbornik, S.: Elektronische Märkte, elektronische Hierarchien und elektronische Netzwerke; Koordination des wirtschaftlichen Leistungsaustausches durch Mehrwertdienste auf Basis von EDI und offenen Kommunikationssystemen, diskutiert am Beispiel der Elektroindustrie. Universitätsverlag Konstanz. Konstanz 1996.
- [ZERD99] Zerdick, A. et al.: Die Internet-Ökonomie; Strategien für die digitale Wirtschaft. European Communication Council Report. Springer Verlag Berlin. Heidelberg 1999.

- [ZIMM95] Zimmermann, H.-D.; Kuhn, C.: Grundlegende Konzepte einer Electronic Mall. In: SCHM95, S. 33-94.

LEBENS LAUF

ANGABEN ZUR PERSON

Name Knüpffer, Wolf
Geburtsdatum, -ort 21.03.1967, München
Familienstand ledig
Staatsangehörigkeit deutsch

SCHULAU SBILDUNG

09/1973-09/1977 Kerschensteiner-Grundschule, Schweinfurt
09/1977-07/1986 Alexander-v.-Humboldt-Gymnasium, Schweinfurt

HOCHSCHULAU SBILDUNG

08/1986-09/1987 Studium an der Eastern Michigan University, USA;
Hauptfach: Computer Science
11/1987-07/1988 Studium der Informatik an der Universität Würzburg
11/1988-11/1994 Studium der Betriebswirtschaftslehre an der Universität Würzburg,
Abschluss als Diplom-Kaufmann
01/1995-12/2000 Promotionsstudium an der Universität Würzburg

BERUFLICHE TÄTIGKEIT

01/1995-12/2000 Wissenschaftlicher Mitarbeiter am Lehrstuhl für Wirtschaftsinfor-
matik der Universität Würzburg, Prof. Thome
02/2001-12/2002 Schulungsleiter der Administration Intelligence AG, Würzburg
Seit 01/2003 Selbstständiger Systemberater