

Betriebswirtschaftliche Wissenspakete als Informationsträger der Softwareadaption

Themenorientierte Adaptionstrategie entsprechend der
unternehmensspezifischen Softwareumgebung

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
Frank Hennermann
aus Würzburg

Würzburg 2001

Erstgutachter

Professor Dr. Rainer Thome

Inhaltsübersicht

Inhaltsübersicht	1
Inhaltsverzeichnis	3
Abbildungsverzeichnis	9
Tabellenverzeichnis	13
Abkürzungsverzeichnis	15
1 Einordnung, Zielsetzung und Aufbau der Arbeit.....	19
2 Grundlagen und Einflüsse.....	35
3 Ausgewählte Vorgehensmodelle zur Adaption von Softwarebibliotheken	69
4 THESEUS-Methode.....	99
5 THESEUS-Fundamentalmodell	141
6 THESEUS-Umsetzung	167
7 Einsatzszenarien	197
8 Bewertung und Ausblick	221
Anhang A Objekttypen (TAB_OBJ_TYP).....	231
Anhang B Wissenspakete der Versorgungsindustrie	233
Anhang C THESEUS-Navigator (Projekt-Wizard)	237
Anhang D THESEUS-Datenmodell	241
Literaturverzeichnis	243

Inhaltsverzeichnis

Inhaltsübersicht	1
Inhaltsverzeichnis	3
Abbildungsverzeichnis	9
Tabellenverzeichnis	13
Abkürzungsverzeichnis	15
1 Einordnung, Zielsetzung und Aufbau der Arbeit.....	19
1.1 Einordnung.....	19
1.2 Betriebswirtschaftliche Standardsoftware	22
1.2.1 Klassifikationskriterien.....	23
1.2.2 Betriebswirtschaftliche Softwarebibliotheken.....	25
1.3 Situationsanalyse und Thesenbildung	28
1.4 Zielsetzung.....	31
1.5 Aufbau der Arbeit.....	32
2 Grundlagen und Einflüsse.....	35
2.1 Adaption betriebswirtschaftlicher Softwarebibliotheken	36
2.1.1 Adaptionenarten.....	37
2.1.2 Adaptionenwerkzeuge	38
2.1.3 ITHAKA-Konzept.....	38
2.1.3.1 Wissenschaftliche Arbeiten im Adaptionsumfeld.....	39
2.1.3.2 Bestandteile.....	40
2.2 Continuous System Engineering.....	47
2.3 Wissensmanagement	49

2.3.1	Grundzüge des Wissensmanagements	49
2.3.2	Strategien und Werkzeuge	53
2.3.2.1	Wissensbasis.....	53
2.3.2.2	Wissensinfrastruktur	54
2.3.2.3	Hyperknowledge	58
2.3.3	Erfahrungsdatenbanken.....	58
2.4	Mind Mapping mit Knowledge Maps	60
2.4.1	Mind Mapping-Methode.....	60
2.4.2	Knowledge Maps	61
2.5	Erkenntnisfortschritt	65
3	Ausgewählte Vorgehensmodelle zur Adaption von Softwarebibliotheken	69
3.1	Vorgehensmodelle	70
3.1.1	Accelerated SAP (ASAP)	72
3.1.1.1	Konzeptionelle Grundlagen von ASAP	74
3.1.1.2	Phasen von ASAP	75
3.1.1.3	Werkzeugverwendung in ASAP.....	79
3.1.1.4	Erweiterung des Grundmodells.....	83
3.1.2	LIVE METHOD/Chestra	84
3.1.2.1	Konzeptionelle Grundlagen von Chestra.....	85
3.1.2.2	Phasen von Chestra	87
3.1.2.3	Werkzeugverwendung in Chestra.....	88
3.1.3	Sonstige Vorgehensmodelle	90
3.2	Beurteilung der Vorgehensmodelle	92
3.3	Prinzipien für eine Adaptions-Workbench	95
4	THESEUS-Methode	99
4.1	Präzisierung der Aufgabenstellung	100

4.1.1	Umsetzungsstrategien	103
4.1.2	Stufenmodell	107
4.1.3	Architektur.....	110
4.2	Modellgestaltung.....	113
4.2.1	Modellintention und Modellierungsansatz.....	113
4.2.2	Softwarebezug.....	115
4.2.3	Darstellung.....	115
4.2.4	Werkzeugunterstützung	116
4.3	Wissenspakete	117
4.3.1	Typisierung	118
4.3.2	Charakterisierung.....	119
4.3.3	Allgemeine Eigenschaften	123
4.3.4	Attribute eines Wissenspakets.....	125
4.3.5	Zuordnungen.....	125
4.3.6	Prüffälle.....	129
4.3.7	Lebenszyklus	131
4.3.8	Informationsquellen	134
5	THESEUS-Fundamentalmodell	141
5.1	Aufbau der Monitorbeschreibung.....	141
5.1.1	Fragestellung.....	141
5.1.2	Gestaltung und Aussage	142
5.1.3	Modellelemente.....	142
5.1.4	Beispiel	143
5.1.5	Technische Umsetzung.....	144
5.2	Allgemeine Gestaltungsprinzipien.....	145
5.3	Analyse-Monitor	148
5.3.1	Gestaltung des Analyse-Monitors	149
5.3.2	Beispiel eines Analyse-Monitors.....	150
5.4	Customizing-Monitor	153
5.4.1	Gestaltung des Customizing-Monitors.....	155
5.4.2	Beispiel eines Customizing-Monitors	158

5.5	Kalkulations-Monitor	160
5.5.1	Gestaltung des Kalkulations-Monitors	163
5.5.2	Beispiel eines Kalkulations-Monitors.....	164
6	THESEUS-Umsetzung	167
6.1	Die Adaptions-Workbench als Component Ware	167
6.2	Komponenten der Workbench.....	171
6.2.1.1	Monitor	171
6.2.1.2	Navigator	172
6.2.1.3	Engineer	173
6.3	Methodische Umsetzung	173
6.3.1	Fundamentalmodell.....	174
6.3.2	Entwicklung des Individualmodells	174
6.3.3	Anpassende Modellierung	175
6.3.4	Aufwandsermittlung.....	176
6.3.5	Baselining	179
6.3.6	Continuous System Engineering.....	185
6.4	Technische Umsetzung.....	187
6.4.1	Präsentationswerkzeug.....	187
6.4.2	Datenmodell.....	189
7	Einsatzszenarien.....	197
7.1	Nutzung im Projektverlauf	197
7.2	Einstiegsszenarien.....	203
7.2.1	THESEUS auf Basis der Rasterhypothese.....	204
7.2.2	THESEUS auf Basis des LIVE KIT Structure.....	206
7.2.3	THESEUS auf Basis des LIVE KIT Power	208
7.2.4	THESEUS auf Basis einer RBE-Analyse	211
7.3	Konzerneinsatz	215
7.3.1	Anforderungen im Konzernumfeld	215
7.3.2	OKEANOS-Methode.....	216

7.3.3	Die Symbiose zwischen THESEUS und OKEANOS.....	218
8	Bewertung und Ausblick	221
8.1	Kontinuierliche betriebswirtschaftliche Entwicklung.....	221
8.2	Internet based Consulting.....	224
8.3	Gesamtbewertung.....	228
Anhang A	Objekttypen (TAB_OBJ_TYP).....	231
Anhang B	Wissenspakete der Versorgungsindustrie	233
Anhang C	THESEUS-Navigator (Projekt-Wizard)	237
Anhang D	THESEUS-Datenmodell	241
	Literaturverzeichnis	243

Abbildungsverzeichnis

Abbildung 1-1:	Aufbau der Arbeit.....	32
Abbildung 2-1:	Bestandteile von ITHAKA [in Anlehnung an VOG97, S. 224]	39
Abbildung 2-2:	Arbeiten zur Adaption betriebswirtschaftlicher Softwarebibliotheken [in Anlehnung an VOG97, S. 50].....	40
Abbildung 2-3:	Anforderungsabgleich mit dem betriebswirtschaftlichen Anforderungsnavigator ODYSSEUS [in Anlehnung an THOM96, S. 100].....	42
Abbildung 2-4:	Zusammenhang von LIVE KIT Structure, LIVE KIT Power und LIVE KIT Control [VOGE97, S. 189].....	44
Abbildung 2-5:	Möglichkeiten zur Reduktion des Fundamentalmodells am Beispiel von LIVE KIT Power	45
Abbildung 2-6:	Komponenten des Wissensmanagements.....	51
Abbildung 2-7:	Wissensinfrastruktur [GENT99, S. 15].....	55
Abbildung 2-8:	Knowledge Map für den Unternehmensbereich Produktion.....	62
Abbildung 2-9:	Wissensbasierte Adaption.....	67
Abbildung 3-1:	Implementation Assistant von ASAP [SAP00d]	73
Abbildung 3-2:	Question and Answer Database (Ausschnitt) [SAP00d]	77
Abbildung 3-3:	Concept Check Tool (Ausschnitt) [SAP00d].....	83
Abbildung 3-4:	Aktivitäten im Implementierungsprozess von LIVE METHOD/Chestra.....	88
Abbildung 3-5:	Einsatzzeitpunkt der integrierten Werkzeuge in Chestra.....	90
Abbildung 4-1:	Stufenkonzept THESEUS.....	110
Abbildung 4-2:	THESEUS-Architektur	111
Abbildung 4-3:	Widmung der Wissenspakete durch MEDEA	119
Abbildung 4-4:	Charakterisierung der Wissenspakete.....	120
Abbildung 4-5:	Ablauf der Relevanzentscheidung bei Zugriff auf eine Checkliste.....	131

Abbildung 4-6:	Lebenszyklus eines Wissenspakets.....	134
Abbildung 4-7:	Online-Dokumentation (Ausschnitt) [SAP00a, PM-WOC-MO/CS-SE].....	136
Abbildung 4-8:	Einführungsleitfaden des SAP-Systems (Ausschnitt).....	137
Abbildung 4-9:	Ereignis-Prozess-Kette (Ausschnitt).....	138
Abbildung 4-10:	Solution Map Utilities (Ausschnitt).....	139
Abbildung 5-1:	Darstellungselemente des Monitor-Datenmodells	144
Abbildung 5-2:	Schematischer Grundaufbau der Adaptions-Workbench	146
Abbildung 5-3:	Detailinformationen zu einem Wissenspaket.....	150
Abbildung 5-4:	Beispiel eines Analysemonitors	151
Abbildung 5-5:	Beispiel eines Customizing-Monitors	158
Abbildung 5-6:	Legende des Customizing-Monitors.....	159
Abbildung 5-7:	Aufwandsermittlung des LIVE KIT Structure	162
Abbildung 5-8:	Beispiel eines Kalkulations-Monitors	165
Abbildung 6-1:	Projektstruktur in der Adaptions-Workbench (Beispiel).....	176
Abbildung 6-2:	Aufwandsberechnung THESEUS	178
Abbildung 6-3:	BC-Set (einfach) [SAP00, BC-CUS].....	182
Abbildung 6-4:	BC-Set (hierarchisch) [SAP00, BC-CUS].....	182
Abbildung 6-5:	Automatische Konfiguration mit THESEUS	183
Abbildung 6-6:	Versionierung der Wissenspakete	186
Abbildung 6-7:	Tabelle zur Steuerung der sprachabhängigen Textvergabe.....	190
Abbildung 6-8:	Tabellenstruktur zur Ablage der Wissenspakete.....	191
Abbildung 6-9:	Tabellenstruktur für die Projekt-, Benutzer- und Berechtigungsverwaltung.....	194
Abbildung 7-1:	Projektphasen.....	198
Abbildung 7-2:	Rasterhypothese als Informationsquelle für THESEUS	205
Abbildung 7-3:	Ergebnistabelle zur Speicherung der Rasterhypothese	206

Abbildung 7-4:	LIVE KIT Structure als Informationsquelle für THESEUS	207
Abbildung 7-5:	Evaluationsdatei des LIVE KIT Structure.....	208
Abbildung 7-6:	LIVE KIT Power als Informationsquelle für THESEUS	210
Abbildung 7-7:	Projektdatenbank des LIVE KIT Power.....	211
Abbildung 7-8:	Reverse Business Engineering als Informationsquelle für THESEUS.....	213
Abbildung 8-1:	Aufbau eines Change Szenarios (in Anlehnung an [BÄTZ99a])	223
Abbildung 8-2:	Nutzung der Wissenspakete bei der Definition eines Change Szenarios.....	224
Abbildung 8-3:	Integrationsmöglichkeit von THESEUS und IANUS.....	228

Tabellenverzeichnis

Tabelle 1-1:	Standardtypen betriebswirtschaftlicher Lösungen [vgl. HUFG94, S. 71].....	25
Tabelle 2-1:	Aufbau des Unternehmensbereichs Produktion.....	61
Tabelle 3-1:	Kriterien für die Beurteilung verschiedener Vorgehensweisen zur Adaption von Softwarebibliotheken.....	70
Tabelle 3-2:	Übersicht der gewidmeten Werkzeuge [in Anlehnung an STRE99, S. 76-77].....	89
Tabelle 3-3:	Beurteilung der Vorgehensmodelle.....	92
Tabelle 4-1:	Attribute eines Wissenspakets (Kopfdaten).....	125
Tabelle 4-2:	Leistungsmerkmale ausgewählter Adaptionswerkzeuge.....	126
Tabelle 4-3:	Zuordnungen eines Wissenspakets.....	128
Tabelle 4-4:	Status der Wissenspakete nach dem Einlesen der Informationsquellen.....	129
Tabelle 5-1:	Gestaltungselemente im Analyse-Monitor.....	152
Tabelle 5-2:	Status der Customizingaktivitäten.....	157
Tabelle 5-3:	Gestaltungselemente im Customizing-Monitor.....	159
Tabelle 5-4:	Gestaltungselemente im Kalkulations-Monitor.....	166
Tabelle 6-1:	Tabellen zur Definition und Ablage der Wissenspakete.....	192
Tabelle 6-2:	Tabellen für die Projektverwaltung.....	195
Tabelle 6-3:	Tabellen für die Benutzer- und Berechtigungsverwaltung.....	195
Tabelle 7-1:	Anforderungen im Konzernumfeld [THOM96, S. 141-142].....	216

Abkürzungsverzeichnis

ABAP/4	Advanced Business Application Programming/4
ALE	Application Linking Enabling
API	Application Programming Interface
ARIS	Architektur integrierter Informationssysteme
ASAP	Accelerated SAP
B2B	Business to Business
B2C	Business to Customer
BAPI	Business Application Programming Interface
BE	Business Engineer
BPR	Business Process Reengineering
CA	Cross Application
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CATT	Computer Aided Testing Tool
CD-ROM	Compact Disc-Read Only Memory
CIM	Computer Integrated Manufacturing
CO	Controlling
CORBA	Common Object Request Broker Architecture
CRM	Customer Relationship Management
CSE	Continuous System Engineering
DBMS	Datenbankmanagementsystem
DDD	Development Data Dictionary
DDE	Dynamic Data Exchange
EDI	Electronic Data Interchange
EPK	Ereignisgesteuerte Prozesskette
ERP	Enterprise Resource Planning
FI	Financials

IANUS	Internetbasierte Abwicklung von Consultingprojekten und -analysen im Umfeld betriebswirtschaftlicher Softwarebibliotheken
IDEAL	Induktiv deduktiver Ansatz zur logischen Betrachtung betrieblicher Informationsflüsse
IM	Investment Management
IMG	Implementation Guide
ITHAKA	Prozess- und strukturintegrierende, toolgestützte, heuristische Architektur der kundenspezifischen Adaption von betriebswirtschaftlichen Softwarebibliotheken
IS-U	Industrial Solution-Utilities
KIRKE	Kontinuierliche Istanalyse produktiver ökonomischer Systeme
KRONOS	Kontinuierliche R/3-basierte betriebswirtschaftliche Evolution, Organisationsentwicklung und Einbindung Neuer Releases in Operierende R/3-Systeme
LKS	LIVE KIT Structure
LKP	LIVE KIT Power
MAPI	Messaging Application Programming Interface
MEDEA	Merkmalsorientierte, dynamische Ermittlung von Anforderungen an Softwarebibliotheken
MENTOR	Management Entscheidungsnavigator zur transparenten, objektorientierten Reorganisation
MM	Material Management
MS	Microsoft
ODBC	Open Database Cconnectivity
ODD	Online Data Dictionary
ODYSSEUS	Organisatorisch-dynamische Spezifikation von Softwarebibliotheken entsprechend der Unternehmensstruktur
OKEANOS	Organisation von Konzernführungen durch Adaption von Standardsoftwarebibliotheken
OLE	Object Linking and Embedding
OLYMP	Organisationsgestaltung und dynamische Adaption
OSS	Online Service System
PANDORA	Projektabwicklung und Dynamische Organisation der R/3-Adaption

PENELOPE	Prozessebenenanalyse für Ergänzungsentwicklungen, Lückenidentifikation und organisatorische Problemlösung
PP	Production Planning
Q&Adb	Question and Answer Database
RDBMS	Relationales Datenbankmanagementsystem
RFC	Remote Function Call
RUP	Rational Unified Process
SAP	SAP AG, Walldorf
SASW	Standardsoftware
SBS	Siemens Business Services
SCM	Supply Chain Management
SD	Sales and Distribution
SMTP	Simple Mail Transfer Protocol
SNI	Siemens Nixdorf Informationssysteme
SPARTA	Spezifische Ableitung von Referenzsystemen und Templates für Anwendersegmente
SQL	Structured Query Language
THESEUS	Themenorientierte Adaptionstrategie entsprechend der unternehmensspezifischen Softwareumgebung
WiPak	Wissenspaket

1 Einordnung, Zielsetzung und Aufbau der Arbeit

Moderne betriebswirtschaftliche Softwarelösungen speichern in den Systemeinstellungen und Tabelleneinträgen eine sehr große Menge des jeweiligen unternehmensspezifischen Wissens über Geschäftsabläufe und Organisationsstrukturen. Der vermehrte Einsatz unterschiedlicher Lösungen zur Abbildung der internen Prozesse auf der einen und zur Öffnung der Geschäftstätigkeiten im Bereich des Electronic Business (e-Business) auf der anderen Seite stellt neue Herausforderungen an die Methoden und Konzepte zur Einführung bzw. Betreuung der eingesetzten Systeme.

Im Rahmen dieser Arbeit wird untersucht, welche Schritte notwendig sind, um eine transparente Speicherung und Nutzung des vorhandenen Wissens sowie der gesammelten Informationen system- bzw. methodenübergreifend sicherzustellen und wie die bestehenden Ansätze aus dem Bereich der Softwareadaption in den Gesamtkontext integriert werden können. Kernpunkte sind die methodische Unterstützung einer zielgerichteten Navigation durch die Informationsflut und die Entwicklung eines geeigneten Instrumentariums, das in der Lage ist, heterogene Softwarelandschaften abzubilden und als Dokumentationsbasis für zukünftige Systemmodifikationen zu dienen.

1.1 Einordnung

Diese Arbeit ist in eine Reihe von Beiträgen einzuordnen, die ihren Anfang in der DORIN-Studie von THOME et al. [THOM79] im Jahre 1978 findet, in der das IDEAL-Modell entwickelt wurde. In dieser Studie wurden Handlungsanweisungen sowohl für den Aufbau von Standardanwendungssoftware als auch für deren Adaption beschrieben. WAGNER [WAGN83] und MÜBIG [MÜSS91] griffen die Ergebnisse dieser Studie auf und verfeinerten sie.

HUFGARD schaffte mit dem ODYSSEUS-Konzept (Organisatorisch-Dynamische Spezifikation von Systemmodulen entsprechend der Unternehmensstruktur) die Grundlage dafür, den Adaptionprozess und die Komplexität von Softwarebibliotheken (am Beispiel des Softwareproduktes R/3 der SAP AG) wesentlich zu reduzieren [HUFG94]. In Kooperation mit der Siemens Business Services GmbH &

Co. OHG wurde dieses Konzept mit dem Anforderungsnavigator LIVE KIT Structure verwirklicht. Mit Hilfe des Anforderungsnavigators können die Kundenanforderungen an das Softwareprodukt werkzeugunterstützt ermittelt werden. Somit kann durch regelbasiertes Auswählen, Anpassen und Ergänzen eine effektive und kostengünstige Implementierung der Standardanwendungssoftware R/3 erreicht werden.

Durch die Steigerung der Leistungsfähigkeit betrieblicher Standardsoftware verlagerte sich in den Unternehmen der Schwerpunkt von der Entwicklung individueller Softwarelösungen hin zur Adaption komplexer Standardanwendungssoftware [KIRC96, S. 1], was zu einer steigenden Bedeutung des ODYSSEUS-Konzepts führte. Die mit dem immensen Funktionsumfang einhergehende Komplexität und Intransparenz der Standardsoftwarepakete erzeugte allerdings auf Anwenderseite eine gewisse Abwehrhaltung [KLUK97, S. 48]. Die Hersteller reagieren darauf mit Methoden und Werkzeugen zur Unterstützung der Softwareeinführung in Form von Vorgehensmodellen (siehe Kapitel 3.1).

VOGELSANG ergänzte ODYSSEUS durch das PENELOPE-Konzept (Prozeß-Ebenen-Analyse für Ergänzungsentwicklung, Lückenidentifikation und Organisatorische Problemlösungen), das eine problemorientierte Darstellung von Prozessinformationen ermöglicht, die nach eindeutig definierten Fragestellungen in Ebenen strukturiert ist [VOGE97].

Der von WEDLICH entwickelte MENTOR-Ansatz (Management-Entscheidungs-Navigator zur transparenten objektorientierten Reorganisation) ist die konsequente Fortsetzung des PENELOPE-Konzeptes. Dieser Ansatz beschäftigt sich mit einer entscheidungsorientierten Adaptionsstrategie für Berichte auf Basis der Prozessmodellierung [WEDL97].

BÄTZ griff die Erkenntnisse dieser Arbeiten auf und übertrug sie auf den Bereich der betrieblichen Organisation. Im Rahmen des OLYMP-Konzepts (Organisationsgestaltung und dynamische Adaption) wurde eine Vorgehensweise entwickelt, die eine systematische Gestaltung und kontinuierliche Anpassung von Organisationsstrukturen bei der Anwendung von betriebswirtschaftlichen Softwarebibliotheken unterstützt [BÄTZ99].

In der Arbeit von MEHLICH wurde das MEDEA-Instrumentarium (Merkmalsorientierte, dynamische Ermittlung von Anforderungen an Softwarebibliotheken)

entwickelt, das auf den Überlegungen des ODYSSEUS-Konzepts basiert. Dabei kommen zusätzliche methodische Ansätze, wie z. B. die Betriebstypologie, zum Einsatz, so dass die Anforderungsnavigation gemäß dem ODYSSEUS-Konzept erweitert und feiner differenziert werden kann [MEHL98].

SCHIPP geht mit seinem SPARTA-Konzept (Spezifische Ableitung von Referenzsystemen und Templates für Anwendersegmente) so weit, dass aufgrund der Definition von bestimmten Anwendersegmenten vorkonfigurierte Systeme erstellt werden können [SCHI99]. Diese Systeme enthalten eine typische Konfiguration für dieses Anwendersegment.

Die beschriebenen Konzepte beruhen alle auf der von HUFgard zugrunde gelegten Annahme der Einführung von Standardanwendungssoftware in mittelständischen Unternehmen [HUF94, S. 19]. WALZ hat mit der OKEANOS-Methode (Organisation von Konzerneinführungen durch Adaption von Standardsoftwarebibliotheken) ein Vorgehensmodell entwickelt, das die toolunterstützte Implementierung einer Standardanwendungssoftware in Konzernunternehmen auf Basis des ODYSSEUS-Konzepts erlaubt [WALZ00].

Die im Rahmen dieser Arbeit entwickelte THESEUS-Methode (Themenorientierte Adaptionstrategie entsprechend der unternehmensspezifischen Softwareumgebung) stellt eine Erweiterung der oben genannten Konzepte dar, die der zunehmenden Bedeutung von ergänzenden Softwarelösungen und Vorgehensmodellen Rechnung trägt. Diese werden über eine neutrale Navigations- und Arbeitsumgebung in das bestehende Instrumentarium integriert.

Neben dem ODYSSEUS-Konzept und seinen Erweiterungen gibt es mehrere andere Ansätze zur Unterstützung des Adaptionprozesses, die sich zum Teil ebenfalls mit der Prozessmodellierung (z. B. [SCHE97]), der Organisationsgestaltung (z. B. [ESSW92]) oder der Branchenausrichtung einzelner Softwaresysteme (z. B. [JDED98]) befassen. So bietet J. D. Edwards beispielsweise branchenorientierte Referenzmodelle für die Konfiguration von unternehmensspezifischen Geschäftsprozessen. In diesem Ansatz werden durch Fragen und Antworten Systemkonfigurationen abgeleitet, die den Anforderungen des jeweiligen Unternehmens entsprechen [JDED98, o. S.]. Die Werkzeuge dieses Ansatzes können, analog zur Integration der ODYSSEUS-Komponenten, über Schnittstellen mit THESEUS verbunden und gemäß der hier entwickelten Methode genutzt werden. SCHEER hat das

Konzept der Architektur integrierter Informationssysteme entwickelt, welches Geschäftsprozesse auf Basis von Referenzmodellen beschreibt und durch Zerlegung der Prozesse in Sichten eine Reduktion der Komplexität gewährleistet [SCHE97, o. S.]. Zentrale Elemente in diesem Ansatz sind die ereignisgesteuerten Prozessketten (EPK), die detailliert den inhaltlichen und zeitlichen Ablauf von Ereignissen und Funktionen beschreiben. Die EPKs können ähnlich der Integration zwischen THESEUS und PENELOPE in die im Rahmen dieser Arbeit entwickelte Navigations- und Arbeitsumgebung eingebunden werden.

Für die weiteren Betrachtungen sind diese Ansätze jedoch nicht von zentraler Bedeutung, da es sich bei der Integration primär um Schnittstellendefinitionen und Zuordnungen handelt, die keinen Einfluss auf die Ausformung von THESEUS haben. Vielmehr zeigt die Möglichkeit der Einbindung anderer Vorgehensweisen und Werkzeuge die Offenheit und Flexibilität der hier entwickelten Methode. Für nähere Informationen zu den oben beschriebenen Ansätzen wird auf die angegebene Literatur verwiesen.

1.2 Betriebswirtschaftliche Standardsoftware

Bereits im Jahre 1994 hat HUGFARD ein Klassifikationssystem für betriebliche Standardanwendungssoftware aufgestellt [HUGF94, S. 69-78]. Diese Klassifikation, die auch 1997 von VOGELANG wieder aufgegriffen wurde [VOGE97, S. 10-19], soll an dieser Stelle übernommen werden und als Basis für weitere Überlegungen dienen.

HUGFARD schlägt, aufbauend auf der Klassifikation von Anwendungssoftware von THOME [THOM90, S. 4], folgende anwendungsorientierte Differenzierung vor [HUGF94, S. 68]:

- Standardbürowerkzeuge,
- technische Lösungen und
- betriebswirtschaftliche Lösungen.

In der Gruppe Standardbürowerkzeuge werden alle endbenutzerorientierten Werkzeuge von Electronic Mail bis zur Tabellenkalkulation zusammengefasst. Das Einsatzspektrum dieser Werkzeuge ist sehr heterogen und jeweils relativ eng begrenzt.

Jedoch zeichnen sie sich durch eine sehr hohe Kombinations- bzw. Integrationsfähigkeit aus. Über standardisierte Schnittstellen, wie DDE, OLE oder CORBA, lassen sie sich sehr effizient miteinander verknüpfen und erreichen damit ein enormes Lösungspotenzial. Die Schnittstellen werden auch vielfältig von den betriebswirtschaftlichen Paketen benutzt. Durch die Verfügbarkeit von Makrosprachen können sie individuell programmiert werden. Zusätzlich ersetzen sie zunehmend für bestimmte Anwendungsbereiche die klassischen Programmiersprachen.

Technische Lösungen, orientiert an Entwicklern und Technikern, stehen i. d. R. der Leistungserstellung des Unternehmens näher. Hierbei handelt es sich um Applikationen, die im Rahmen des Computer Integrated Manufacturing (CIM) die Produktentwicklung und Herstellung unterstützen (CAD, CAE, CAM etc.). Die Integration mit den betriebswirtschaftlichen Softwarepaketen wird nicht zuletzt durch die ISO 9000 Norm vorangetrieben, z. B. in Form einer Dokumentenverwaltungs- und -archivierungspflicht.

Betriebswirtschaftliche Lösungen umfassen diejenigen Funktionen und Aufgaben von Unternehmen, die immer gleich oder ähnlich vorhanden sind. Die klassischen "horizontalen" Anwendungskomponenten, wie Personalabrechnung und Finanzbuchhaltung, werden erweitert um Anwendungen aus den Bereichen Anlagenwirtschaft, Controlling, Einkauf, Verkauf, Marketing, Materialwirtschaft, Produktionsplanung und -steuerung und Lagerwirtschaft. Grundvoraussetzung ist hierbei, dass jede dieser Anwendungen an die konkreten ablauf- und aufbauorganisatorischen Eigenheiten unterschiedlicher Unternehmen anpassbar sein muss.

Diese allgemeine Definition von betriebswirtschaftlichen Lösungen soll auch für die nachfolgenden Betrachtungen gelten. Aufgrund der unterschiedlichen Erscheinungsformen von betriebswirtschaftlichen Lösungen, z. B. von kleinen PC-basierten Lösungen wie KHK auf der einen und SAP R/3 auf der anderen Seite, hat HUFGARD ein umfassendes Klassifikationsschema für die betriebswirtschaftliche Standardsoftware entwickelt.

1.2.1 Klassifikationskriterien

Folgende fünf Kriterien ermöglichen eine qualitative Unterscheidung der angebotenen Lösungen. Sie sind für den Anwender maßgebend in bezug auf Einstufung

und Auswahl und berücksichtigen die für die Adaption entscheidenden Aspekte von Standardlösungen [VOGE97, S. 13 und HUEG94, S. 69-79].

1. Der betriebswirtschaftliche Abdeckungsgrad einer Software kann an der Funktionsbreite, der Quantität der Fachbereiche und Funktionen, ermittelt werden. Die Funktionsbreite kann von gering (hier sind nur wenige Funktionen in einigen Funktionsbereichen abgedeckt) bis hin zu sehr groß (hier werden auch Randgebiete mit umfangreichen Funktionen abgedeckt) bewertet werden.
2. Das betriebswirtschaftliche Profil lässt auf die Funktionstiefe, also die Detailliertheit der Funktionen, schließen. Zusätzlich wird analysiert, ob unterschiedliche Alternativen für Funktionen vorhanden sind. Die Ausprägungen reichen von einem nur punktuell detaillierten Funktionsprofil bis hin zu einer gleichmäßig hohen Funktionstiefe mit alternativen Funktionen.
3. Die Systematik der Entwicklung beschreibt, ob bei der Softwareentwicklung bereits auf eine hohe Verwendungsstandardisierung hingearbeitet wurde. Die konzeptionelle und technische Softwarequalität spielt auch eine wichtige Rolle. Die Entwicklung kann von unsystematisch bis hin zu betriebstyp- und branchenorientiert erfolgt sein.
4. Die Möglichkeit der Anpassung an unternehmensindividuelle Vorstellungen erfordert spezielle Adaptionswerkzeuge, die ein schnelles und effizientes Vorgehen ermöglichen. Wenn keine Adaptionswerkzeuge vorhanden sind, müssen Änderungen im Sourcecode vorgenommen werden. Im besten Fall sind mächtige Adaptionswerkzeuge vorhanden, die methodisch unterstützt werden.
5. Unter Flexibilität wird die Möglichkeit der Anpassung an die Unternehmensorganisation verstanden. Je weniger Restriktionen vorhanden und organisatorische Abläufe zwingend vorgegeben sind, desto höher ist die Flexibilität. Die Flexibilität wird zusätzlich erhöht, wenn Anpassungen revidierbar und releasewechselneutral sind.

Tabelle 1-1: Standardtypen betriebswirtschaftlicher Lösungen [vgl. HUG94, S. 71]

Standardtypen Merkmale	I Individualentwicklungen für mehrere Unternehmen	II Branchenlösungen	III Adaptierbare Standardsoftware	IV Betriebswirtschaftliche Softwarebibliotheken
1. Funktionsbreite	gering	mittel	groß	sehr groß
2. Betriebswirtschaftliches Profil	punktuell sehr speziell	teilweise sehr speziell	generell und speziell ausprägbar	generell, speziell und alternativ ausprägbar
3. Systematik der Entwicklung	unsystematisch	branchenorientiert	betriebstyporientiert	betriebstyp- und branchenorientiert
4. Mächtigkeit der Adaptionswerkzeuge	keine	rudimentär	technisch mächtig	technisch und methodisch mächtig
5. Flexibilität	starre Lösung	begrenzt flexibel	sehr flexibel	sehr flexibel und revidierbar

Jedes dieser Merkmale erlaubt dabei vier Ausprägungen, die für die Bewertung eingesetzt werden können. Daraus lassen sich vier Standardtypen ableiten, die sich aus der Kombination der Merkmalsausprägungen jeweils einer Stufe ergeben (vgl. Tabelle 1-1). Die Standardtypen I bis III sind für das zu entwickelnde Instrumentarium nicht weiter von Bedeutung, da sie nicht adaptierbar sind. Als Standardtyp IV ergibt sich die betriebswirtschaftliche Softwarebibliothek, auf die im folgenden näher eingegangen wird.

1.2.2 Betriebswirtschaftliche Softwarebibliotheken

Eine betriebswirtschaftliche Softwarebibliothek besitzt eine große Funktionsbreite mit vielen alternativen, sich ergänzenden Programmfunktionen, aus denen die benötigten Elemente selektiert werden. Die Möglichkeit einer individuellen Kombinations-, Erweiterungs- und Anpassungsfähigkeit mittels Adaption ist der Hauptvorteil gegenüber den anderen Kategorien betriebswirtschaftlicher Standardsoftware. Die einzelnen Kategorien werden bei HUGARD ausführlich erläutert [HUG94, S. 67-114].

Folgende Eigenschaften zeichnen eine betriebswirtschaftliche Softwarebibliothek aus [vgl. HUF94, S. 77-79]:

1. Ein sehr hoher Funktionsumfang sorgt für eine breite Abdeckung der relevanten Bereiche eines Unternehmens, wie Vertrieb, Rechnungswesen und Produktion. Dadurch können komplexe und bereichsübergreifende Abläufe unterstützt werden. Zahlreiche Schnittstellen ermöglichen zudem die Anbindung spezieller Ergänzungsprogramme, wie Bürosoftware, Kommunikationsprogramme, technische Lösungen oder Fremdsysteme.
2. Die Verbesserung der Verwendungsstandardisierung ermöglicht ein durchgängiges und tief ausgeprägtes Anforderungsprofil, indem für betriebswirtschaftliche Funktionen alternative Ausprägungsformen zur Verfügung stehen.
3. Mit Hilfe von branchen- und betriebstypenorientierten Kriterien kann eine breite und detaillierte Entwicklung erreicht werden. Weiterhin wird durch Funktions-, Daten- und Prozessmodelle eine Homogenität des Systems gesichert, die bei einer gewachsenen Standardanwendungssoftware (SASW) nicht möglich ist.
4. Durch die große Funktionsbreite, das ausgeprägte Anforderungsprofil und die integrierte Sichtweise der Zusammenhänge werden methodisch und technisch leistungsfähige Adaptionswerkzeuge benötigt. Diese müssen die Komplexität der Einführung von der Anforderungsanalyse bis zur Re-Adaption verringern.
5. Durch eine Trennung von Anwendungen, Darstellung, Schnittstellen und Datenbank verfügen Softwarebibliotheken über eine große Flexibilität bezüglich der Topologie und der Skalierung eines Systems. Diese flexible Architektur bildet zudem die Voraussetzung, um Ergänzungen im laufenden Betrieb vornehmen und periodische Wechsel von Versionen (Releasewechsel) durchführen zu können.

Die Sammlung betriebswirtschaftlicher Programme wird nicht allein durch eine große funktionale Mächtigkeit zu einer Softwarebibliothek. Erst wenn eine systematische Ordnung und entsprechende Mechanismen zum Auffinden und Aufbewahren der Funktionen vorhanden sind, kann von einer Bibliothek gesprochen

werden. THOME hat sieben Bedingungen formuliert, die erfüllt werden müssen, um die Voraussetzungen für die in Kapitel 2.1 beschriebene Adaption einer betriebswirtschaftlichen Softwarebibliothek zu schaffen [THOM98, S. 49f. und THHU96, S. 44f.]:

1. „Abdeckung von 80% der potentiell benötigten betriebswirtschaftlichen Funktionen,
2. betriebswirtschaftliche Adaptionfähigkeit der Module und Funktionen durch Parametrisierung oder andere Adaptionsverfahren,
3. konsistente Verarbeitung und Verwaltung der Daten bei asynchroner Anpassung oder Weiterentwicklung,
4. dynamische Adaptionfähigkeit unter Beibehaltung der Konsistenz des produktiven Systems,
5. dynamische Modulaustauschfähigkeit durch festgelegte Schnittstellen,
6. gemeinsamer Zugriff aller Module auf eine laufend aktualisierte Datenbank sowie
7. Unabhängigkeit von der Hardwareplattform.“

VOGELANG hat anhand dieser Kriterien Aussagen für die Anpassung betriebswirtschaftlicher Softwarebibliotheken abgeleitet [vgl. VOG97, S. 16-18]:

- Sowohl das Anwenderunternehmen als auch der fachliche Experte sehen sich durch das starke Wachstum der Softwarebibliotheken mit einem zunehmend unübersichtlichen Funktionalitätsangebot konfrontiert. Aus diesem Grund werden neben einem Verzeichnis des Inhalts auch Hinweise und Unterstützung bei der Auswahl benötigt. Dabei müssen Funktionen aber als Teil eines integrierten Systems betrachtet und dargestellt werden.
- Aufgrund der Universalität und Variantenvielfalt kann eine Vielzahl von Kombinationsmöglichkeiten entstehen. Dies führt zu einer großen Zahl möglicher Strukturen und Abläufe. Zur Beurteilung und Bewertung individueller Ausprägungen werden entsprechende Werkzeuge benötigt, um Restriktionen und Lücken aufzudecken.

- Mit zunehmendem Umfang des betriebswirtschaftlichen Inhalts bei gleichzeitiger Modularität nimmt die Zahl der Schnittstellen zwangsläufig zu. Zusammen mit den vorhandenen Verbindungspunkten zur Außenwelt durch die Anbindung technischer Subsysteme (BDE-, CAD-Systeme etc.), durch generell nutzbare standardisierte Schnittstellen (DDE, OLE, CORBA etc.), über das Internet durchführbare Zugriffe (BAPI und Internetszenarien) oder auch spezielle Verteilungsmodelle (ALE) kann die Anpassbarkeit und Leistungsfähigkeit einer Softwarebibliothek gesteigert werden. Dies setzt aber Transparenz bezüglich der vorhandenen Optionen voraus. Sonst können die technisch existierenden Möglichkeiten nicht entsprechend umgesetzt und genutzt werden.
- Durch die umfangreiche Funktionalität bei gleichzeitiger Variantenvielfalt und ausgeprägter Modularität entsteht eine Komplexität, die eine Adaptionnotwendigkeit einer betriebswirtschaftlichen Softwarebibliothek zwangsläufig erzeugt. Zur Reduzierung der Komplexität ist es nicht ausreichend, wenn durch Methoden und Werkzeuge lediglich das Einstellen der Funktionen oder das Setzen von Parametereinstellungen unterstützt werden. Benötigt werden vielmehr Adaptionswerkzeuge, die bei der Entwicklung übergreifender Lösungen helfen und durch die Freiheitsgrade offengelegt werden.
- Selbst wenn nur die Einführung einer SASW betrachtet wird, werden sich Änderungen während der Adaption nicht vermeiden lassen. Spätestens im produktiven Einsatz müssen durch die Möglichkeit einer dynamischen Konfigurierbarkeit Veränderungen im laufenden Betrieb vorgenommen werden können. Dies setzt eine entsprechende Unterstützung der Re-Adaption durch die Werkzeuge voraus.

Die Auflistung zeigt, dass hohe Anforderungen an Werkzeuge gestellt werden, die geeignet sind, die Einführung und den produktiven Einsatz eines derartigen Systems zu unterstützen.

1.3 Situationsanalyse und Thesenbildung

Die weite Verbreitung des Internet ermöglicht es den Unternehmen, neue Geschäftsfelder und Potenziale zu nutzen. Die Wettbewerbsfähigkeit wird zuneh-

mend an der Ausrichtung im Bereich des Electronic Business gemessen. Die Tendenzen zu vernetzten und unternehmensübergreifenden Geschäftsabwicklungen lassen sich unter Schlagworten, wie CRM (Customer Relationship Management), SCM (Supply Chain Management), B2B (Business to Business) oder B2C (Business to Customer), zusammenfassen. Klassische ERP-Systeme, wie beispielsweise die SAP-Software R/3, bilden den Kern für die internen Abläufe eines Unternehmens und werden zunehmend durch Softwarelösungen aus dem Bereich des Electronic Business ergänzt, z. B. Twister der Brokat AG [BROK00] oder Marketing Manager der Update.com Software AG [UPDA00]. Wichtig ist in diesem Zusammenhang, dass die Unternehmen eine Gesamtstrategie im Sinne des „Best-of-Breed“-Ansatzes verfolgen und das Software-Portfolio für die internen und externen Geschäftsprozesse sorgfältig auf die Anforderungen abstimmen. Moderne Austauschformate (z. B. XML) und weitreichende Kooperationen der Softwarehersteller, beispielsweise der SAP AG mit dem CRM-Spezialist Clarify [PLAT00 und CLAR00], ermöglichen es den Unternehmen, die spezifischen Vorteile unterschiedlicher Lösungen zu nutzen. Große Softwarehersteller bieten darüber hinaus Lösungsangebote, die sowohl die klassische ERP-Welt als auch die moderne „e-Business-Welt“ umfassen.

Diese Entwicklungen induzieren eine zunehmende Heterogenität in der Systemlandschaft, wobei die Lösungen untereinander integriert sind. Für die Einführung und Betreuung dieses heterogenen Lösungs-Portfolios bedeutet dies, dass über Vorgehensmodelle und Adaptionswerkzeuge unterschiedliche Softwareprodukte bedient werden müssen. Das derzeitige Instrumentarium in diesem Bereich ist den neuen Herausforderungen nur bedingt gewachsen, wie das folgende Beispiel zeigt.

Ein Unternehmen der Konsumgüterindustrie setzt für die Abwicklung der internen Prozesse das ERP-System R/3 der SAP AG ein. Die Einführung wurde über das Vorgehensmodell ASAP [SAP00d] durchgeführt, wobei nahezu alle unterstützenden Werkzeuge zur Beschleunigung der Implementierung eingesetzt wurden. Zur Umsetzung des CRM wird aus strategischen Gründen eine e-Business-Application der Siebel Systems Inc. [SIEB00] eingesetzt, die über standardisierte Schnittstellen an das Back Office-Produkt R/3 gekoppelt ist. Die Einführung dieser Softwarelösung stützt sich auf das Vorgehensmodell „Rational Unified Process“ (RUP) der Rational Software GmbH [RATI00]. Für die Betreuung und gezielte Anpassung der Systemlandschaft bedeutet der Einsatz dieser beiden unterschiedlichen Kom-

ponenten eine erhöhte Intransparenz bezüglich der Informationen über den Zustand bzw. die Einstellungen der einzelnen Softwareprodukte. Besonders deutlich wird die mangelnde Transparenz bei dem Einführungsszenario der Siebel-Software. Wichtige Informationen für die Ausprägung der Software, z. B. über die Schnittstellendefinition des bereits eingesetzten ERP-Systems, sind in den Werkzeugen des ASAP-Vorgehensmodells hinterlegt, ein Zugriff von RUP auf dieses Wissen ist nicht möglich. Somit entsteht zusätzlicher Informations- und Koordinationsbedarf, der bei einem integrierten Implementierungsansatz vermieden werden könnte. Dieses einfache Beispiel zeigt das Dilemma der Systemadministration, das durch die zunehmende Öffnung der Softwarelösungen und den damit verbundenen Software-Mix hervorgerufen wird.

Die folgenden Punkte fassen die eben geschilderte Problemsituation zusammen:

- Die Unternehmen erschließen zunehmend neue Geschäftsfelder im Bereich des Electronic Business.
- Auch in diesem Bereich ist eine Abkehr von Individualentwicklungen, ähnlich der Entwicklung in den klassischen ERP-Systemen, zu beobachten.
- Diese Entwicklung induziert eine heterogene Systemlandschaft, die mit Hilfe moderner Austauschformate (z. B. XML) integriert wird.
- Die Einführung und laufende Betreuung der verschiedenen Lösungen wird komplexer.
- Das derzeitige Instrumentarium zur Unterstützung der Softwareimplementierung ist meist für eine Lösung gewidmet, selbst die Werkzeuge eines Herstellers decken nicht das gesamte Produkt-Portfolio ab (z. B. ASAP für das klassische R/3 und ASAP für SAP-CRM).
- Der transparente Zugriff auf die Informationen bzgl. der eingesetzten Systeme ist durch die heterogenen Wissensbasen der Unterstützungswerkzeuge nicht möglich.

Das vornehmliche Ziel zukünftiger Bemühungen muss es demnach sein, eine betriebswirtschaftliche Navigations- und Arbeitsumgebung für die verschiedenen Inhalte der unternehmensweiten Softwarelandschaft zu entwickeln, die eine stetige Verbesserung im Sinne des Continuous System Engineering über die Grenzen einzelner Lösungen hinweg ermöglicht.

1.4 Zielsetzung

Ziel dieser Arbeit ist es, das bestehende ITHAKA-Konzept (siehe Kapitel 2.1.3) zur Unterstützung der Adaption von Softwarebibliotheken um die neuen Anforderungen zu erweitern, die durch die zunehmende Anzahl der integriert eingesetzten Softwarelösungen gestellt werden. Dafür müssen durch eine problemadäquate Vorgehensweise die in Kapitel 1.3 aufgeführten Defizite beseitigt werden. Dazu soll die Ausarbeitung der THESEUS-Methode die Realisierung eines entsprechenden Werkzeugs ermöglichen.

Im einzelnen werden folgende Ziele verfolgt:

- Unterstützung verschiedener Softwarelösungen (z. B. R/3, Intershop [INTE00], Siebel Applications [SIEB00], Navision [NAVI00], PeopleSoft [PEOP00]),
- Einbeziehung unterschiedlicher Vorgehensmodelle und Werkzeuge (z. B. RUP [RATI00], ASAP [SAP00d], LIVE METHOD/Chestra [CHES99]),
- Definition betriebswirtschaftlicher Themen, die mit Hilfe der verschiedenen Softwarelösungen umgesetzt werden,
- Vermeidung unnötiger Werkzeugwechsel und Schaffung einer transparenten Navigationsumgebung sowie
- Offenheit gegenüber Veränderungen durch neue Komponenten oder Werkzeuge.

Durch den Einsatz von THESEUS soll es möglich werden, übergreifende Adaptionenprojekte durchzuführen und verschiedene Unterstützungswerkzeuge abzubilden. Zu diesem Zweck wird eine zentrale Wissensbasis aufgebaut, die alle Informationen bezüglich der unternehmensweiten Softwarelandschaft beinhaltet. Der Zugriff auf Detailinformationen wird über neutrale und rein betriebswirtschaftlich orientierte Wissensträger sichergestellt, die in der Lage sind, mehrere Lösungen bzw. Werkzeuge abzubilden.

Zur Umsetzung der THESEUS-Methode wurde ein Prototyp entwickelt, der vornehmlich die im Rahmen des ITHAKA-Konzepts eingesetzten Werkzeuge (z. B. LIVE KIT Structure, LIVE KIT Power) sowie die SAP-Softwarelösungen (z. B. R/3, SAP-CRM, SAP-B2B) berücksichtigt. Die dargestellten Beispiele beziehen

sich auf diesen Prototyp und beinhalten deshalb größtenteils Zusammenhänge und Interaktionen zwischen diesen Komponenten. Grundsätzlich ist die hier vorgestellte Methode aber für den Einsatz in einer heterogenen Software- und Werkzeuglandschaft konzipiert.

1.5 Aufbau der Arbeit

In **Kapitel 1** erfolgt zunächst eine Einordnung der vorliegenden Arbeit in das wissenschaftliche Themenumfeld. Im Anschluss daran wird das Konzept der betriebswirtschaftlichen Softwarebibliothek in ihrem Wesen und Aufbau erläutert, da es für die weiteren Ausführungen von zentraler Bedeutung ist. Darüber hinaus wird die Motivation zur Entwicklung einer neuartigen Methode für die Unterstützung der softwareübergreifenden Adaption erläutert. Die gesamte Struktur der Arbeit ist in Abbildung 1-1 dargestellt.

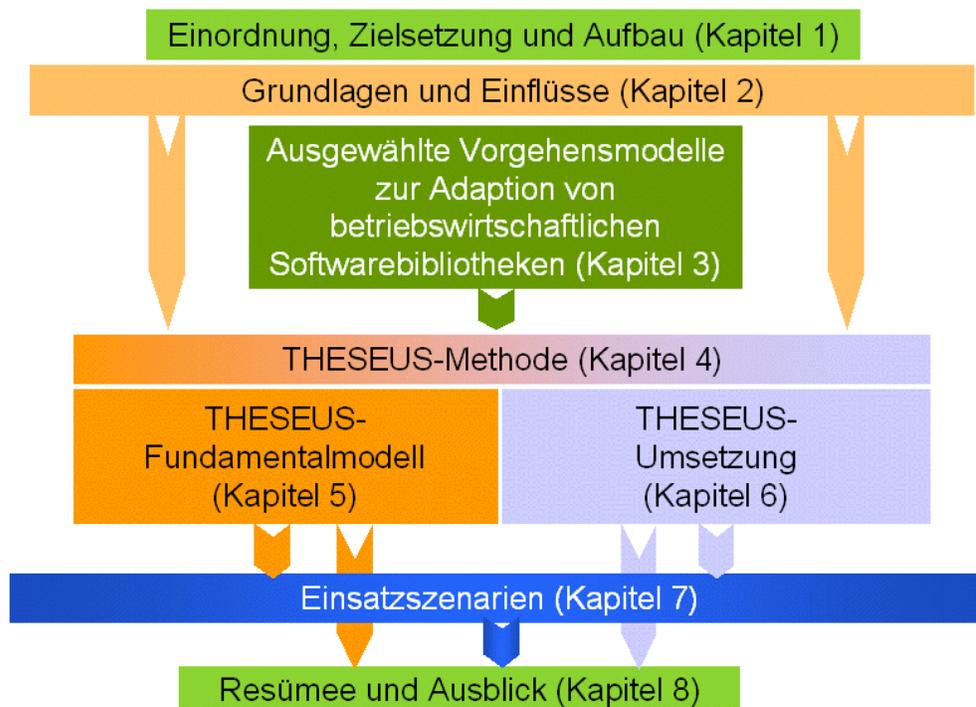


Abbildung 1-1: Aufbau der Arbeit

Kapitel 2 geht auf die Ausgangspunkte und Einflussbereiche ein, die wichtige Erkenntnisse für die Gestaltung der THESEUS-Methode geliefert haben. Dabei werden zunächst die Grundlagen und Begriffe der Adaption erläutert sowie ein Über-

blick über den Stand des Adaptionsumfelds gegeben. Es schließt sich die Vorstellung des von HUGGARD entworfenen ODYSSEUS-Konzepts und dessen Erweiterung durch PENELOPE und MENTOR an. In diesem Zusammenhang werden auch die konkreten Werkzeuge als technische Realisierung der Konzepte kurz beschrieben. Maßgeblichen Einfluss auf die Entwicklung von THESEUS hatten die Vorgehensphilosophie des Continuous System Engineering, Strategien aus dem Bereich des Wissensmanagements und die Nutzung der Mind Mapping-Technik. Diese Themen werden ebenfalls in ihren Grundzügen skizziert, um das Verständnis für die neue Methode zu fördern.

Kapitel 3 gibt einen Überblick über zwei Vorgehensmodelle zur Adaption von Softwarebibliotheken. Diese Modelle werden bezüglich der in diese Arbeit behandelten Problemstellung bewertet und es erfolgt die Ableitung von Prinzipien zur Gestaltung einer softwareunabhängigen Navigationsumgebung.

Kapitel 4 stellt die THESEUS-Methode detailliert dar. Dabei werden zunächst die Architektur und die Komponenten vorgestellt, bevor im Anschluss daran eine neuartige Form der Adaptionobjekte in Gestalt von Wissenspaketen definiert wird.

Die Abbildung und der Aufbau eines Fundamentalmodells auf Basis der in Kapitel 4 entwickelten Wissenspakete ist Gegenstand von **Kapitel 5**. Die Ausführungen umfassen dabei sowohl die allgemeinen Gestaltungsprinzipien des entwickelten Werkzeugs als auch eine detaillierte Beschreibung der einzelnen Monitore, die für die operative Unterstützung der Adaption notwendig sind.

Kapitel 6 zeigt die Umsetzung von THESEUS. Neben der methodischen Umsetzung wird auch auf technische Realisierungsmöglichkeiten eingegangen. Den Abschluss bildet die Darstellung und Beschreibung des Datenmodells.

In **Kapitel 7** werden die Nutzung von THESEUS im Rahmen von Adaptionprojekten diskutiert und typische Einsatzszenarien skizziert. Darüber hinaus wird gezeigt, wie THESEUS bei umfangreichen Projekten im Umfeld von Konzernunternehmen eingesetzt werden kann und welche Vorteile sich in diesem Bereich gegenüber der bisherigen Vorgehensweise ergeben.

Die Prüfung, ob THESEUS den formulierten Ansprüchen und Zielsetzungen gerecht wird, ist Gegenstand von **Kapitel 8**. Es schließt sich die Betrachtung zukünftiger Entwicklungen an, welche auf den Grundlagen des Ansatzes aufbauen und Erweiterungs- bzw. Ergänzungspotenzial beinhalten.

Anhang A zeigt eine Aufstellung von Objekttypen, die im Fundamentalmodell abgelegt sind.

Anhang B stellt die mit dem Prototypen entwickelten Wissenspakete für das Anwendersegment Versorgungsindustrie vor.

Anhang C beinhaltet einige Beispiele aus dem entwickelten Prototypen. Der Schwerpunkt liegt dabei auf der Abbildung der Projektverwaltung, da die einzelnen Monitore in Kapitel 5 beschrieben sind.

Das komplette Datenmodell der THESEUS-Datenbank, d. h. das integrierte Fundamental-Datenmodell über alle Monitore, ist in **Anhang D** dargestellt.

2 Grundlagen und Einflüsse

In Verbindung mit der Implementierung und dem Einsatz einer Softwarebibliothek werden sehr viele weitere Themen diskutiert, die maßgeblichen Einfluss auf das im Rahmen dieser Arbeit entwickelte Konzept zur themenorientierten Umsetzung der Adaptionsergebnisse haben. Die wichtigsten werden hier in ihren Grundzügen vorgestellt. Es werden jedoch nur die relevanten Aspekte behandelt, für detaillierte Informationen sei an dieser Stelle auf die jeweils angegebene Literatur verwiesen.

Die folgenden Ausführungen dienen

- der Vorstellung von konkreten Methoden und Konzepten, die für die Gestaltung dieser Arbeit von Bedeutung sind,
- der Vermeidung von Missverständnissen in Bezug auf die jeweils betrachteten Inhalte,
- der Schaffung einer gemeinsamen Kommunikationsbasis und
- dem Aufzeigen der Zusammenhänge zwischen den angesprochenen Themen.

Teilweise haben die beschriebenen Konzepte keinen direkten Bezug zur Adaption einer betriebswirtschaftlichen Softwarebibliothek, in diesen Fällen wird neben der inhaltlichen auch eine konzeptionelle Darstellung geliefert, welche die Beziehungen der betrachteten Themen hervorhebt.

Ausgehend von der grundsätzlichen Beschreibung der Adaption einer Softwarebibliothek (2.1) wird das Konzept zur kontinuierlichen Anpassung der betrieblichen Informationsverarbeitung und Organisation - das Continuous System Engineering - vorgestellt (2.2). Nach diesen Grundlagen werden die maßgeblichen Einflussgebiete näher erläutert, die sich in die Bereiche Wissensmanagement (2.3) und Mind Mapping (2.4) gliedern, bevor dann die gewonnenen Erkenntnisse zusammengefasst werden (2.5).

2.1 Adaption betriebswirtschaftlicher Softwarebibliotheken

Nachdem in Kapitel 1.2.2 die betriebswirtschaftliche Softwarebibliothek vorgestellt wurde, ist deren Adaption nun Gegenstand dieses Kapitels. Die folgenden Ausführungen beinhalten allgemeine Erkenntnisse zur Adaption betriebswirtschaftlicher Standardanwendungssoftware (SASW), welche die Grundlage für die Entwicklung spezieller Methoden und darauf basierender Werkzeuge bilden. Es bestehen Gemeinsamkeiten mit einer Reihe von anderen Arbeiten aus diesem Umfeld - beispielsweise VOGELSANG [VOGE97, S. 46-69], WEDLICH [WEDL97, S. 30-62] und STRELLER [STRE99, S. 26-54] -, da sie auf der kollektiven Grundlage beruhen.

Die Ausführungen sind bewusst kurz gehalten und es wird lediglich auf Punkte detailliert eingegangen, auf die in den nachfolgenden Kapiteln Bezug genommen wird. Für weitergehende Informationen in diesem Zusammenhang siehe THOME [THOM96], HUFgard [HUG94], VOGELSANG [VOGE97], WEDLICH [WEDL97] und BÄTZ [BÄTZ99].

Die Anpassung einer SASW an die Anforderungen eines Kunden durch die Veränderung von Parametern wird als Customizing bezeichnet. Durch eine große Zahl von Einstellungen besteht die Möglichkeit, den Ablauf der Programmlogik zu verändern. Darüber hinaus ist es oftmals notwendig, auch in den Stammdaten entsprechende Einstellungen vorzunehmen, wie z. B. die Festlegung der Dispositionsart pro Material. Eine durchgängige Vereinfachung des Customizing konnte trotz zahlreicher Versuche und Hilfsmittel nicht erreicht werden [THOM98, S. 52f.]. Auf der einen Seite fehlte die Verknüpfung der Werkzeuge mit der Ebene der Steuerparameter, andererseits nimmt die Transparenz von Hilfsmitteln, die einen hohen Detaillierungsgrad bezüglich der Einstellungen besitzen, mit Zunahme des Funktionsumfangs und damit der Zahl der Parameter ab. Die Folge dieser Entwicklung sind eine mangelnde Transparenz, verlängerte Einführungszeiten und Resignation der Anwender, die einen dem einer Individualentwicklung vergleichbaren Anpassungsaufwand befürchten [THOM98, S. 53].

Unter Adaption als eine Vorgehensweise zur Anpassung von SASW wird der Prozess der Auswahl, Anpassung und Ergänzung aus einer betriebswirtschaftlichen Softwarebibliothek verstanden, der über das reine Einstellen von Parametern hinausgeht. Der Ausgangspunkt sind die betriebswirtschaftlichen Anforderungen des

betrachteten Unternehmens. Dabei handelt es sich nicht um einen einmaligen, sondern um einen kontinuierlichen Vorgang, da sich die Anforderungen im Zeitablauf ändern. Die Adaption muss demnach beliebig fortgeführt werden können, ohne dass an einem operativ genutzten System Inkonsistenzen entstehen [HUFG97a, S. 5].

2.1.1 Adaptionarten

HUFGARD unterscheidet drei Formen der Adaption, die in einer sequentiellen Reihenfolge durchlaufen werden.

- Erster Schritt ist die Auswahl aus der Softwarebibliothek. Die Identifikation der passenden Elemente und Bestandteile ist die einfachste und wirtschaftlichste Form der Adaption. Von der Softwarebibliothek setzt dies eine ausgeprägte Modularisierung und Kombinationsmöglichkeit der Bereiche bis hin zu den Funktionen voraus.
- Im zweiten Schritt erfolgt die Anpassung der ausgewählten Bestandteile im vordefinierten Rahmen. Durch die Einstellung von steuernden Parametern, die Aktivierung von Datenobjekten bzw. -beziehungen oder durch die Anpassungen der Darstellungsweise werden die benötigten Bereiche benutzer-spezifisch verändert. Dabei bewegen sich die Modifikationen allerdings in einem durch den Softwareproduzenten festgelegten parametrisierbaren Bereich. Als Folge steigen die Anforderungen an das Datenmodell und die Entwicklung der SASW, da möglichst viele sinnvolle Kombinationsmöglichkeiten zur Verfügung gestellt werden sollten.
- Der Adaptionsprozess wird mit der Ergänzung individueller Funktionen und Vorgänge abgeschlossen, indem Lücken gefüllt werden, die nicht durch die Softwarebibliothek geschlossen werden können. Diese entstehen, falls für bestimmte Anforderungen keine Funktionalität vorhanden ist oder diese nicht ausreichend abgedeckt werden. Abhilfe kann durch die Modifikation vorhandener Funktionen, die Eigenentwicklung zusätzlicher Bestandteile oder die Anbindung von Fremdprogrammen über Schnittstellen geschaffen werden [HUFG94, S. 178-182].

2.1.2 Adaptionswerkzeuge

Werkzeuge unterstützen und erleichtern die Lösung einer Aufgabenstellung. Im Umfeld von Softwareentwicklung und -einsatz handelt es sich dabei meist wiederum um Programme, die auf einem Verfahren oder einer Methode beruhen [THOM90, K1-2]. Das Ziel von Adaptionswerkzeugen ist eine durchgängige Unterstützung des Anpassungsprozesses, wobei aus Sicht der softwaretechnischen Umsetzung folgende Arten unterschieden werden können:

- Werkzeuge, die Bestandteil der eingesetzten Entwicklungsumgebung sind. Mit ihnen werden Grundeinstellungen für weitere Adaptionmöglichkeiten vorgenommen und die Voraussetzungen für weitere Hilfsmittel geschaffen.
- Hilfsmittel, die als fester Bestandteil einer Softwarebibliothek zur Verfügung stehen oder die als zusätzliche Werkzeuge Unterstützung bieten und über Schnittstellen mit der SASW verbunden sind, z. B. das LIVE KIT Structure für die SAP-Software [VOGE97, S. 48f.].

Die betriebswirtschaftliche Ausrichtung ist bei eigenständigen Werkzeugen, die auf einer softwareunabhängigen Vorgehensweise basieren, stärker ausgeprägt. Darüber hinaus sind diese aufgrund der Entkopplung von der SASW dezentral verwendbar und können bereits benutzt werden, bevor eine Entscheidung für die SASW getroffen wird oder diese verfügbar ist. Somit können die Tools bereits bei der Anforderungsanalyse eingesetzt werden und den Adaptionsprozess begleiten. Dabei sollten die Informationen kontinuierlich gesammelt werden und in allen Phasen zur Verfügung stehen, um das Ziel einer durchgängigen Unterstützung der gesamten Adaption zu erreichen.

2.1.3 ITHAKA-Konzept

Die von HUFGARD entwickelte Methodik ITHAKA (Prozess- und strukturIntegrierte, Toolgestützte, Heuristische Architektur der Kundenspezifischen Adaption von Softwarebibliotheken) umfasst als Rahmen mehrere spezielle Ansätze zur Unterstützung der Adaption betriebswirtschaftlicher Softwarebibliotheken. Dazu zählen u. a. ODYSSEUS, PENELOPE, MENTOR und OLYMP (vgl. Abbildung 2-1).

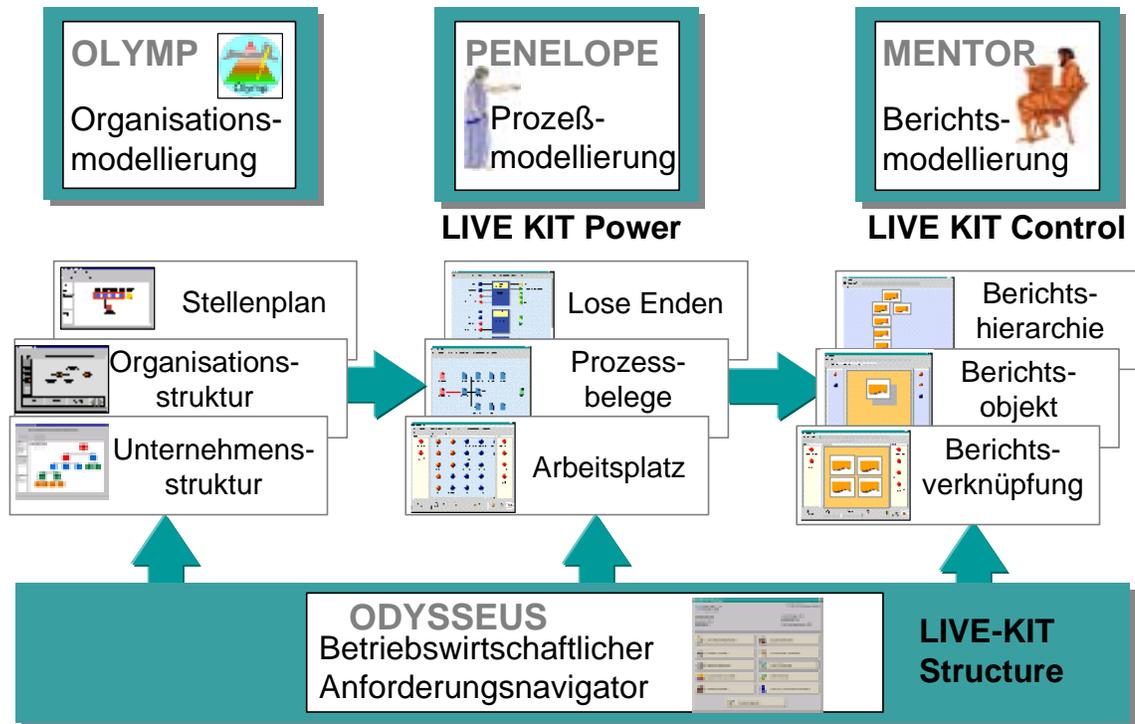


Abbildung 2-1: Bestandteile von ITHAKA [in Anlehnung an VOG97, S. 224]

2.1.3.1 WISSENSCHAFTLICHE ARBEITEN IM ADAPTIONSUMFELD

Abbildung 2-2 zeigt die wichtigsten der bisher am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik von Prof. Dr. R. Thome an der Bayerischen Julius-Maximilians-Universität Würzburg entstandenen Diplomarbeiten und Dissertationen aus dem ITHAKA-Umfeld. Ziel dieser Arbeiten ist es, einen gesamtheitlichen wissenschaftlichen Ansatz zur Unterstützung der Einführung von Standardsoftware zu liefern.

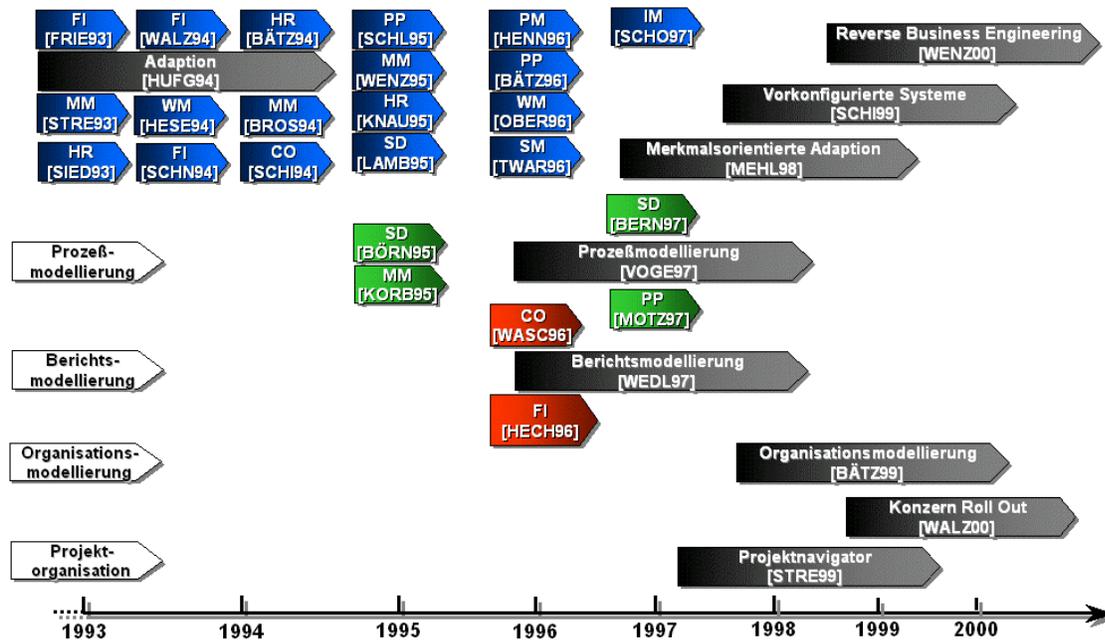


Abbildung 2-2: Arbeiten zur Adaption betriebswirtschaftlicher Softwarebibliotheken [in Anlehnung an VOG97, S. 50]

Durch die aufgelisteten Arbeiten wird eine nahezu vollständige Abdeckung der fachlichen Bereiche und der Konzeption von Methoden zur Anforderungsanalyse, der Gestaltung der Aufbau- und Ablauforganisation, der Berichtsmodellierung, der branchenspezifischen Unterstützung sowie der Gestaltung der organisatorischen Projektentwicklung erreicht.

2.1.3.2 BESTANDTEILE

In Abbildung 2-1 sind einige der bereits vorhandenen Komponenten im Zusammenhang dargestellt. Grundlage des Vorgehens bildet das ODYSSEUS-Konzept, das die Erfassung aller Unternehmensanforderungen unterstützt. Die dabei ermittelten Informationen dienen in einem weiteren Schritt der Darstellung und Modellierung von Geschäftsprozessen durch PENELOPE. Analog hierzu ermöglichen MENTOR eine Analyse der verfügbaren Standardberichte und OLYMP die Modellierung der Organisationsstruktur. An dieser Stelle wird lediglich auf die drei grundlegenden Konzepte eingegangen, andere Ansätze, wie z. B. OLYMP, MEDEA oder SPARTA, werden in späteren Kapiteln beschrieben.

ODYSSEUS

Bei ODYSSEUS handelt es sich um eine speziell für den Mittelstand entwickelte Adaptionstrategie. Ziel ist ein schneller Abgleich der Anforderungen eines Unternehmens mit dem Potenzial einer Softwarebibliothek. Die Vorgehensweise umfasst drei Schritte:

Umfang ermitteln:

Grundlage der Auswahl ist eine Gliederung des betriebswirtschaftlichen Inhalts der Softwarebibliothek in Fachbereiche und Komponenten. Unterstützt durch ein Standard-/Options-Prinzip wird der Projektumfang und damit die Zahl der zu beantwortenden Fragen festgelegt. Dies kann sehr schnell durchgeführt werden, da die vorhandenen Optionen offengelegt werden und durch die in einem Expertensystem definierten Regeln gewährleistet wird, dass verpflichtende Bereiche automatisch aktiviert werden.

Reduktion nicht benötigter Bestandteile:

Im nächsten Schritt wird durch die Beantwortung von Fragen nicht benötigte Funktionalität abgewählt. Durch diese Reduktion wird die Transparenz erhöht und somit die Komplexität verringert.

Auswahl typischer Lösungen:

Ergänzt wird die Reduktion durch die Möglichkeit, aus einer Sammlung typischer Lösungen auszuwählen, die in der Softwarebibliothek eingestellt werden können. Durch die Dokumentation der durchzuführenden Anpassungen in den Zuordnungsberichten wird die Verbindung mit der Ebene der Parameter hergestellt, wobei alle Lösungen aus betriebswirtschaftlicher Sicht dargestellt werden. Vor allem in Verbindung mit den vorkonfigurierten Systemen zur Abbildung spezieller Branchenfunktionalitäten spielt die Bereitstellung fertiger Alternativen eine zentrale Rolle.

Alle drei Schritte werden durch ein Expertensystem unterstützt. Die logischen Beziehungen ermöglichen einen interaktiven Anforderungsabgleich durch die Beantwortung der in einem Leitfaden angeordneten Fragen und Zuordnungselemente. Anhand der gegebenen Antworten werden Informationen abgeleitet, um Fachbereiche und Komponenten zu aktivieren bzw. zu deaktivieren, um überflüssige Fra-

gen automatisch zu beantworten und auszublenden sowie um typische Lösungen an- bzw. abzuschalten [THOM96, S. 95-103; erstmals bei HUG94, S. 215-231].

In Abbildung 2-3 sind die drei Schritte, die beim Anforderungsabgleich mit ODYSSEUS durchlaufen werden, in Form eines Leitfadens dargestellt. Während der Bearbeitung werden auf allen Ebenen Konsistenzprüfungen durch das Expertensystem vorgenommen, um ein aus Sicht der SASW ablauffähiges Ergebnis zu gewährleisten.

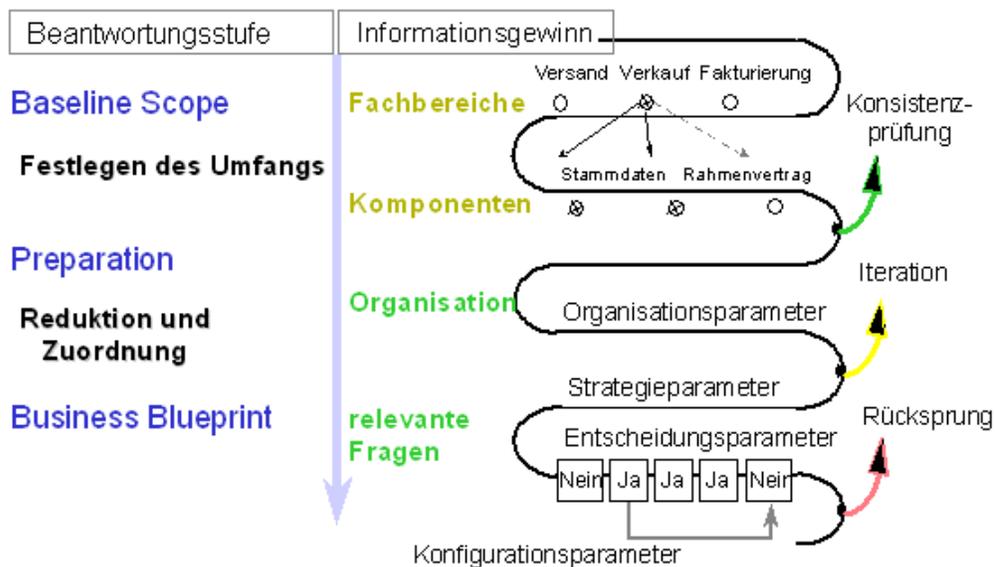


Abbildung 2-3: Anforderungsabgleich mit dem betriebswirtschaftlichen Anforderungsnavigator ODYSSEUS [in Anlehnung an THOM96, S. 100]

Die technische Umsetzung von ODYSSEUS erfolgt seit 1995 mit der Entwicklung des Werkzeugs LIVE KIT Structure. Für eine weitergehende Beschreibung sei auf die Arbeiten von STRELLER [STRE99, S. 29-52] und VOGELANG [VOGE97, S. 49-76] verwiesen.

PENELOPE

Zentraler Gegenstand von PENELOPE ist die Visualisierung der Ablauforganisation einer Softwarebibliothek. Zu diesem Zweck besteht das Konzept aus der Schnittstellen-, der Prozessablauf- und der Prozessdaten-Analyse. Mit diesen drei Bestandteilen erfolgt eine Modellierung der Prozesse mit unterschiedlichen Detaillierungsgraden. Zur Darstellung und Untersuchung der Abläufe stehen in jeder

Analysephase ein oder mehrere Monitore zur Verfügung. Jeder Monitor ist auf eine konkrete Fragestellung ausgerichtet und bietet spezifische Darstellungselemente. Insgesamt sind folgende Monitore vorhanden:

- Schnittstellen-,
- Prozessbeleg- und
- Arbeitsplatzmonitor.

Im Gegensatz zu konventionellen Modellierungswerkzeugen, wie ARIS [IDS00], dient PENELOPE nicht nur zur Veranschaulichung eines Referenzmodells oder zur manuellen Modellierung einer individuellen Situation. Durch die Integration zu ODYSSEUS besteht die Möglichkeit einer anpassenden Referenzmodellierung, indem durch die Ergebnisse des Anforderungsabgleichs die Referenzmodelle des Fundamentalmodells automatisch kundenspezifisch konfiguriert werden und auf diese Weise ein Individualmodell entsteht (siehe Abbildung 2-4).

Nach der Anpassung sind in den einzelnen Monitoren nur noch die Elemente und Verknüpfungen vorhanden, die beim Anforderungsabgleich nicht abgewählt bzw. die als typische Lösungen aktiviert worden sind. Damit entsteht ein kundenspezifisches Modell, das in einem zweiten Schritt, im Adaption-Monitoring, auf Unterbrechungen im Prozessablauf untersucht werden kann. Werden solche Lücken entdeckt, kann gezielt ermittelt werden, ob die Beantwortung im Anforderungsabgleich geändert werden muss bzw. falls keine solche Funktionalität vorhanden ist, inwieweit Prozessmodifikationen und Ergänzungsentwicklungen benötigt werden. Das Ziel des kombinierten Einsatzes von ODYSSEUS und PENELOPE ist es, die Anforderungen eines Unternehmens den Adaptionkategorien Auswahl, Anpassung und Ergänzungsentwicklung zuzuordnen, um schnell ein Ausgangssystem (Baseline-System) erstellen zu können, welches anschließend sukzessive verbessert und um Ergänzungsentwicklungen angereichert wird [vgl. VOGÉ97, S. 188f.].

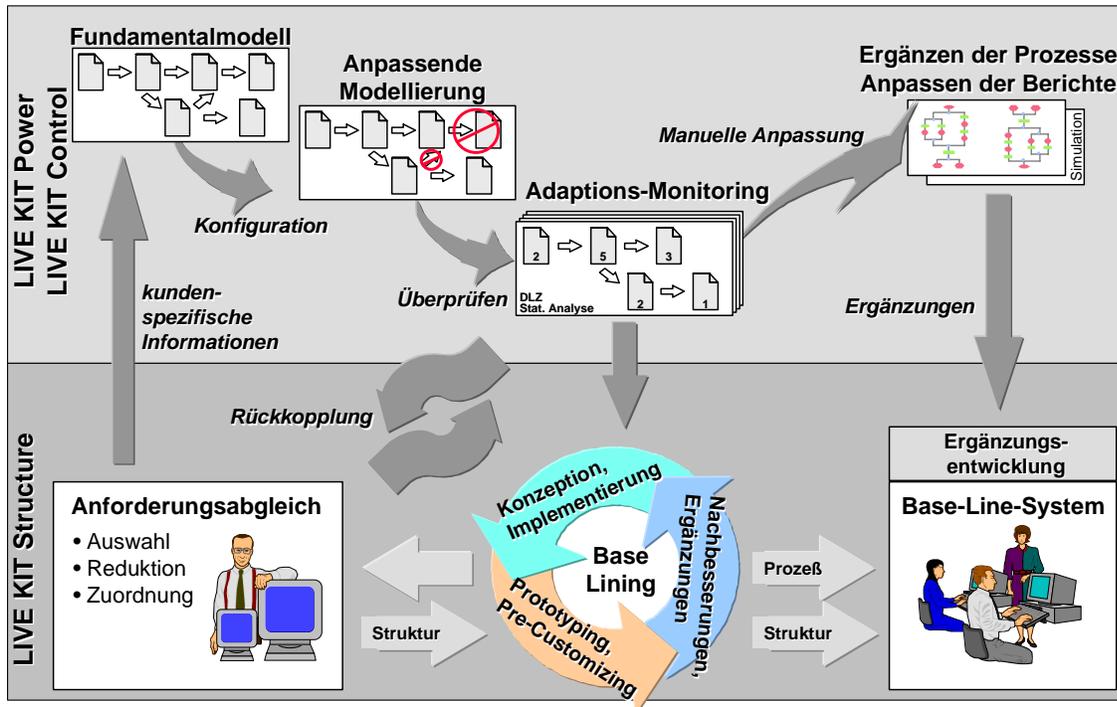


Abbildung 2-4: Zusammenhang von LIVE KIT Structure, LIVE KIT Power und LIVE KIT Control [VOGE97, S. 189]

Die technische Umsetzung des PENELOPE-Konzepts erfolgt seit 1997 mit der Entwicklung des Werkzeugs LIVE KIT Power durch die IBIS Prof. Thome AG und die Siemens Business Services GmbH & Co OHG.

Mit den Regeln des Expertensystems von LIVE KIT Structure können nach der Beantwortung der ausgewählten Fragen die Informationen evaluiert werden, die zur anpassenden Referenzmodellierung benötigt werden.

Die in Abbildung 2-5 dargestellten Bildschirmausschnitte verdeutlichen die in LIVE KIT Power realisierten Reduktionsmöglichkeiten. Einzelne Objekte, wie Kernprozesse, Schnittstellen oder Prozessbelege können bei der Erstellung eines Individualmodells automatisch oder manuell deaktiviert werden.

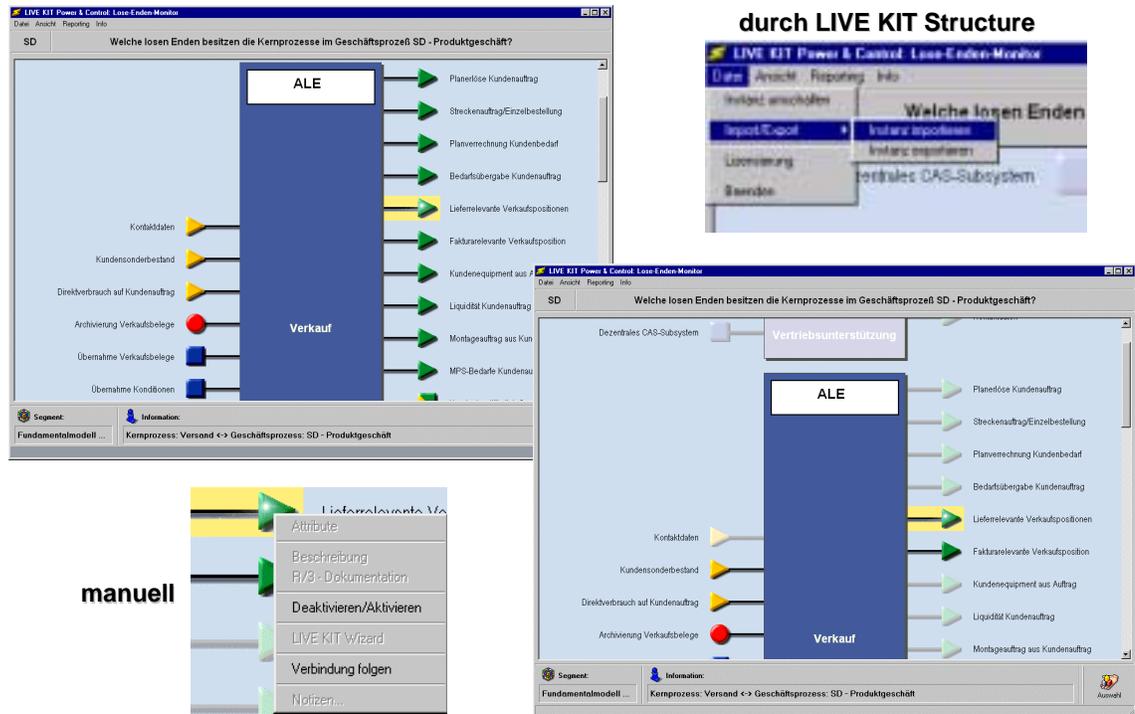


Abbildung 2-5: Möglichkeiten zur Reduktion des Fundamentalmodells am Beispiel von LIVE KIT Power

MENTOR

Betriebswirtschaftliche SASW wird mit einer großen Zahl von Standardberichten und Auswertungen ausgeliefert. Da solche Systeme aber nach der Installation über keine Bewegungs- oder Stammdaten verfügen, kann nicht überprüft werden, inwieweit die bereits vorhandenen Berichte den Anforderungen der Anwender entsprechen. Mit MENTOR wird dieses Defizit ausgeglichen. Zu diesem Zweck werden die Standardberichte einer SASW klassifiziert, die zugrundeliegenden betriebswirtschaftlichen Verfahren beschrieben und Berichtshierarchien gebildet. Auch hier werden zur Darstellung und Analyse Monitore benutzt:

- Berichtsobjekt-,
- Berichtsverknüpfungs- und
- Berichtshierarchiemonitor.

Der Berichtsobjektmonitor stellt die Klassifikation des Berichts, z. B. die Zielgruppe, den betriebswirtschaftlichen Hintergrund und das Layout dar. Da für jeden Be-

richt ein Screenshot mit Beispieldaten aufgerufen werden kann, besteht die Möglichkeit, die Berichte auch ohne verfügbares System anzuzeigen. So kann bereits in einer frühen Phase eines Projekts festgelegt werden, ob das Layout einer Auswertung individuell angepasst werden muss.

Für den Monitor stehen zwei Sichten zur Verfügung:

- Der Berichtsobjektmonitor mit vor- und nachgelagerten Berichten und
- die Anzeige einzelner Berichte mit der Datenherkunft und dem Ausführungszeitpunkt.

Mit der ersten Sicht wird die Aggregation der Informationen verdeutlicht, indem die vor- bzw. nachgelagerten Berichte im entsprechenden Berichtsobjektmonitor aufgerufen werden können. Dies ermöglicht eine sequentielle Navigation durch die Berichtshierarchie.

Die zweite Sicht stellt durch Ereignisse und Funktionen, welche die entsprechenden Ausführungszeitpunkte determinieren bzw. die benötigten Informationen erzeugen, eine Verbindung zu den LIVE KIT Power Arbeitsplätzen her.

Durch den Berichtsverknüpfungsmonitor wird die direkte Verbindung zu den Arbeitsplätzen hergestellt. In diesem Monitor werden aus Sicht eines Arbeitsplatzes, beispielsweise für einen Auftragssteuerer, alle Berichte angezeigt, die zur Durchführung der dortigen Aufgaben benötigt werden. Zur Erhöhung der Übersichtlichkeit sind die Berichte nach Themen sortiert. Auf diese Weise erfolgt die Verknüpfung von Ablauforganisation und Berichtswesen.

Gegenstand des Berichtshierarchiemonitors ist die Koordination von Informationen, d. h. die Reihenfolge, in der die Auswertungen ausgeführt werden müssen, um zu einem betriebswirtschaftlich sinnvollen Ergebnis zu kommen.

Analog der Vorgehensweise für die Ablauforganisation ist auch MENTOR der Methode ODYSSEUS nachgelagert. Hier werden allerdings im Fundamentalmodell Berichte und Berichtshierarchien, geordnet nach Prozessbereichen, Kernprozessen, Arbeitsplätzen und Themen vordefiniert und verwaltet. Basierend auf den Informationen von ODYSSEUS kann mit der anpassenden Referenzmodellierung das Individualmodell erzeugt werden, in dem nur noch die sinnvoll einsetzbaren Standardberichte enthalten sind. Im Adaption-Monitoring muss dann überprüft werden, ob die verbliebenen Berichte den Anforderungen gerecht werden oder an-

gepasst werden müssen bzw. welche Standardauswertungen reduziert wurden, aber trotzdem benötigt werden (siehe Abbildung 2-4). MENTOR bildet also die Basis, um zu überprüfen, ob die Anforderungen an das betriebliche Berichts- und Informationswesen erfüllt werden.

Die technische Realisierung erfolgt seit 1997 mit der Entwicklung des Werkzeugs LIVE KIT Control durch die IBIS Prof. Thome AG und die Siemens Business Services GmbH & Co OHG.

2.2 Continuous System Engineering

Das von THOME und HUGGARD entwickelte Konzept des Continuous System Engineering (CSE) sieht in der kontinuierlichen Anpassung von betrieblicher Informationsverarbeitung und Organisation bezüglich der Prozesse und den organisatorischen Strukturen den Schlüssel zum Erfolg [THHU96, S. 78]. Die Idee für diesen Ansatz liegt hauptsächlich darin, dass sich die Unternehmen in der heutigen Zeit ständig an veränderte Marktbedingungen anpassen müssen, um konkurrenzfähig zu bleiben. Dies wird durch die immer kürzer werdenden Zyklen zur Steigerung der Leistungsfähigkeit bestehender Produkte und die damit verbundenen Entwicklungszyklen für neue Technologien im Bereich der Informationsverarbeitung verschärft. Diese Notwendigkeit zu einem ständigen Wandel wirkt sich ebenfalls auf die organisatorischen Abläufe eines Unternehmens aus. Eine schnelle Einführung der Softwarebibliothek verhindert, dass eine aufwendige und langwierige Analyse und Neugestaltung der Geschäftsprozesse durch die sich ändernden Rahmenbedingungen bereits vor Einsatz der angestrebten Softwarelösung unbrauchbar ist. Dem CSE liegt also die Annahme zugrunde, dass den oben beschriebenen Herausforderungen nur durch einen kontinuierlichen Anpassungsprozess begegnet werden kann [THHU96, S. 78].

Das CSE verfolgt das Ziel, die Gestaltung der im Unternehmen ablaufenden Prozesse zu vereinfachen, vor allem aber die stetige Anpassung an sich ändernde Rahmenbedingungen sicherzustellen. Ein wesentlicher Faktor ist dabei die Integration der Betriebswirtschaft in Form von organisatorischen Anforderungen und der Informationstechnologie. Der Schwerpunkt liegt jedoch mehr auf einer evolutionären und weniger auf einer revolutionären Vorgehensweise. Zunächst soll möglichst schnell über ein funktionierendes Informationssystem verfügt werden können,

welches dann im Rahmen der Schritte und des Einsatzes des CSE weiter verbessert wird.

Das CSE ist nur unter den folgenden zwei Grundvoraussetzungen realistisch umsetzbar:

- Eine organisatorische Änderung darf keine aufwendige Softwareentwicklung oder Modifikation der Standardanwendungssoftware auslösen.
- Die Beschreibung der organisatorischen Notwendigkeiten muss in einer schnellen und für den Implementierungsprozess verwendbaren Form erfolgen.

Grundlage für eine erfolgreiche Umsetzung des CSE sind die Verfügbarkeit einer Softwarebibliothek mit einem breiten betriebswirtschaftlichen Methodenpotenzial und dynamischer Adaptionfähigkeit sowie eines Anforderungsnavigators, der das Beschreiben des Unternehmens, seiner Anforderungen und Ziele durch die Zerlegung, Reduktion und Zuordnung vereinfacht und in Form einer Individualbibliothek übergibt [THHU96, S. 88].

Durch die Klassifizierung der Prozessabläufe wird die Identifizierung und Herausarbeitung der wirklich notwendigen Hauptprozesse zur Unterstützung der wesentlichen Aufgaben des Unternehmens erleichtert. THOME und HUFGARD verweisen zudem darauf, dass die durch die Verwendung einer Standard-Softwarebibliothek eingeführten Ablaufstrukturen nicht in jedem Unternehmen dieselben sind. Vielmehr wird durch die Vielzahl an Einstellungsmöglichkeiten erreicht, dass für jedes Unternehmen eine spezifische Aufbau- und Ablaufstruktur möglich wird. Die Vorteile des CSE gegenüber den anderen Vorgehensweisen zur Softwareentwicklung bzw. -adaption lassen sich wie folgt zusammenfassen [vgl. THHU96, S. 84]:

1. Das Informationssystem läuft bereits nach einer kurzen Adaptions- und Installationsphase produktiv und unterstützt somit das betriebliche Geschehen frühzeitig.
2. Durch die Softwarebibliothek wird das Unternehmen in die Lage versetzt, jederzeit bestehende Prozesse zu ändern oder zu erweitern.

3. Die konsequente und jederzeit aktuelle Beschreibung der letzten Einstellung bezüglich des Zustands der Informationsverarbeitung, d. h. welche Prozesse und Organisationsstrukturen auf welche Art im System abgebildet sind, bietet eine solide Basis für weitere Überlegungen das betriebliche Ablaufgeschehen zu überdenken.

2.3 Wissensmanagement

Gegenstand dieses Kapitels ist die Einordnung des Adaptionsumfeldes in das unternehmensweite Wissensmanagement. Ziel der Betrachtungen ist vor allem die effektive Nutzung der Methoden und Werkzeuge aus dem Bereich des Knowledge Managements und deren Bedeutung im Lebenszyklus einer betriebswirtschaftlichen Softwarebibliothek. Die Ausführungen erheben bezüglich der Thematik Wissensmanagement keinen Anspruch auf Vollständigkeit, sie stellen nur einen Auszug dar. Schwerpunkt ist der Bereich des technologischen Wissensmanagements, der sich hauptsächlich auf operativer Ebene abspielt [GENT99, S. 122].

Zunächst werden die für die Themenstellung relevanten Grundzüge des Wissensmanagements erläutert (2.3.1), bevor dann im folgenden ausgewählte Strategien und Werkzeuge näher untersucht werden (2.3.2), die für die im Rahmen dieser Arbeit entwickelte Methode genutzt werden können.

2.3.1 Grundzüge des Wissensmanagements

Ein zentrales Anliegen der Unternehmen ist die effiziente Verwaltung von Wissen, das Knowledge Management. In vielen Unternehmen ist das Hauptproblem die Abhängigkeit von einigen wenigen Wissensträgern. Dies ist besonders dann von Bedeutung, wenn diese aus dem Unternehmen ausscheiden und das verfügbare Wissen auf diese Weise verloren geht [BOLL99, S. 181]. Neben zahlreichen anderen Theorien hat sich das Wissensmanagement als eigenständiges Managementkonzept etabliert und gewinnt zunehmend an Bedeutung. Der Produktionsfaktor Wissen wird immer mehr als Wettbewerbsvorteil gesehen. Dieses umfassende Konzept schließt alle Informationsquellen und -prozesse eines Unternehmens ein, hinsichtlich der Zielsetzung dieser Arbeit beziehen sich die Ausführungen aber lediglich auf Informationsquellen für die Adaption von Softwarebibliotheken.

Im Vorfeld der Ausführungen muss kurz auf die Ähnlichkeiten und Unterschiede von Wissen und Information eingegangen werden, um eine Abgrenzung der beiden Begriffe vornehmen zu können. Entscheidend sind in diesem Zusammenhang drei Punkte [NONA97, S. 70]:

1. Im Gegensatz zu Information befasst sich Wissen mit Vorstellungen und Engagement. Wissen ist die Konsequenz einer bestimmten Einstellung, Perspektive oder Absicht.
2. Wissen beinhaltet Handeln und ist immer zweckgerichtet.
3. Wissen ist kontextspezifisch und hat immer eine Bedeutung.

Obwohl die Begriffe Information und Wissen in ihrer Verwendung oft nicht auseinandergehalten werden, gibt es eine klare Unterscheidung zwischen beiden. „Informationen liefern neue Gesichtspunkte zur Interpretation von Geschehnissen oder Dingen und enthüllen zuvor nicht erkannte Bedeutungen und Zusammenhänge.“ [NONA97, S. 70] Information bildet somit ein notwendiges Medium oder Material zur Bildung von Wissen. Demnach ist Information „ein Fluss von Botschaften, der im Zusammentreffen mit den Vorstellungen und dem Engagement eines Menschen Wissen erzeugt.“ [NONA97, S. 70] Die Transformation von Informationen in Wissen spielt im Rahmen der hier entwickelten Methode eine untergeordnete Rolle, da sich der Wissensgehalt im Umfeld der Softwareadaption direkt aus den strukturierten Informationen (z. B. Analyseergebnisse) ableitet, daher werden die beiden Begriffe in der folgenden Ausarbeitung synonym verwendet.

Der Adaptionprozess einer Softwarebibliothek generiert eine große Menge an nutzbarem Wissen. Im Allgemeinen unterscheidet man dabei individuelles [ALBR93, S. 34-52] von organisatorischem Wissen [SCHÜ96, S. 83]. Individuelles Wissen ist die Informationsverarbeitung durch das Bewusstsein einer einzelnen Person. Es ist eng mit der jeweiligen Person verknüpft und beeinflusst deren Handlung. Organisatorisches Wissen hingegen ist das Wissen einer Organisation bzw. eines Unternehmens. Dieses wiederum muss weiter in explizites und implizites Wissen unterteilt werden. Das implizite Wissen befindet sich in den Köpfen der Mitarbeiter und ist in der Regel nicht für alle Mitarbeiter eines Unternehmens zugänglich. Explizites Wissen hingegen ist transparent dokumentiert und nimmt konkrete Formen an. Diese beiden Formen lassen sich noch in weitere Teilbereiche, wie z. B. das Wissen über Patente, Kunden, Prozesse oder Produkte, unter-

gliedern. Abbildung 2-6 zeigt die einzelnen Komponenten des Wissensmanagements.

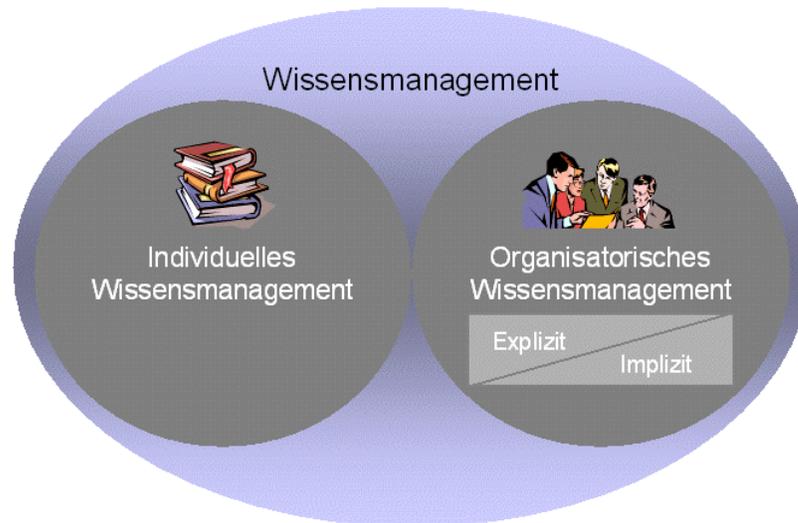


Abbildung 2-6: Komponenten des Wissensmanagements

SERVATIUS betont, dass die Ressource Wissen eine immer wichtigere Komponente für ein erfolgreiches Wachstum und die Wertsteigerung eines Unternehmens bildet und daher die effektive Nutzung dieses Rohstoffs als Zielvorgabe definiert wird [SERV98, S. 325]. Dieser Herausforderung kann nur begegnet werden, wenn die gezielte Entwicklung, Überwachung und Verbesserung von Strategien, Prozessen, Organisationsstrukturen und Technologien zur Wissensverarbeitung sichergestellt werden [PROB97]. Wissen wird also immer mehr zum wettbewerbsentscheidenden Produktionsfaktor. Dabei gilt es zu beachten, dass das Managen dieser Ressource allein nicht ausreicht, vielmehr ist es notwendig, dass das Unternehmen das Lernen und Anwenden von Wissen effektiv und effizient organisiert und nutzt. Das Wissensmanagement kann somit als Erweiterung des Business Reengineering betrachtet werden, da diese beiden Ansätze auf die Steigerung des Unternehmenswertes abzielen. Das Reengineering strebt dabei eher die Überwindung des Taylorismus in der physischen Wertschöpfung an, wohingegen das Wissensmanagement auf das Managen der intellektuellen Wertschöpfung abzielt [TEUF99, S. 34].

Im Gesamtkontext des Wissensmanagements beschränken sich die Überlegungen im Rahmen dieser Arbeit auf die Betrachtung des Wissens, das im Rahmen einer Softwareadaption generiert wird. Bei der Adaption einer Softwarebibliothek wird

sowohl explizites als auch implizites Wissen verwendet bzw. offengelegt, daher sind beide Arten im Umfang der folgenden Betrachtungen enthalten.

Während der Anforderungsanalyse, der Erstellung von Soll-Konzepten, während des weiteren Verlaufs des gesamten Einführungsprojektes, aber auch über den gesamten Einsatzzeitraum der Software hinweg, wird sehr viel Wissen erzeugt, von implizitem in explizites Wissen transformiert und dokumentiert. Das vorrangige Ziel ist es nun, diese Informationen in einen sogenannten Wissenstransferzyklus einzubetten, um eine effiziente Nutzung dieser Ressource zu ermöglichen. Der Kreislauf beschreibt den Prozess der Erzeugung bzw. Transformation, der Speicherung, der Distribution und der Nutzung des Wissens im Lebenszyklus einer Softwarebibliothek. Die einzelnen Phasen werden im folgenden näher beleuchtet.

- *Erzeugung bzw. Transformation*

Bei der Anforderungsanalyse wird durch verschiedene Personen Wissen bezüglich des Istzustands, des gewünschten Sollkonzepts, der Prozessabläufe usw. zusammengetragen. In der Regel kommt dabei sowohl unternehmensinternes Wissen (z. B. Berichte, Arbeitsplatzbeschreibungen) als auch externes Wissen (z. B. Softwaredokumentation, Beratererfahrung) zum Einsatz. Zusätzlich wird durch Diskussionsrunden sehr viel implizites Wissen freigesetzt. Im weiteren Verlauf des Projektes und über den gesamten Lebenszyklus der Software hinweg wird noch weiteres Material erzeugt. In diesen Bereich fällt zum Beispiel auch die Dokumentation des Customizing, das wichtige Informationen über die Ausprägung bzw. die Einstellungen der Softwarelösung enthält.

- *Speicherung*

Der Punkt der Speicherung ist von essentieller Bedeutung. Vor allem das freigesetzte implizite Wissen muss durch eine sorgfältige Dokumentation in explizites Wissen transformiert und nutzbar gemacht werden. Generell fällt unter diesen Punkt die Zusammenfassung und zentrale Dokumentation der erzeugten Informationen in einer unternehmensweiten Wissensbasis.

- *Distribution*

Die Speicherung des erzeugten Wissens alleine reicht für ein effektives

Knowledge Management nicht aus. Das Wissen muss der gesamten Organisation zugänglich gemacht werden. Dabei spielen die Unternehmenspolitik und die zur Verfügung stehende Systemlandschaft eine entscheidende Rolle. In der heutigen Zeit wird zur Verteilung des Wissens meist eine Groupware oder das Internet bzw. Intranet genutzt. Zugriffsmechanismen regeln dabei im Rahmen eines Berechtigungskonzepts den kontrollierten Zugang zu Informationen.

- *Nutzung*

Ungenutztes Wissen ist nicht produktiv und hat keinen sozialen oder wirtschaftlichen Wert [COHE97, S. 14]. Das bedeutet, dass das erzeugte, gespeicherte und verteilte Wissen durch die Mitglieder der Organisation auch genutzt werden muss. Die Anwendung wird durch entsprechende Werkzeuge, die den einfachen und schnellen Zugriff auf die relevanten Informationen ermöglichen, gefördert. Ein solches Werkzeug soll im zweiten Teil dieser Arbeit entwickelt werden, um den gezielten Zugriff auf die Informationen über Systemeinstellungen sicherzustellen. Besonders in dem von WENZEL entwickelten Ansatz zur Analyse von produktiven Systemen im Sinne des Reverse Business Engineering spielt die Nutzung des Wissens bezüglich der Systemeinstellungen eine entscheidende Rolle [WENZ00].

2.3.2 Strategien und Werkzeuge

In diesem Abschnitt werden methodische und technologische Konzepte zur Realisierung eines integrierten Wissensmanagements erläutert, die dann in den folgenden Kapiteln als Anregung zur Entwicklung von THESEUS dienen.

2.3.2.1 WISSENSBASIS

Die Wissensbasis bildet den Kern des unternehmensweiten Wissensmanagements. In ihr ist im Idealfall das gesamte, relevante Unternehmenswissen abgebildet, daher ist neben den verschiedenen informationstechnologischen Speichermedien die logische Darstellung des Wissens für die effektive Nutzung desselben von essentieller Bedeutung. Auch für die systematische Ablage der erzeugten Informationen ist die Strukturierung der Inhalte sowie der Aufbau einer spezifischen Semantik erforder-

derlich [GENT99, S. 33]. Wie Unternehmen die Wissensbasis einsetzen, hängt von vielen verschiedenen Faktoren, z. B. der Produktstruktur, ab. Der Umfang der Anforderungen lässt sich anhand der folgenden Anwendungsgebiete verdeutlichen. Je nach individuellem Bedarf des Unternehmens können u. a.

- die Schaffung einer Webpräsenz und eines Intranets,
- die Erstellung von Informationsprodukten,
- die Unterstützung von Softwareeinführungen sowie stetige Verbesserungen des Systems und
- die Unterstützung von Anwenderschulungen

von Bedeutung sein [SAP00b, S. 10]. Der für diese Arbeit interessante Teil der Wissensbasis umfasst alle Informationen, die im Zusammenhang mit der Einführung bzw. dem Einsatz einer Softwarebibliothek generiert werden. Diese sind über Schnittstellen mit der unternehmensweiten Wissensbasis verbunden und somit in den Gesamtzusammenhang des Wissensmanagements integriert. Die Informationen, die in den Systemeinstellungen und Adaptionprozessen stecken, werden dadurch auch für andere Unternehmensbereiche nutz- und verwertbar.

Das einfache Auffinden von Wissen trägt somit ganz wesentlich zum Erfolg des Wissensmanagements bei. Ziel muss es daher sein, Informationen unternehmensweit transparent und für verschiedene Sichtweisen verständlich abzubilden. Die technologischen Speichermedien werden in Kapitel 2.3.2.2 beschrieben.

2.3.2.2 WISSENSINFRASTRUKTUR

Die Infrastruktur ist die Basis des Wissensmanagements auf der die Generierung und Nutzung aufsetzen (vgl. Abbildung 2-7). Sowohl organisatorische als auch informationstechnologische Entwicklungen der letzten Jahre haben die Wissensinfrastruktur entscheidend geprägt.

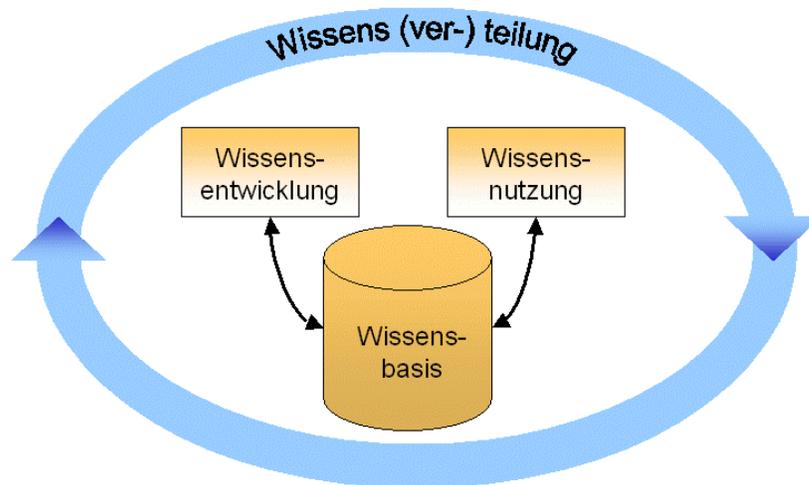


Abbildung 2-7: Wissensinfrastruktur [GENT99, S. 15]

Während zentralisierte, hierarchische Organisationen das relativ stabile Wissen zentral verwaltet haben, steigt die Notwendigkeit dezentrale Wissensbestände zu koordinieren, Redundanzen zu vermeiden und Synergien zu nutzen [GENT99, S. 17]. Eine moderne Infrastruktur besteht u. a. aus Internet und Intranet, solider Verteilungssoftware, wie Groupware-Systeme, sowie moderner Wissensspeicher, beispielsweise das Data Warehouse.

Im folgenden werden kurz die möglichen technologischen Plattformen für den Aufbau einer Wissensbasis skizziert, wobei nur auf die für die Themenstellungen wichtigen Aspekte und Werkzeuge ausführlicher eingegangen wird. Für weiterreichende Informationen zu den einzelnen Plattformen bzw. Speichermedien sei auf die einschlägige Literatur aus dem Bereich der Informationsverarbeitung verwiesen [z. B. HAN98].

Die technologische Plattform bestimmt die Entwicklungs- und Nutzungsmöglichkeiten des Wissens und ist zugleich abhängig von den zu speichernden Wissensarten. Generell lassen sich die Systeme in zwei Kategorien einteilen, zum einen Systeme, die für die Speicherung und Verwaltung von strukturierten bzw. formatierten Daten eingesetzt werden. Daneben müssen zunehmend semi- bzw. unstrukturierte Informationen, z. B. Bilder, Videosequenzen usw., abgebildet und bedient werden.

SYSTEME FÜR STRUKTURIERTE BZW. FORMATIERTE DATEN

Systeme, die strukturierte Daten verwalten, können unter dem Begriff Datenbanksysteme zusammengefasst werden. Im Allgemeinen unterscheidet man traditionelle

und objektorientierte Datenbanksysteme. Im Bereich der traditionellen Systeme spielen die modernen relationalen Systeme neben den hierarchischen Modellen die entscheidende Rolle. Die für die Verwaltung von großen Datenmengen konzipierten relationalen Systeme sind insbesondere für die Realisierung eines Data Warehouse zur Selektion und Speicherung entscheidungsrelevanter Informationen interessant. Für den Bereich der anwenderfreundlichen und transparenten Dokumentation der Systemeinstellungen im Lebenszyklus einer Softwarebibliothek sind die sogenannten objektorientierten Datenbanksysteme von zentraler Bedeutung, wie die folgenden Ausführungen zeigen.

Die reale Welt, die durch eine Datenbank abgebildet wird, kann als Verbindung von einzelnen miteinander in Verbindung stehenden Objekten gesehen werden. Während die traditionellen Datenbanksysteme diese in künstliche Strukturen zerlegen, ermöglichen objektorientierte Systeme eine vollständige und damit realitätsnahe Abbildung der Wirklichkeit. Durch diesen Ansatz ist der Anwender in der Lage, die Datenbank als eine Sammlung von komplexen, untereinander in Beziehung stehenden Objekten zu sehen, die je nach Zielsetzung und Fragestellung mit unterschiedlicher Detaillierung betrachtet werden können. Ein weiterer Vorteil ist, dass die benötigten Informationen mit den dazugehörigen Attributen logisch gekapselt sind und die Datenbank das Zusammenführen von Daten und Applikation ersetzt. Darüber hinaus besitzen objektorientierte Datenbanksysteme noch weitere Vorteile, insbesondere in Bezug auf die Modularität und Wiederverwendbarkeit. Die integrierte Sichtweise von Daten und Funktion weist eine geringere kognitive Distanz zum menschlichen Denken auf, deshalb sind objektorientierte Datenbanken besser geeignet die Realität ohne Strukturbruch für den Anwender verständlich abzubilden. Da in objektorientierten Systemen jede Form von Objekten - Zeichen, Bilder, Videosequenzen etc. - einfach durch den Anwender abgelegt werden können, sind diese Datenbanksysteme als technologische Plattform zum Aufbau einer Wissensbasis besonders geeignet. Gerade im Bereich des Wissensmanagements gibt es eine Vielzahl an Informationen, die nur sehr schwer zu formalisieren oder nach bestimmten Regeln zu automatisieren sind. Objektorientierte Datenbanksysteme können die daraus resultierenden Konstrukte mit ihren Abhängigkeiten besser abbilden, als die traditionellen Datenbanktechnologien [GENT99, S. 20-32].

SYSTEME FÜR UNSTRUKTURIERTE BZW. UNFORMATIERTE DATEN

Die Systeme für semi- bzw. unstrukturierte Daten haben weniger restriktive Anforderungen an die Struktur der abzubildenden Informationen. Zur Speicherung solcher unformatierter Wissensbestände hat sich der Aufbau einer Intranetlösung als verteiltes, dokumentenorientiertes Datenverwaltungssystem etabliert.

Der Einsatz der Internet-Technologie ist nicht nur für die Wissensgewinnung und -nutzung außerhalb des Unternehmens hilfreich, vielmehr kann diese Technologie durch den Aufbau von flexiblen Informations- und Kommunikationskanälen dazu dienen, die Wissensdarstellung und -verteilung innerhalb des Unternehmens in Form eines sogenannten Intranets sicherzustellen. Ein Intranet nutzt die Protokolle und Dienste des Internet bzw. des World Wide Web (WWW) für das unternehmensinterne lokale Netzwerk. Zur Abbildung der unformatierten Daten ist die Beschreibungssprache HTML (Hypertext Markup Language) von besonderer Bedeutung. Über diese einfache Beschreibungssprache lassen sich Web-Seiten erstellen, die Texte, Bilder oder Videos beinhalten können. Damit lässt sich mit geringem Aufwand der Teil der Wissensbasis aufbauen, der die unstrukturierten Informationen enthält. Dabei werden die Inhalte der einzelnen Dokumente von den Zusatzinformationen getrennt. Die Inhalte werden als einzelne Dokumente gespeichert, die Zusatzinformationen werden als Attribute in einer Datenbank abgelegt. Diese Vorgehensweise hat folgende Vorteile [SIED99, S. 183]:

- Durch die Verwaltung der Links in Datenbanken können Inkonsistenzen, wie sie durch einfache Webserver möglich sind (z. B. durch Manipulation oder Löschen von Dokumenten), leichter vermieden werden,
- über Berechtigungen kann die Zugriffsmöglichkeit der Benutzer auf Dokumente sowie die Art der Bearbeitung geregelt werden,
- die Datenbank kann anhand der Attribute effektiv durchsucht werden (z. B. Erstellungsdatum, Autor und Verfallsdatum),
- Attribute können persönlich sowie projektspezifisch hinzugefügt werden und
- die Dokumente können leichter administriert werden.

Aus technischer Sicht wird die Wissensbasis physisch auf einem Web-Server abgebildet, der über einen Standardbrowser jedem Mitglied des Unternehmens zugänglich gemacht werden kann.

Für die Speicherung der unstrukturierten Daten durch diese Technologie sprechen der hohe Verbreitungsgrad der internetbasierten Technologien in den Unternehmen sowie die hohe Akzeptanz bei den Benutzern, die aus der intuitiven Verständlichkeit resultiert. Als Nachteil muss das Fehlen von zentralen Datenbank-Management-Systemen erwähnt werden, das die Vermeidung von Datenredundanz und -inkonsistenzen verhindert. Dennoch fügt sich das Intranet durch die hohe Interaktivität und Hardwareunabhängigkeit in die modernen Konzepte der Informations- und Kommunikationstechnologien ein und bietet damit eine geeignete Plattform für zahlreiche Methoden des Wissensmanagements [GENT99, S. 25-26].

2.3.2.3 HYPERKNOWLEDGE

Das Konzept des Hyperknowledge ist angelehnt an das Hypertextprinzip. Hypertext ermöglicht, zwischen Wissensobjekten, die in der Logik des entsprechenden Anwendungsfeldes flexibel verknüpft sind, zu navigieren, diese zu aktivieren und zu manipulieren [BEIN00, S. 310]. Wenn Informationen in einen richtigen kausalen Zusammenhang gestellt werden, wird daraus aus Benutzersicht Wissen. Die Reihenfolge der Informationsobjekte ist dabei nicht starr, sondern flexibel kombinierbar, so dass für unterschiedliche Anwendungszusammenhänge durch Kombination der einzelnen Objekte flexibel Wissen in verschiedener Form generiert werden kann [BEIN00, S. 321]. Das erklärte Ziel des Hyperknowledge ist es, einzelne Informationen zu Wissensobjekten zusammenzufassen und diese flexibel je nach Anwendungsgebiet miteinander zu verknüpfen. Die Multidimensionalität der abgebildeten semantischen Zusammenhänge bildet dabei den Schwerpunkt. Die Wissensobjekte stehen in keinem hierarchischen, sondern einem netzwerkartigen Zusammenhang. Jedes Objekt bildet einen Knoten des Wissensnetzes, wobei potentiell alle Knoten miteinander verknüpft werden können.

2.3.3 Erfahrungsdatenbanken

Eine produktiv genutzte Softwarebibliothek kann bei mittelfristiger Betrachtung nicht als fixierte Einheit betrachtet werden. Lösungen, welche die von HUFGARD

definierten Eigenschaften besitzen, eröffnen den Unternehmen eine Vielzahl an Möglichkeiten, um den Nutzen aus dem Einsatz der Software stetig zu vergrößern [HUFG94, S. 76-78]. Die Softwaresysteme müssen an sich ändernde Anforderungen angepasst werden. Neben neuen betriebswirtschaftlichen Verfahren in den Bereichen Fertigung, Logistik und Rechnungswesen, sind auch technologische Neuerungen und neue zusätzliche Funktionen der Software die Initiatoren für Systemänderungen. Schlecht oder gar nicht dokumentierte Systemeinstellungen werden dabei schnell zu einem großen Handicap, besonders dann, wenn ein wichtiger Wissensträger wechselt oder ausscheidet und so das verfügbare Wissen verloren geht. Die vorangegangenen Ausführungen machen deutlich, dass das gesammelte Wissen bezüglich des Softwaresystems in einer Erfahrungsdatenbank abgelegt werden muss, um unnötigen Aufwand zu vermeiden und damit die Einführungs- und Einarbeitungszeiten zu verkürzen.

Die Erfahrungsdatenbank zur Dokumentation einzelner Kundenprojekte bzw. Kundensysteme beinhaltet im wesentlichen zwei Gruppen von Dokumenten, zum einen die Projektdokumentation zur Beschreibung von

- Projektverlauf,
- Projektorganisation,
- Projektstandard,
- Projektplan usw.

Daneben ist die Systemdokumentation zu nennen, zu der

- das Fachkonzept als technisches Systemkonzept,
- die Beschreibung von Geschäftsabläufen und -prozessen,
- die Endbenutzerdokumentation,
- die Schnittstellenbeschreibungen,
- der Aufbau und Inhalt des Berichtswesens sowie
- die Beschreibung der Systemeinstellungen (Customizing)

zählen. Die Erfahrungsdatenbanken werden aus technischer Sicht über verschiedene Medien zur langfristigen Speicherung der Dokumentation realisiert, diese hängen im Einzelfall von den individuellen Belangen des Unternehmens ab.

2.4 Mind Mapping mit Knowledge Maps

Die Ausführungen in Kapitel 2.3.2.2 zeigen, dass die Vielzahl an Informationen geordnet und strukturiert werden muss, um eine effektive Nutzung zu ermöglichen. Die technologische Umsetzung dieser Aufgabe reicht alleine nicht aus, um Wissen transparent für den Anwender zu verwalten. Auch aus methodischer und konzeptioneller Sicht müssen die Inhalte der Wissensbasis aufbereitet werden. Im folgenden werden zwei Ansätze vorgestellt, die eine verständliche Darstellung komplexer Informationszusammenhänge im Rahmen der Softwareadaption ermöglichen. Die Mind Mapping-Methode von BUZAN [BUZA97] als allgemeines Konzept zur Reduzierung der Komplexität bildet dabei die Grundlage für den zweiten Ansatz, die sogenannten Knowledge Maps, die eine Anwendung der Mind Mapping-Methode auf betriebswirtschaftliches Wissen bezüglich eines bestimmten Sachverhalts vorstellen. Diese beiden Ansätze dienen für den im Rahmen dieser Arbeit entwickelten Ansatz als Basis.

2.4.1 Mind Mapping-Methode

Die Mind Mapping-Methode basiert auf einem Konzept des britischen Wissenschaftlers Tony Buzan [BUZA97]. Dieses Konzept berücksichtigt die Funktionsweise des Gedächtnisses und die Arbeitsweise des menschlichen Gehirns. BUZAN geht von der Erkenntnis aus, dass das Denken ein sehr komplexer Vorgang ist, in dem ständig neue Assoziationen und Strukturen gebildet werden. Die von BUZAN entwickelte Mind Map-Technik spricht daher gezielt beide Gehirnhälften des Menschen an und nutzt die Synergieeffekte aus beiden Gehirnregionen. Informationen werden nicht mehr linear in Form von Listen oder Fließtext dargestellt, vielmehr wird die Darstellung auf das Notwendige beschränkt und bei Bedarf durch den Einsatz von Schrift, Bildern, Zahlen, Zeichenelementen, Codes usw. ergänzt. Zentraler Gedanke des Mind Mapping ist die Gruppierung aller relevanten Aspekte um eine Grundidee. Dieses Hauptthema wird zentral angeordnet und kann somit schnell erfasst werden. Dabei werden alle Informationen zu einem Thema visualisiert, unabhängig davon, ob es sich um Grafik, Text usw. handelt. Übertragen auf die Adaption einer Softwarebibliothek kann dies das Zusammenführen aller Informationen zu einem Geschäftsprozess sein, d. h. die Verknüpfung von Anforderungsanalyse, Systemeinstellungen, Systemtransaktionen usw. Über das Mind Map-

ping ist es möglich, Ideen und Konzepte in einer wachsenden grafischen Struktur, bei der maximalen Ausnutzung des geistigen Potenzials, darzustellen.

2.4.2 Knowledge Maps

Die Knowledge Maps sind die Anwendung der Mind Mapping-Methode aus betriebswirtschaftlicher Sicht für einen bestimmten Sachverhalt. Alle relevanten Informationen für diesen Sachverhalt werden in Form einer Mind Map visualisiert und strukturiert. TEUFEL nutzt dieses Konzept zur Unterstützung der Prozessanalyse beim Einsatz der Softwarebibliothek der SAP AG und adaptiert die Knowledge Maps daher auf die Funktionalitäten der SAP-Software [TEUF99]. Knowledge Maps können sich aber ohne diese Beschränkung auf allgemeine betriebswirtschaftliche Sachverhalte beziehen. Die Wissensinhalte einer Knowledge Map unterliegen einer ständigen Veränderung, bezogen auf die eingesetzte Software. Sie stellen eine dynamische Dokumentation dar und werden während des Einsatzes in Adaptionenprojekten um Kundenanforderungen ergänzt. Die Verbindung der einzelnen Knowledge Maps erfolgt über die aus der Internet-Technologie bekannten Hyperlinks, so dass ein großes Potenzial an Wissensvernetzung entsteht.

Grundsätzlich können Wissensinhalte auch mit Hilfe von Tabellen dargestellt werden (vgl. Tabelle 2-1).

Tabelle 2-1: Aufbau des Unternehmensbereichs Produktion

Produktion	
Serienfertigung	
Prozessfertigung	
Diskrete Auftragsfertigung	
Kanban	
Zusätze	<ul style="list-style-type: none"> • CAD Integration • Variantenkonfiguration • Klassifizierung • Dokumentenverwaltung

Die beschränkten grafischen Möglichkeiten der tabellarischen Darstellung verdeutlichen die Stärke der Knowledge Maps. Die Erfassung komplexer Sachverhalte und

Zusammenhänge wird durch die Zentralisierung des Hauptthemas in Verbindung mit der grafischen Aufbereitung wesentlich erleichtert (vgl. Abbildung 2-8).

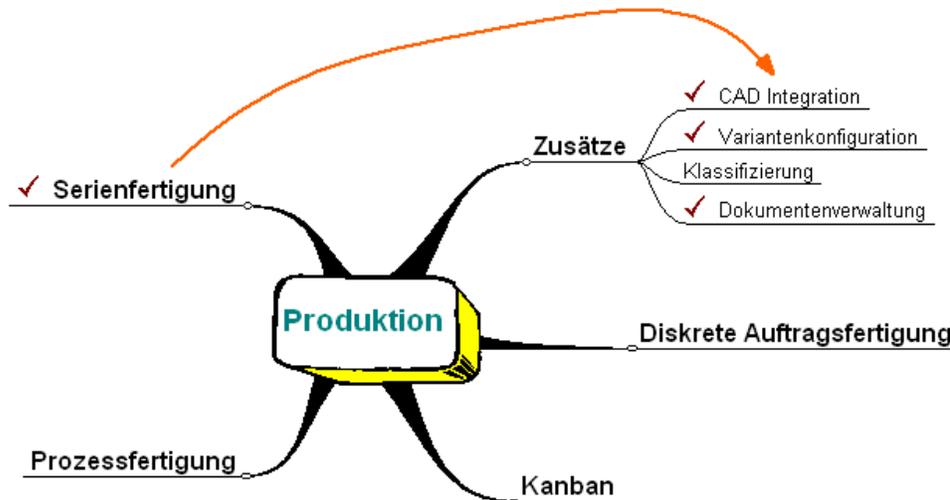


Abbildung 2-8: Knowledge Map für den Unternehmensbereich Produktion

Abbildung 2-8 zeigt exemplarisch den Aufbau einer Knowledge Map für die Analyse des Unternehmensbereichs Produktion.

Der Aufbau einer Knowledge Map ist durch fünf grundsätzliche Gestaltungsmerkmale bestimmt, die hauptsächlich in den Grundregeln der allgemeinen Mind Mapping-Methode von BUZAN begründet sind [TEUF99, S. 85]:

- Wurzeln stellen das zentrale Thema einer Knowledge Map dar,
- Hauptäste bilden die grundsätzlichen Kapitel bzw. Ordnungsbegriffe des Themas ab,
- Nebenäste untergliedern die grundsätzlichen Ordnungsbegriffe weiter,
- Symbole heben einzelne Äste hervor und

Verbindungslinien zwischen den Ästen veranschaulichen wichtige Interdependenzen.

Das Zusammenspiel dieser Merkmale erlaubt es, umfangreiche und komplexe Themen, die z. B. bei der Adaptionanalyse einer Softwarebibliothek zu bearbeiten sind, schnell zu verstehen und zu vermitteln. Knowledge Maps können in ver-

schiedene Typen untergliedert werden, um so die unterschiedlichen Problemstellungen abzubilden. Im Adaptionsumfeld werden bspw. folgende Arten von Knowledge Maps verwendet [TEUF99, S. 91-97]:

- Überblicks-Knowledge Maps zur Einführung in einen Unternehmensbereich (vgl. Abbildung 2-8),
- Szenario-Knowledge Maps zur Darstellung von zusammengehörenden Geschäftsprozessen,
- Organisations-Knowledge Maps zur Ausprägung der individuellen Organisationsstruktur,
- Business-Object-Knowledge Maps zur Darstellung von Belegen und Stammdaten sowie
- Prozess-Knowledge Maps zur detaillierten Prozessanalyse.

Knowledge Maps unterstützen die einzelnen Phasen eines Adaptionprojekts, wie sie beispielsweise bei STRELLER beschrieben werden [STRE99, S. 189-197]. Im folgenden werden die einzelnen Einsatzmöglichkeiten der Knowledge Maps kurz beschrieben, da diese für die weitere Verwendung des Ansatzes von zentraler Bedeutung sind. Eine ausführliche Beschreibung der Einsatzbereiche ist bei TEUFEL [TEUF99, S. 97-100] zu finden.

Während der Presalesphase bzw. der Einsatzuntersuchung werden Knowledge Maps dazu genutzt, um eine schnelle Identifikation der relevanten Softwarebausteine sicherzustellen. Besonders die Visualisierung des Sachverhalts schafft im Vergleich zu den bisherigen Vorgehensweisen (z. B. mit Checklisten) eine höhere Transparenz und damit Akzeptanz beim Anwender.

In der Blue Print Phase können die Knowledge Maps helfen, die Einarbeitungszeit in ein spezielles Thema durch die einfache und transparente Darstellung des Sachverhalts zu verkürzen und so den Aufwand für einzelne Workshops zu verringern. Zudem besteht die Möglichkeit, die bestehenden Knowledge Maps zu erweitern und um individuelle Informationen zu ergänzen. Die strukturierte Vorgehensweise unterstützt zudem die Umwandlung von implizitem in explizites Wissen, wie sie ein durchgängiges Wissensmanagement verlangt.

Nach der Anforderungsanalyse unterstützen die Knowledge Maps die Umsetzung der Analyseergebnisse im Rahmen der Realisierungsphase durch zahlreiche Zusatzinformationen, die zu einer Knowledge Map hinterlegt sind. Bei diesen Informationen handelt es sich beispielsweise um Standarddokumentationen, Referenzmodellausschnitte, Transaktionsbeschreibungen und Erfahrungswissen, das zusätzlich dokumentiert wurde. Der dokumentarische Charakter der Knowledge Maps sorgt dafür, dass die wichtigen Zusatzinformationen aus der Blue Print Phase während der Systemkonfiguration genutzt und ergänzt werden können.

Die stetige Verbesserung des Systems im Rahmen des Continuous Change setzt voraus, dass auf das Wissen, das im Rahmen der Einführung der Software gesammelt wurde, zurückgegriffen werden kann, um inkonsistente Konstellationen der Software frühzeitig auszuschließen. Die Knowledge Maps dokumentieren dieses Wissen in allen Phasen des Lebenszyklus der Softwarebibliothek und ermöglichen so den Zugriff auf alle relevanten Informationen. Notwendige Erweiterungen werden durch Ergänzungen in den Knowledge Maps dokumentiert und nachvollziehbar gemacht. Bei diesen Erweiterungen kann es sich um neue Prozessbausteine handeln, die bisher noch nicht im Einsatz sind, oder um bereits produktive Bausteine, von denen bislang nur Teilfunktionalitäten genutzt werden. Im ersten Fall wird eine neue Map zu diesem Prozessbaustein bearbeitet und in das Knowledge Warehouse eingefügt, im zweiten Fall wird die bestehende Knowledge Map um die neuen Ausprägungen bzw. Funktionen erweitert. Ein Beispiel soll die Verwendung der Knowledge Maps im Continuous Change verdeutlichen: Bei der Ersteinführung wurde die Kundenanfragebearbeitung nicht implementiert. Die Prozesse im Bereich des Vertriebs sollen nun um genau diesen Prozess erweitert werden. Wie bei der Ersteinführung wird die Knowledge Map zu diesem Prozessbaustein bearbeitet und mit den bereits bestehenden Maps verknüpft. In einem anderen Fall wurde bei der Ersteinführung auf die Kreditlimitprüfung im Rahmen der Kundenauftragsbearbeitung verzichtet. Neue Anforderungen machen nun die Einführung dieser Teilfunktionalität notwendig. Die bereits bearbeitete Knowledge Map des Prozessbausteins der Kundenauftragsbearbeitung wird erneut aufgegriffen und der gewünschte Hauptast mit den Informationen zur Kreditlimitprüfung wird ergänzt, wobei jederzeit auf die relevanten Informationen aus dem gesamten Prozess, die im Rahmen der Ersteinführung hinterlegt wurden, zurückgegriffen kann. Der In-

formationsgehalt einer Knowledge Map erhöht sich also bei einem Einsatz im Rahmen des Continuous Change ständig.

Knowledge Maps sind ein Hilfsmittel, um ein organisatorisches Wissensmanagement für die Bereich der Organisation und Geschäftsprozesse zu realisieren. Der Einsatz der Knowledge Maps ist ein wissensbasiertes Vorgehen bei der Adaption einer Softwarebibliothek und kann als erster Schritt zu einem umfassenden Wissensmanagement aufgefasst werden [TEUF99, S. 100]. Alle Materialien, Arbeitsergebnisse usw., die während der Adaption anfallen, werden der entsprechenden Knowledge Map zugeordnet. Die Sammlung der Knowledge Maps bildet dann ein Knowledge Warehouse für den Bereich der Softwareadaption, das für alle Anwender und sonstigen Mitarbeiter des Unternehmens Informationen bzgl. der Prozessabläufe, Organisationsstrukturen, Systemeinstellungen etc. sammelt, verwaltet und bei Bedarf transparent zur Verfügung stellt.

2.5 Erkenntnisfortschritt

Der Aufstieg des Wissensmanagements, dem die Erkenntnis um die zentrale Bedeutung des Faktors Wissen zugrunde liegt, geht einher mit dem Wandel der Informationsverarbeitung von individuell entwickelten Softwareprodukten hin zu betriebswirtschaftlicher Anwendungssoftware und dem vermehrten Einsatz von Softwarebibliotheken. THOME und HUGFARD haben diese Entwicklung ausführlich verfolgt und kommentiert [THHU96]. Die moderne Informationsverarbeitung spielt bei der Realisierung des unternehmensweiten Wissensmanagements die zentrale Rolle, wie die vorangegangenen Kapitel zeigen. Die dort betrachteten Grundlagen und Methoden können in das Umfeld der Softwareadaption einbezogen werden, um das gesammelte Wissen effizient zu nutzen. Im Rahmen der in dieser Arbeit behandelten Problemstellung geht es nicht darum, eine Konzeption zur Implementierung des Knowledge Managements zu entwickeln, vielmehr sollen die Erfahrungen und Kenntnisse aus diesem Bereich genutzt werden, um dem Gedanken des Continuous System Engineering Rechnung zu tragen.

Zunächst soll noch einmal herausgestellt werden, welche Gemeinsamkeiten das Wissensmanagement und die Adaption von betriebswirtschaftlichen Softwarebibliotheken haben, um daraus grundsätzliche Anforderungen und Vorgehensweisen für die Gestaltung eines ganzheitlichen Ansatzes zur Verbesserung der Adaptions-

qualität abzuleiten. Wie bereits ausführlich erläutert, sind die Hauptziele des Wissensmanagements die Sammlung, Speicherung, Verteilung und Nutzung des gesamten Wissens, das in einem Unternehmen vorliegt. Eine ähnliche Zielsetzung verfolgt auch die Philosophie des Continuous System Engineering. Im ersten Zyklus werden die Anforderungen des Unternehmens an die Software ermittelt und gesammelt. Basierend auf diesen Erkenntnissen werden die notwendigen Teile der Softwarebibliothek identifiziert und weitere Informationen bezüglich der System-einstellungen zusammengetragen. Die implementierte Software beinhaltet dabei sehr viel Wissen über Organisationsstrukturen, Prozessabläufe usw., das als Vorbedingung für die weiteren Änderungen bzw. Verbesserungen im Sinne des CSE dient [THHU96, S. 88]. An dieser Stelle wird die Affinität zum Wissensmanagement deutlich, da für alle Folgeprojekte bzw. Folgeaktivitäten nach der Eröffnungslösung auf genau die Informationen zurückgegriffen werden muss, die in der produktiven Software ihre Anwendung finden. Es handelt sich auch bei den Prozessen der Softwareadaption um einen Wissenszyklus, in dem Wissen erzeugt, gespeichert, verteilt und genutzt wird. Die Strategien und Methoden des Wissensmanagement können daher helfen, die Informationen über die Systemkonfiguration effizienter zu nutzen, als dies mit den bisherigen Werkzeugen möglich ist. Abbildung 2-9 verdeutlicht noch einmal, dass im Bereich der Softwareadaption eine Wissensbasis aufgebaut wird, die alle Angaben bezüglich der Softwareeinführung bzw. des aktuellen Systems sammelt und für weiterführende Projektierungen im Sinne des Continuous Improvement zur Verfügung stellt. Diese Wissensbasis sollte dabei integraler Bestandteil der unternehmensweiten Wissensbasis sein, um mögliche Redundanzen zu vermeiden und Synergien zu nutzen.

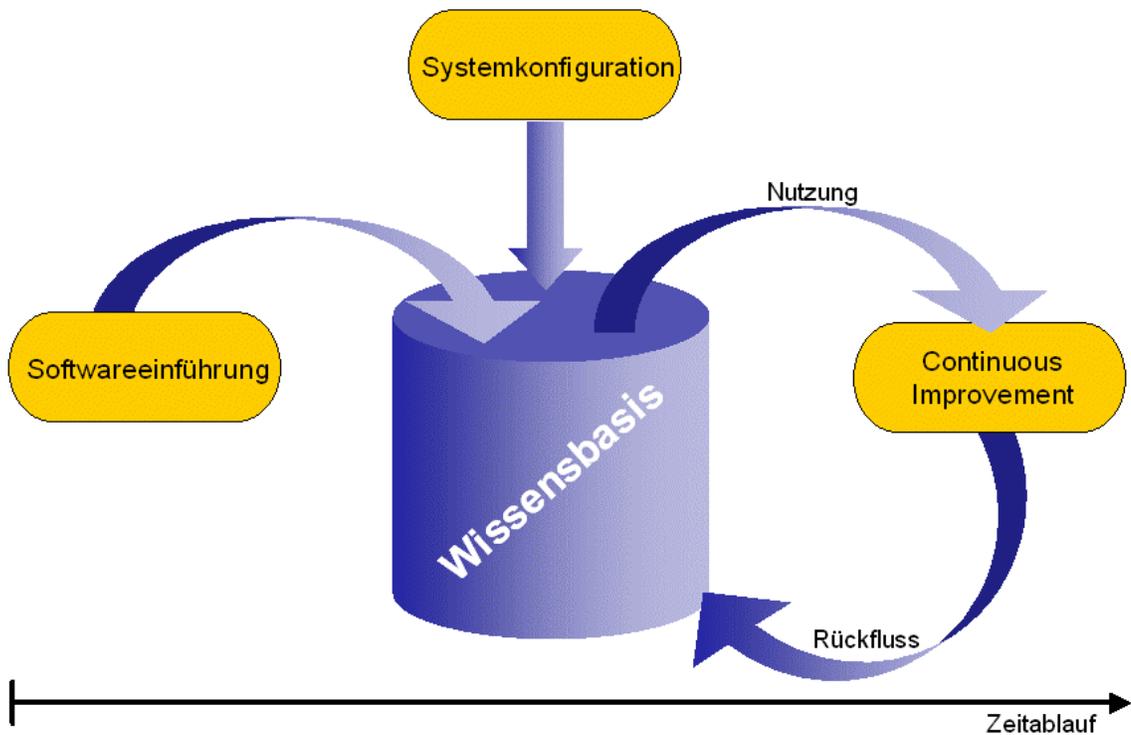


Abbildung 2-9: Wissensbasierte Adaption

Die zentrale Frage ist, welche Erkenntnisse sich aus diesen Gemeinsamkeiten für die weiteren Untersuchungen ableiten lassen. Folgendes gilt es festzustellen:

1. Für den Aufbau einer Wissensbasis bilden Datenbanken die Grundlage. Durch die strukturierte Ablage des gesammelten Wissens kann der Zugriff auf alle relevanten Informationen flexibel gestaltet werden. Moderne Datenbankmodelle erlauben dabei die Verwaltung von heterogenen Informationen (z. B. Videomaterial, Grafiken).
2. Die Wissensbasis für die Softwareadaption ist ein integraler Bestandteil der unternehmensweiten Wissensbasis. Alle gespeicherten Informationen können jederzeit verfügbar gemacht werden.
3. Die Integration der Wissensbasis setzt voraus, dass die Technologie der Datenbank auf Standards basiert. Gerade der Zugriff auf die Information durch verschiedene Werkzeuge bedingt eine offene Architektur.
4. Der Zugriff auf die Wissensbasis muss sowohl intern innerhalb der firmeneigenen Datenstruktur als auch extern möglich sein. Für die Umsetzung ist daher die Internettechnologie geeignet.

5. Für die Softwareadaption bietet sich eine objektorientierte Speicherung der Informationen an, da bei dieser Vorgehensweise Redundanzfreiheit und Wiederverwendbarkeit von Wissen sichergestellt werden kann.
6. Die Flexibilität in der Bereitstellung der erforderlichen Informationen verlangt eine stabile inhaltliche und organisatorische Abgrenzung der Informationseinheiten.
7. Zur Erhöhung der Transparenz im Rahmen der Softwareeinführung bzw. des Continuous Improvement hilft die Mind Mapping-Methode die Visualisierung der Sachverhalte zu unterstützen und das Verständnis für die Problemstellung zu fördern.
8. Die Konzentration auf ein zentrales Thema, wie sie das Mind Mapping propagiert, dient dem Aufbau eines Knowledge Warehouse. Dieses Konzept kann auch für das Continuous System Engineering genutzt werden.
9. Der Informationsgehalt des Wissensträgers steigt im Zeitablauf durch die Sammlung zusätzlicher Informationen stetig an. Für die durchgängige Dokumentation der Softwarebibliothek über den gesamten Lebenszyklus gilt dies entsprechend.

Diese Erkenntnisse gilt es im folgenden umzusetzen.

3 Ausgewählte Vorgehensmodelle zur Adaption von Softwarebibliotheken

Die Komplexität der Adaption von Softwarebibliotheken erfordert ein methodisches Vorgehen, um die effiziente Einführung bzw. Reorganisation eines Systems sicherzustellen. Dabei ist die Abstimmung der informellen und organisatorischen Einbindung innerhalb des Unternehmens von entscheidender Bedeutung. Dies wird durch die sogenannten Vorgehensmodelle aufgezeigt [SCHE97, S. 6-7]. Ein Modell wird im allgemeinen als komplexitätsreduzierende Abbildung der Realität oder eines Ausschnittes daraus verstanden [SCHM97, S. 161].

Im folgenden sollen ausgewählte Vorgehensmodelle zur Einführung betrieblicher Anwendungssysteme vorgestellt und untersucht werden. Prinzipiell decken die vorhandenen Vorgehensmodelle alle Phasen der Implementierung von Softwarebibliotheken ab und unterstützen darüber hinaus die stetige Weiterentwicklung des Systems, wie sie durch den von THOME und HUGFARD entwickelten Ansatz des Continuous System Engineering [THHU96] gefordert wird. Die vorgestellten Vorgehensmodelle werden insbesondere dahingehend untersucht, inwieweit sie geeignet sind, eine anwenderfreundliche und effektive Navigation durch das komplexe Wissen, das im Rahmen des Softwareeinsatzes gespeichert wird, sicherzustellen.

Darauf aufbauend werden Prinzipien entwickelt, die den Anwender bei der Lösung eben dieser Aufgabe unterstützen. Die Ausführungen beschränken sich dabei nicht auf ein spezielles Vorgehensmodell, sondern haben allgemeine Gültigkeit. Tabelle 2-1 zeigt eine nach technischen und inhaltlichen bzw. methodischen Gesichtspunkten sortierte Auflistung von Kriterien, die eine Beurteilung verschiedener Vorgehensweisen für die Adaption von Softwarebibliotheken im Hinblick auf die Eignung zur Unterstützung des CSE zulassen. Die Kriterien leiten sich vorwiegend aus heuristischen Überlegungen ab, da es bislang für derartige Beurteilungen keine generellen Lösungsvorschläge bzw. Merkmale gibt.

Tabelle 3-1: Kriterien für die Beurteilung verschiedener Vorgehensweisen zur Adaption von Softwarebibliotheken

Kriterium	Bedeutung
Technik	
Offenheit	Anbindung unterschiedlicher Adaptionswerkzeuge
Flexibilität	Erweiterungsmöglichkeiten
Informationsverwaltung	Technische Verwaltung der Informationssammlung
Softwareabhängigkeit	Adaptierbarkeit der Methode auf andere Softwareprodukte
Systemintegration	Zugriff auf das System
Versionsabhängigkeit	Technische Unterstützung eines Releasewechsels
Inhalt / Methodik	
Übersichtlichkeit	Transparente Aufbereitung der Informationen
Nachvollziehbarkeit	Erklärungskomponente
Stabilität	Änderung der Navigations- bzw. Informationsstrukturen
Vergleichbarkeit	Analyse von Änderungen
Vollständigkeit	Inhaltliche und methodische Abdeckung des Adaptionprozesses
Integration	Integration der eingesetzten Werkzeuge
Versionsabhängigkeit	Inhaltliche Update-Möglichkeiten

Die Auflistung erhebt keinen Anspruch auf Vollständigkeit und ist sicherlich durch das eine oder andere Kriterium zu ergänzen. Der Autor hat sich auf diese Kriterien beschränkt, da sie eine Bewertung der Vorgehensmodelle im Hinblick auf die Problemstellung dieser Arbeit ermöglichen. STRELLER hat für die Bewertung der Modelle im Hinblick auf die Projektabwicklung beispielsweise andere Kriterien herangezogen [STRE99, S. 57]

Aus den ermittelten Defiziten und Anforderungen werden Prinzipien für die Gestaltung einer Adaption-Workbench abgeleitet.

3.1 Vorgehensmodelle

Inhalt eines Vorgehensmodells sind Empfehlungen für den methodischen Ablauf der Entwicklung bzw. Implementierung eines Anwendungssystems. Ziel ist eine möglichst wirtschaftliche Realisierung, wobei die Vorgehensmodelle zur Planung, Steuerung und Kontrolle dienen. Der Arbeitsprozess muss in einzelne planbare

Einheiten zerlegt werden, um eine übersichtliche Darstellungsweise und ein zielgerichtetes Vorgehen zu ermöglichen. Den Abschluss der einzelnen Phasen bilden fest definierte Meilensteine, die eine ständige Fortschrittskontrolle gewährleisten [SEIB97, S. 431-433].

„Unter Methode wird der versachlichte, gedanklich objektivierbare, auf andere Personen übertragbare und von ihnen reproduzierbare Gang der Gewinnung wissenschaftlicher Erkenntnis verstanden“ [ALBA93, S. 38-43]. THOME beschreibt den Methodenbegriff mehr aus einem praxisorientierten Blickwinkel. Er versteht unter einer Methode ein geplantes, d. h. ein erprobtes und für ein Problem passendes, Vorgehen. Sie beschreibt überwiegend, mit welchen Aufgaben begonnen werden soll, wie diese zu bearbeiten sind und wie das Ergebnis zu verstehen ist [THOM90, K 1-1].

Nach SCHEER beschleunigt eine durchgängige Werkzeugintegration den Ablauf des Vorgehens [SCHE97, S. 7]. Je intensiver die Benutzung von integrierten Werkzeugen und Methoden ist, desto konsequenter kann ein Vorgehensmodell eingesetzt werden. Maschinell unterstützte Verfahren erleichtern dabei besonders den Entwicklern die Lösung komplexer Problemstellungen. Für die Unterstützung der Einführung bzw. Reorganisation einer Standardanwendungssoftware stehen eine Vielzahl unterschiedlicher Werkzeuge zur Verfügung, einige dieser Tools werden in den folgenden Kapiteln näher vorgestellt. Die von SCHEER geforderte Werkzeugintegration bzw. -verwendung ist aufgrund der sehr heterogenen Werkzeuglandschaft aber noch nicht in ausreichendem Maße gegeben.

In diesem Kapitel liegen die Schwerpunkte auf der Beschreibung des Vorgehensmodells AcceleratedSAP (ASAP) der SAP AG, das die größte Verbreitung im Rahmen der Adaption der marktführenden Softwarebibliothek R/3 aufweist und von nahezu allen Beratungshäusern eingesetzt wird. Des Weiteren befasst sich dieses Kapitel mit dem Vorgehensmodell Chestra bzw. LIVE METHOD/Chestra der Firma Siemens Nixdorf, da dieses Modell konsequent die im Rahmen des ITHAKA-Konzepts entwickelten Werkzeuge einsetzt und daher Gegenstand der Betrachtungen sein muss. Die ausführlichen Darstellungen sollen neben der Vorstellung der spezifischen Eigenschaften und Inhalte der Vorgehensmodelle einen Überblick über die grundsätzliche Projektabwicklung im Rahmen der Softwareadaption geben. STRELLER hat sich bei der Entwicklung des PANDORA-Konzepts ausführlich mit dieser Thematik befasst [STRE99].

3.1.1 Accelerated SAP (ASAP)

Die SAP AG hat mit der Einführungslösung AcceleratedSAP (ASAP) auf die Forderungen der Unternehmen reagiert, die Vorteile und Potenziale eines produktiven SAP-Systems so schnell wie möglich zu nutzen. ASAP wurde mit der Zielsetzung entwickelt, den Unternehmen zum einen eine rasche und effiziente Einführung der Software zu ermöglichen und zum anderen den Grundstein für eine kontinuierliche Geschäftsprozessoptimierung zu legen [TEUF99, S. 70].

Hinter ASAP und den integrierten Komponenten steht ein internationales Team aus Beratern und Projektleitern, die über Jahre hinweg Erfahrungen aus Implementierungsprojekten gesammelt, ausgewertet und schließlich in einem pragmatischen Ansatz umgesetzt haben. Redundante Projektaktivitäten wurden eliminiert und bestimmte, vormals sequentiell durchgeführte Schritte parallelisiert. Die gesamte Methode wird als Roadmap bezeichnet, die durch diverse unterstützende Beschleuniger (Accelerators) und Werkzeuge (Tools) ergänzt wird [APPE00, S. 71-73].

ASAP wurde nach dem Bauskastenprinzip entwickelt und besteht aus einzelnen Bestandteilen, die im Rahmen der Projektabwicklung selektiert werden können. Dadurch müssen nicht alle Komponenten von ASAP genutzt werden. Die Aufgaben, die vornehmlich während eines Projekts durchgeführt werden müssen, sind

- das Projektmanagement (Zeit-, Ressourcen- und Budgetplanung sowie Planung und Koordination aller Aktivitäten, z. B. Schulungen, Tests),
- die Adaption bzw. Konfiguration der Organisations- und Geschäftsprozessstruktur (Business Blueprint, Einstellen der Systemparameter im Rahmen des Customizing, Berechtigungskonzept etc.) und
- die Klärung bzw. Einrichtung des technischen Umfelds (Installation, Netzwerke, Drucker, Schnittstellen, Migration der Altdaten, Erweiterungen etc.).

Für die Lösung dieser Aufgabenbereiche sind in ASAP die folgenden Schlüsselemente enthalten:

1. Roadmap (Projektplan mit genauen Angaben darüber, welche Aufgaben wann, von wem warum und wie durchgeführt werden müssen),

2. Werkzeuge und Beschleuniger (Spreadsheets, Formulare, Fragebögen, Checklisten, Vorlagen, Werkzeuge für die Systemkonfiguration, Interface Advisor usw.) sowie
3. Serviceleistungen in den Bereichen Beratung, Ausbildung Support, z. B. Hotline, Early Watch Service sowie Schulungsstrategien für Projektteams und Endanwender.

Diese Elemente werden durch den sogenannten Implementation Assistant (Abbildung 3-1) entweder direkt zur Verfügung gestellt oder zugänglich gemacht.

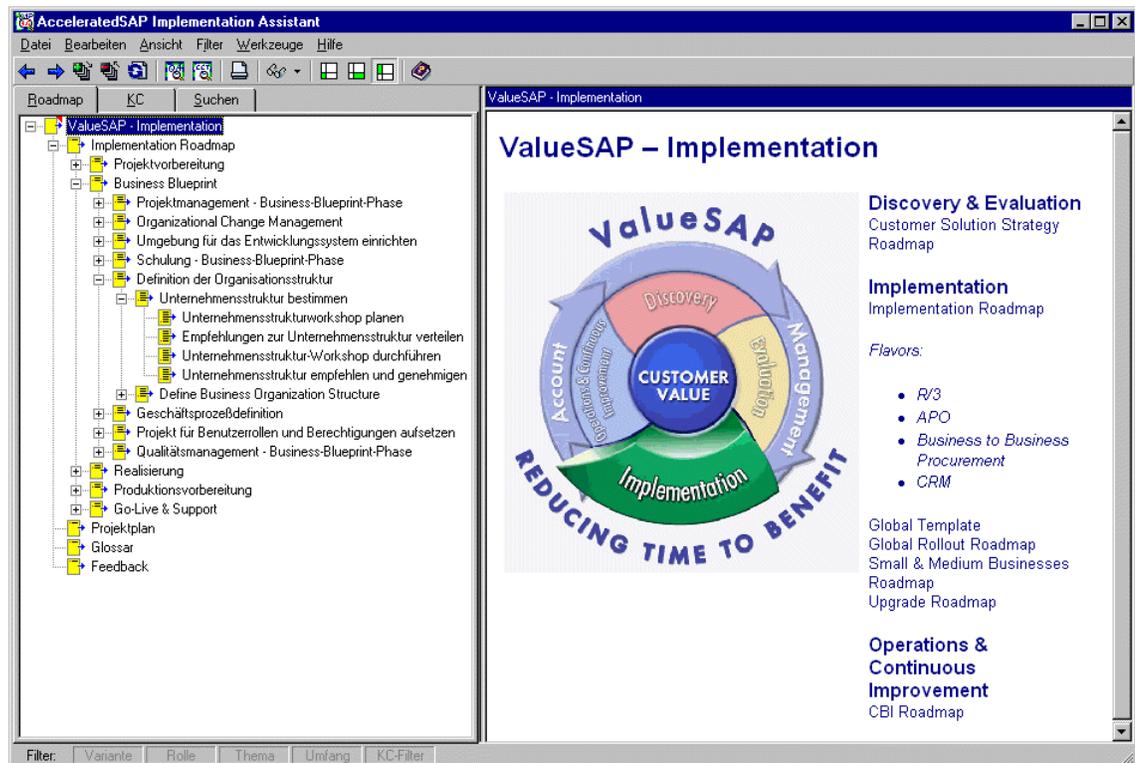


Abbildung 3-1: Implementation Assistant von ASAP [SAP00d]

STRELLER fasst die Zielsetzungen der ASAP-Entwicklung wie folgt zusammen [STRE99, S. 61-62]:

- Rasche R/3-Implementierung,
- einheitliche Vorgehensweise der Beratungspartner,
- effiziente Nutzung der vorhandenen Ressourcen,
- gesicherte Qualität,

- nachgewiesenes Know How, Wiederverwendung vorhandener Ergebnisse für nachfolgende Einführungsphasen und
- reduzierte Einführungskosten sowie schnellerer ROI.

3.1.1.1 KONZEPTIONELLE GRUNDLAGEN VON ASAP

AcceleratedSAP (ASAP) stellt eine umfassende Lösung zur effizienten Einführung sowie kontinuierlichen Verbesserung der SAP-Software dar und zeichnet sich durch einen praxisorientierten Aufbau, einen hohen Detaillierungsgrad und eine flexible Gestaltung von Einführungsprojekten aus. Das Vorgehensmodell wird unabhängig von der SAP-Software ausgeliefert und erlaubt somit einen frühzeitigen Beginn des Implementierungsprojekts ohne Installation des Systems.

Dem ASAP-Ansatz liegen drei wesentliche Konzepte zugrunde, die im folgenden kurz erläutert werden sollen [TEUF99, S. 77-78].

PROZESSORIENTIERUNG

ASAP fördert durch die konsequente Prozessorientierung das bereichsübergreifende Denken und Zusammenarbeiten. Alle Hilfsmittel, die in dieser PC-gestützten Lösung beinhaltet sind bzw. zur Verfügung stehen, sind prozessorientiert ausgerichtet, damit die in der Software systemimmanent vorhandenen Optimierungspotenziale vollständig erschlossen werden können.

REFERENZBASIERTE IMPLEMENTIERUNG

Bei der Implementierung wird auf Referenzprozesse zurückgegriffen, die als Standardabläufe definiert wurden und somit die Nutzung des Verbesserungspotenzials der Softwarebibliothek bei einer kurzen Projektlaufzeit sicherstellen. Der augenscheinliche Nachteil dieses Ansatzes, dass sich ein Unternehmen an den Standardprozessen des Systems orientiert und dadurch seine Individualität gegenüber der Konkurrenz verliert, wird dabei durch die Vielzahl an Einstellungs- und Kombinationsmöglichkeiten sowie die Offenheit (Business Framework) entkräftet. ASAP basiert auf denselben Prämissen. Dieses Konzept lässt ausreichend Spielraum für individuelle Prozessabläufe, ein Reengineering auf der grünen Wiese wird jedoch nicht unterstützt.

BIG BANG-IMPLEMENTIERUNGSSTRATEGIE

Aus der konsequenten Prozessorientierung resultiert die Favorisierung der Big Bang-Implementierungsstrategie durch ASAP. Dabei werden nicht, wie in den herkömmlichen Vorgehensweisen üblich, nacheinander verschiedene Komponenten (Module) eingeführt, statt dessen werden ausgesuchte Hauptprozessbereiche in ihrer Gesamtheit implementiert. Ein Beispiel hierfür ist die Kundenauftragsabwicklung, welche die Bereiche Vertrieb, Produktion, Beschaffung, Finanzbuchhaltung und Controlling umfasst. Somit werden in einem Schritt (Big Bang) viele Komponenten der SAP-Software (z. B. SD, MM, PP, FI etc.) eingeführt. Diese Vorgehensweise bedingt jedoch, dass die einzelnen Module nicht bis in das kleinste Detail umgesetzt, sondern vielmehr die wichtigsten Bestandteile in einem initialen Projekt implementiert werden, so dass der Prozess über alle integrierten Bereiche ablauffähig ist. Die systemimmanente Flexibilität einer Softwarebibliothek erlaubt jederzeit die Erweiterung der eingeführten Funktionalitäten.

3.1.1.2 PHASEN VON ASAP

Die generelle Vorgehensweise folgt den durch die Roadmap illustrierten Phasen und Inhalten, die nachfolgend im Detail erläutert werden. Am Ende jeder Phase schließen sich Qualitäts-Checks an, die wiederum durch zusätzliche Werkzeuge unterstützt werden. Im einzelnen werden folgende Phasen durch ASAP abgedeckt [SAP00a, SV-ASA]:

PHASE 1: PROJEKTVORBEREITUNG

Das Ziel dieser Phase ist es, alle Vorbereitungen für einen reibungslosen Ablauf des Gesamtprojekts zu treffen und die dafür notwendigen Grundvoraussetzungen zu schaffen. ASAP zeigt in dieser ersten Phase auf, wie folgende Punkte umgesetzt werden können:

- Festlegung des groben Projektumfangs,
- Erstellung des Projektauftrags sowie der Projektorganisation,
- Detaillierte Planung bezüglich Zeiten, Kosten, Ressourcen und Terminen,
- Planung des technischen Umfelds und die Festlegung von Standards, z. B. für die Dokumentation, Kommunikation usw.

Am Ende dieser Phase findet in der Regel ein Projekt-Kick-Off statt, der den Übergang zur zweiten Phase darstellt.

PHASE 2: BUSINESS BLUEPRINT

Hauptziel des Business Blueprint ist die Dokumentation der Anforderungen unter Berücksichtigung organisatorisch-, geschäftsprozess- und technischer Gesichtspunkte. Dabei werden im Rahmen des Detailscoping der Projektumfang ausführlich festgelegt und in zahlreichen Workshops die Anforderungen an Organisationsstruktur und Geschäftsprozesse gesammelt. ASAP stellt dafür eine Datenbank mit entsprechenden Checklisten bzw. Fragen zur Verfügung, in der die Antworten des Kunden, teils strukturiert, teils als Freitext, erfasst werden. Neben diesen Fragebögen kommen auch grafische Prozessketten zum Einsatz, die unterstützend und beschleunigend eingesetzt werden können. Insbesondere die Visualisierung bereits vorhandener und ablauffähiger Prozesse über Referenzmodelle spielen in dieser Phase eine entscheidende Rolle.

Das Hauptwerkzeug von ASAP, die Question and Answer Database (Q&Adb), spiegelt die Mehrzahl dieser Hilfsmittel wider. Dabei bildet die Q&Adb in Abhängigkeit des jeweiligen Releasestands der SAP-Software das Referenzmodell des Systems ab und beinhaltet allgemeine Fragen zu Unternehmensstrategie, Organisationsstruktur und -einheiten einerseits sowie Unternehmensprozessbereichen andererseits. Abbildung 3-2 zeigt einen Ausschnitt der Q&Adb für das Referenzmodell 4.6C.

Das Ergebnis dieser Phase ist ein Konstruktionsplan, der zeigt, wie das einführende Unternehmen in der SAP-Software abgebildet werden soll (Implementierung) bzw. wie die Anforderungsänderungen des Unternehmens umgesetzt werden müssen (Continuous Improvement). Das Business Blueprint-Dokument (Soll-Konzept) bildet das zentrale Medium, da alle nachfolgenden Aktivitäten auf ihm aufbauen.

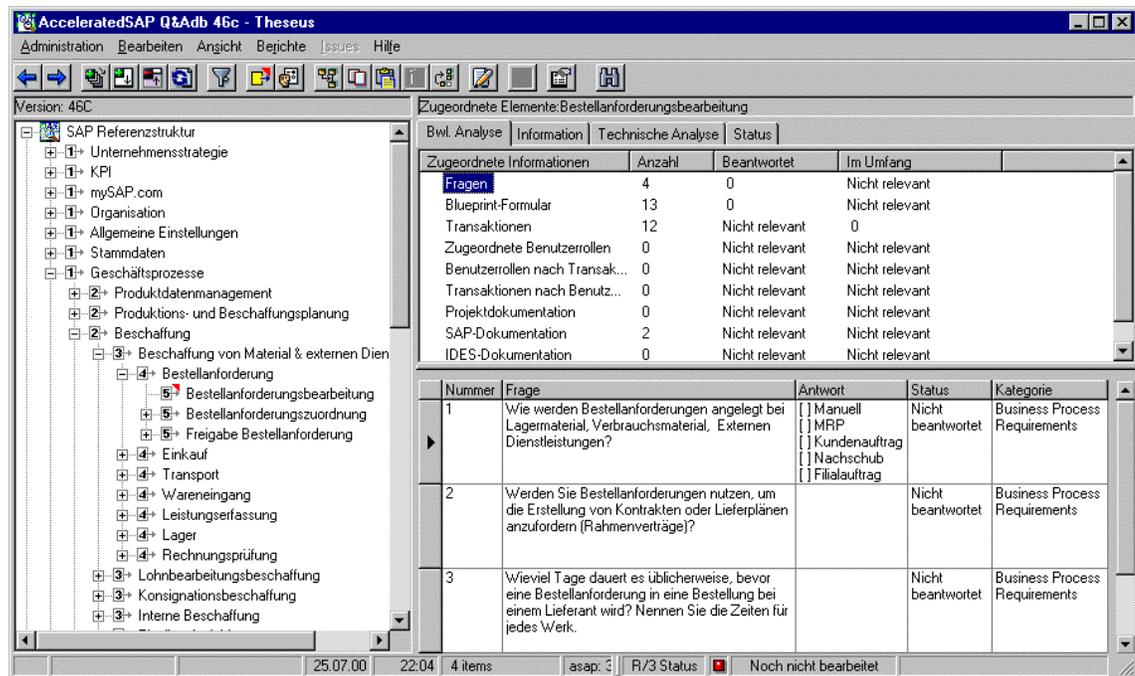


Abbildung 3-2: Question and Answer Database (Ausschnitt) [SAP00d]

PHASE 3: REALISIERUNG

Die Realisierungsphase ist durch viele parallel ablaufende Aktivitäten gekennzeichnet. Auf Basis des Business Blueprint wird die Konfiguration des Systems gemäß den ermittelten und formulierten Anforderungen vorgenommen. Dabei kommt ein weiteres Werkzeug, die Business Process Master List (BPML) zum Einsatz, die den gesamten Konfigurationsprozess bezüglich Dokumentation, Meilensteinüberwachung, Testausführung usw. verwaltet. Für die eigentliche Einstellung der Systemparameter wird der prozessorientierte und direkt von der BPML aus zugängliche Einführungsleitfaden (IMG) verwendet.

Die Realisierung wird von ASAP vor allem in zwei Schritten unterstützt. Zuerst wird ein Grundsystem (Baseline) eingerichtet, das ca. 70-80% der wichtigsten Geschäftsprozesse und die gesamte Organisationsstruktur enthält [TEUF00, S. 74]. Bei diesem System handelt es sich nicht um einen Prototyp klassischer Ausprägung, der nicht weiter verwendet wird, vielmehr bildet es die Grundlage für das spätere Produktivsystem, das alle Anforderungen enthält. Parallel zu den System-einstellungen finden Detailschulungen für das Projektteam statt.

Falls notwendig, werden Schnittstellenprogramme, eigene Berichte, Erweiterungen und Formulare (z. B. für Rechnungen) erstellt bzw. programmiert. Außerdem wird

das Berechtigungskonzept aufgebaut und die Endanwenderdokumentation verfasst. Umfassende Integrations- und Funktionstests schließen diese Phase ab.

PHASE 4: PRODUKTIONSVORBEREITUNG

Die vierte Phase innerhalb der ASAP-Roadmap ist durch zahlreiche Funktionalitätstests und das Training der Endanwender bestimmt. Hauptaufgabe dabei ist die intensive Vorbereitung auf einen reibungslosen und problemfreien Produktivstart. Die in Phase 3 pro Unternehmensprozessbereich durchgeführten Integrationstests werden auf das gesamte System ausgeweitet. Darüber hinaus erfolgen sogenannte Massen- und Stresstests, mit denen versucht wird, das erwartete Datenvolumen unter Performance-Gesichtspunkten und Produktivbedingungen durchzuspielen, um die Stabilität des Systems sicherzustellen. Die Tests schließen alle Schnittstellenprogramme, Berichte, Formulare usw. mit ein.

Für alle offenen und kritischen Punkte müssen in dieser Phase Lösungsvorschläge gefunden werden. Der letzte Schritt in dieser Phase wird die Migration der Stamm- und Bewegungsdaten in das neue System sein. Dazu wird ein detaillierter Plan zur Übernahme der relevanten Daten oder Einstellungen in das Produktivsystem erstellt.

PHASE 5: GO-LIVE UND SUPPORT

Ziel dieser Phase ist der reibungslose Ablauf des Übergangs vom projektorientierten, vorläufigen System zu einem erfolgreichen Produktivstart. Wichtiger Bestandteil ist dabei die Bereitstellung aller verfügbaren Dienste (z. B. OSS-Online Service System) zur Unterstützung der SAP-Anwender. Dies muss sowohl in der Anfangsphase nach dem Produktivstart, als auch als Grundlage für spätere Veränderungen gewährleistet sein. Gerade zu Beginn der Produktivphase treten bei den End-Anwendern zahlreiche Fragen auf, die durch einen schnell verfügbaren Service abgefangen werden müssen. Außerdem muss das laufende System ständig bezüglich der Performance, aber auch seiner Arbeitsfähigkeit und Akzeptanz in den Abteilungen überprüft werden.

Diese Phase geht direkt in die Aufgaben und Tätigkeiten des Continuous Change im Sinne des CSE über. Dabei kann es sich um Änderungen handeln, die entweder technische Gründe (z. B. Releasewechsel), betriebswirtschaftliche Gründe (z. B. Erweiterung der Funktionalität) oder gesetzliche Gründe (z. B. neue Steuergesetze)

haben. Die sich hier anschließenden Phasen werden ebenfalls durch ASAP in Form eigener Roadmaps unterstützt.

3.1.1.3 WERKZEUGVERWENDUNG IN ASAP

Bei der Betrachtung der in ASAP verwendeten Werkzeuge muss man zwei Kategorien unterscheiden. Zum einen sind Unterstützungswerkzeuge des Vorgehensmodells in das SAP-System selbst integriert und stehen auch anderen Einführungskonzepten zur Verfügung. Als Beispiel sei an dieser Stelle der bereits erwähnte Einführungsleitfaden (Implementation Guide, IMG) genannt, der zum besseren Verständnis kurz beschrieben wird, obwohl er kein Werkzeug von ASAP im klassischen Sinn darstellt. Andere Werkzeuge, die teilweise schon bei der Beschreibung der Phasen angesprochen wurden, sind generischer Bestandteil des Vorgehensmodells, z. B. die Fragen und Antwort-Datenbank (Q&Adb).

EINFÜHRUNGSLEITFADEN

Der Einführungsleitfaden ist das Werkzeug zur Anpassung des SAP-Systems an die Anforderungen eines Unternehmens. Mit seiner Hilfe wird die Konfiguration des Systems strukturiert und organisiert.

Der Einführungsleitfaden ist nach Anwendungsbereichen in einer hierarchischen Struktur gegliedert, welche die Komponentenhierarchie abbildet. Er enthält alle für die Einführung notwendigen Arbeitsschritte mit der dazugehörigen Dokumentation.

Mit Hilfe des Einführungsleitfadens werden

- SAP-Funktionen schnell, sicher und kostengünstig konfiguriert,
- Standardfunktionen auf die individuellen Unternehmensanforderungen zugeschnitten,
- Einführungs- oder Erweiterungsprojekte verwaltet, bearbeitet und analysiert,
- Einführungsphasen mit einem Projektmanagement-Werkzeug dokumentiert bzw. überwacht und

- Konfigurationsinformationen automatisch aus dem Qualitätssicherungssystem in das Produktivsystem übertragen, wodurch die Konsistenz gewährleistet ist.

Der Einführungsleitfaden spielt bei den weiteren Betrachtungen vor allem im Hinblick auf die Umsetzung der THESEUS-Methode eine zentrale Rolle.

QUESTION & ANSWER DATABASE (Q&ADB)

Die Q&Adb stellt das zentrale Werkzeug innerhalb von ASAP dar und wird hauptsächlich während der Blueprint-Phase eingesetzt. Sie erlaubt ein prozessorientiertes Vorgehen bei der Auswahl des Projektumfangs und der Festlegung des Sollkonzeptes. Neben der Standard-Q&Adb sind Versionen für verschiedene Industrien (z. B. Konsumgüterindustrie, Automobilindustrie) verfügbar.

Den Strukturelementen der SAP-Referenzstruktur sind Elemente zugeordnet, die in einem eigenen Bildbereich über Registerkarten (z. B. die Betriebswirtschaftliche Analyse, Information und Technische Analyse) gruppiert sind (vgl. Abbildung 3-2). Die Bereiche beinhalten zusätzliche Informationen, wie z. B. SAP-Dokumentation, Transaktionen, Input/Output und Organisationseinheiten. Zudem werden unternehmensspezifische Informationen gesammelt, wie z. B. Projektdokumentation, Fragen, Formulare und technische Daten (Datenübernahme, Schnittstellen etc.).

Als Datenbank enthält die Q&Adb ein flexibles Berichtswesen, das u. a. Überblicksberichte, Business Blueprint-Berichte und Masterlists (Business Process Master List, Organisationsstrukturliste, Entwicklungsliste, Berechtigungsliste) bereitstellt. Für offene bzw. unbewältigte Fragen und Probleme ist die sogenannte Issues-Datenbank integriert, die eine Verzögerung des Projektfortschritts durch mangelnde Lösungskonzepte verhindern soll [SAP00a, SV-ASA].

BUSINESS PROCESS MASTER LIST

Die Business Process Master List (BPML) stellt zusammen mit dem Business Blueprint das wichtigste Ergebnis der zweiten Phase dar. Microsoft Excel Tabellen bieten eine Zusammenstellung aller SAP-Szenarios, Geschäftsprozesse und Transaktionen, die in den Projektumfang genommen wurden. Während der nachfolgenden Realisierungsphase bildet die BPML die Grundlage für die Überwachung und Steuerung der Konfiguration und der Testaktivitäten.

Die BPML wird als Bericht aus der Q&Adb generiert und bietet folgende Möglichkeiten:

- Anzeige von verknüpften Dokumenten, wie z. B. Arbeitsablaufbeschreibungen (BPPs) oder Blueprint-Formulare,
- Aufruf eines SAP-Systems sowie Zugriff auf Transaktionen, deren Funktionalität eruiert werden soll sowie
- Navigation von einzelnen Prozessen zu genau den IMG-Aktivitäten, die für die Konfiguration des jeweiligen Prozesses durchzuführen sind.

Über die BPML wird die Reihenfolge der Realisierung, des Testens und der Abnahme konfigurierter Funktionalität des SAP-Systems festgelegt.

PROJECT ESTIMATOR

Der Project Estimator unterstützt in der Projektvorbereitungsphase die Planung des Einführungsprojektes. Dabei finden sowohl der Projektumfang als auch verschiedene Risikofaktoren und die Komplexität der Geschäftsprozesse Berücksichtigung. Mit dem Einsatz des Project Estimators soll der Grad an Planungsgenauigkeit erhöht werden.

Der Project Estimator erlaubt es, erforderliche externe und interne Ressourcen zu bestimmen, den Zeitrahmen der Implementierung abzustecken und eine erste Kostenschätzung vorzunehmen [STRE99, S. 69].

CONCEPT CHECK TOOL

Das Concept Check Tool ist ein Analyseinstrument, mit dem sich Qualitätsprüfungen der Projekt- und Implementierungsarbeiten während der ersten beiden Phasen durchführen lassen. Die Beantwortung von Checklisten ermöglicht es frühzeitig Issues und potentielle Konflikte bezüglich Datenvolumen und Konfiguration zu erkennen und darauf zu reagieren.

Das Concept Check Tool enthält Checklisten, die eine detaillierte Analyse der Systemeinstellungen erlauben. Diese sind dynamisch konzipiert, d. h. bei der Beantwortung jeder Frage werden die nachfolgenden Fragen in Abhängigkeit von der aktuellen Antwort ausgewählt. Die Strukturierung der Checklisten zur Phase Projektvorbereitung entspricht der Roadmap-Struktur, in der Business Blueprint Phase bilden sie die SAP-Referenzstruktur ab. In der Version 4.6C sind

bilden sie die SAP-Referenzstruktur ab. In der Version 4.6C sind Checklisten zu Organisationseinheiten, Stammdaten und Geschäftsprozessen verfügbar. Die Fragestellungen sind überwiegend als Ja/Nein- oder Multiple Choice-Fragen konzipiert und überprüfen insbesondere die Projektorganisation, grundlegende System-einstellungen sowie Performanceaspekte. Um zu jedem Zeitpunkt einen Überblick über den Aufbau der Checkliste zu ermöglichen, werden sowohl die Fragefolge als auch die bislang gegebenen Antworten in einem eigenem Bildbereich neben dem aktiven Frage-/Antwortrahmen angezeigt.

In Phase 2 wird der in der Q&Adb festgelegte Projektumfang in das Concept Check Tool importiert. Es werden dann genau die Checklisten zur Bearbeitung vorgelegt, die für das Einführungsprojekt relevant sind.

Nach Abschluss jeder Checkliste wird ein Bericht generiert. Dieser Bericht

- gibt Empfehlungen zur Verbesserung der Konfiguration,
- zeigt kritische Datenmengen bzw. Werte auf und
- enthält wichtige Informationen für die Durchführung von Streßtests und Going Live Checks.

Nachdem alle Checklisten abgearbeitet wurden, kann aus den Einzelberichten ein Gesamtbericht generiert werden, der einen Überblick über die wesentlichen Charakteristika aller Kernfunktionen gibt. Abbildung 3-3 zeigt einen Ausschnitt aus dem Concept Check Tool für die Phase des Business Blueprint.

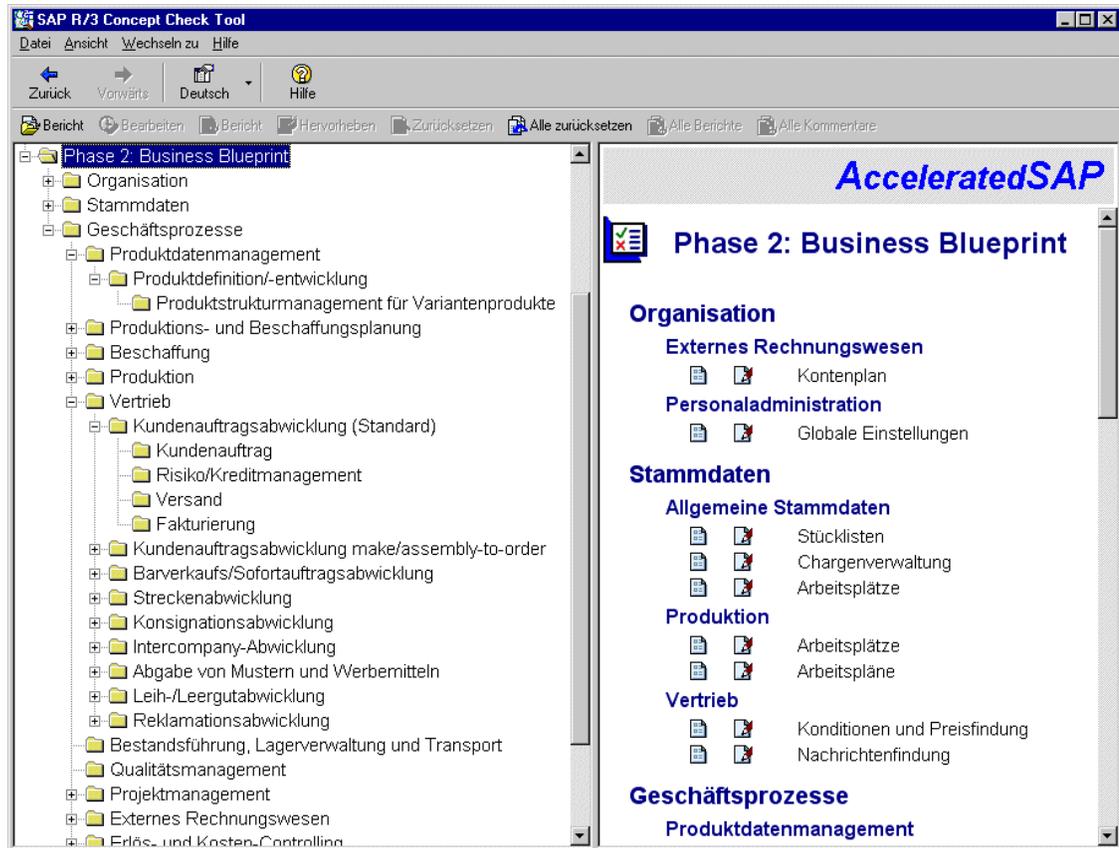


Abbildung 3-3: Concept Check Tool (Ausschnitt) [SAP00d]

3.1.1.4 ERWEITERUNG DES GRUNDMODELLS

Die vielfältigen Anforderungen unterschiedlicher Branchen sind nur sehr schwer in einem Standardanwendungssystem vereinbar. Um diesen Anforderungen gerecht zu werden, hat die SAP AG branchenspezifische Gesamtlösungen entwickelt, welche die Infrastruktur für eine durchgängige Unterstützung der Kernprozesse bereitstellen. Diese sogenannten Industry Business Solutions (z. B. SAP Utilities, SAP Banking usw.) beinhalten neben einem um branchentypische Komponenten erweiterten SAP-System auch weitere Elemente, wie z. B. die Einbindung von komplementären Softwarebausteinen oder ASAP [TEUF99, S. 79]. Organisatorisch werden diese Speziallösungen in Form von Industry Business Units (IBUs) bereitgestellt. Dabei kann es vorkommen, dass ein Großteil der Prozesse und Funktionen aus der Kernentwicklung weitestgehend ohne Modifikation übernommen wird, je nach Branche variiert demnach der Anteil an spezifischen Erweiterungen.

Zu jeder Branchenlösung gibt es von Seiten der SAP AG ein Geschäfts- und Planungsmodell, das die funktionalen Anforderungen auf verschiedenen Ebenen detailliert darstellt. Diese Solution Maps erweitern das Vorgehensmodell ASAP um branchenspezifische Informationen und geben einen Überblick über die Vorstellungen der SAP AG und ihrer Partner von Geschäftsprozessen und Funktionen in einer Branche. Dabei werden über verschiedene Ebenen hinweg die Funktionalitäten gezeigt, die notwendig sind, um bestimmte Prozesse zu steuern. Mit Hilfe eines weiteren Werkzeugs, dem Solution Map Composer wird es dem Kunden ermöglicht, auf Basis einer Branchenlösung eine eigene Map zu erstellen und in dieser beispielsweise Fremdsysteme zur Abbildung von Spezialprozessen abzubilden. Aus der so generierten Kunden-Solution Map kann dann automatisch der Projektumfang in der Q&Adb festgelegt und geladen werden. Die Branchenausrichtung spiegelt sich in vorkonfigurierten Systemen, Solution Maps und eigenen Q&Adb's wider. Neue Lösungen der SAP AG, z. B. in Form der mySAP.com-Komponenten, werden ebenfalls durch eigene Roadmaps und spezielle Question & Answer Databases in das Vorgehensmodell eingebunden [ASAP00, SV-ASA].

3.1.2 LIVE METHOD/Chestra

Seit 1997 wird das von der Firma CSC Plönzke entwickelte Vorgehensmodell Chestra von der Siemens Nixdorf Informationssysteme AG (SNI) als ein flexibles Methoden-Framework eingesetzt.

Diese Vorgehensweise wird mit dem Fokus der Veränderungen in großen Organisationen angewandt und bietet einen kompletten Prozess für den geschäftlichen Wandel an. Die Anpassung an kundenindividuelle Anforderungen im Rahmen eines einheitlichen methodischen Vorgehens sowie eines durchgängigen Beratungskonzepts steht dabei im Vordergrund.

Chestra behandelt alle Phasen der Geschäftsprozessentwicklung und unterstützt die Einleitung, den Entwurf, die Motivation, die Durchführung, die Steuerung sowie die Koordination des angestrebten Wandels [CHES99]. Die ausgewählten Phasen des Projekts werden mit exakt definierten Arbeitspaketen und Meilensteinen durchlaufen. In jeder einzelnen Phase werden die Aktivitäten mit ihren zu erstellenden Arbeitsergebnissen beschrieben.

Das Vorgehensmodell wurde ursprünglich unter dem Namen Catalyst von CSC (Computer Sciences Corporation) entwickelt. Dabei stand die Einbindung praktischer Erfahrungen im Vordergrund. 1995 übernahm SNI die Lizenz für Catalyst und passte es als Methoden-Framework unter der Bezeichnung Chestra an deutsche und europäische Verhältnisse an.

Zur Verbesserung der Einführung der Standardanwendungssoftware R/3 auf Basis des Continuous System Engineering wurden die bereits vorhandenen Werkzeuge zur Adaption der SAP-Software, z. B. LIVE KIT Structure, LIVE KIT Power, LIVE Migration und LIVE Project, in das Vorgehensmodell integriert. Seit dieser Integration wurde der Name des Modells zu LIVE METHOD/Chestra erweitert.

Für die nachfolgenden Untersuchungen stand die Version 1.1 Productivity Toolkit for Chestra [CHES99] zur Verfügung.

3.1.2.1 KONZEPTIONELLE GRUNDLAGEN VON CHESTRA

Die folgenden Grundlagen dienen zum einen dazu, den Aufbau von Chestra zu erläutern, zum anderen sollen die Unterschiede zu ASAP verdeutlicht werden.

Chestra ist eine Methode, um Veränderungen in Organisationen zu initiieren, zu begründen, zu designen, zu entwickeln und zu koordinieren.

GESCHÄFTSPROZESS-GETRIEBENER ANSATZ

Das Unternehmen wird grundsätzlich aus Sicht der Geschäftsprozesse betrachtet. Der erste Schritt ist die Ermittlung der Unternehmensziele, dem sich die Ableitung der Ziele für die Geschäftsprozesse anschließt. Aufgrund dieser Informationen werden im nächsten Schritt neue oder veränderte Prozesse entwickelt. Durch diese streng prozessorientierte Vorgehensweise wird gewährleistet, dass die Entwicklungsaktivitäten nahe an den globalen Geschäftszielen ausgerichtet sind. Die Geschäftsprozesse bilden somit die Basis für die Ableitung der weiteren Prioritäten.

KOMMUNIKATION UND PARTNERSCHAFT

Der Anwender des Systems wird in den Entwicklungs- bzw. Adaptionprozess einbezogen und zur Mitarbeit motiviert, zusätzlich werden über die Beteiligung an fast allen Entwicklungsschritten ein Konsens aufgebaut und Unstimmigkeiten weitestgehend unterbunden. Ein weiteres Ziel der engen und stetigen Kooperation

von Anwendern und Entwicklern bzw. Beratern ist die Vermeidung unnötiger formaler Spezifikationen, die sich negativ auf den Projektverlauf auswirken.

STETIGE ÄNDERUNGEN

Die Schnelllebigkeit der Märkte induziert ständige Veränderungen im Geschäftsumfeld und bei den Anforderungen der Anwender. Der CSE-Gedanke in Verbindung mit Chestra trägt dieser Tatsache Rechnung und versucht, eine frühzeitige und strenge Anforderungsspezifikation zu vermeiden. Vielmehr liegt der Fokus auf einer breiten Architektur. Die detaillierte Spezifikation wird erst in späten Projektphasen bei der Erstellung des Systemdesigns erstellt. Dieses Vorgehen verkürzt die Zeit zwischen der Definition der Anforderungen und der tatsächlichen Umsetzung in Form der Adaption der Softwarebibliothek. Das produktiv genutzte System erfüllt somit die Geschäftsziele besser als die bisherigen Vorgehensweisen.

MODELLBASIERTE ENTWICKLUNG

Chestra ermöglicht die Definition von Modellen, mit deren Hilfe Unternehmen aus unterschiedlichen Sichtweisen mehrdimensional beschrieben werden können. Dies erlaubt die Beantwortung von Fragen aus unterschiedlichen Blickwinkeln auf das Unternehmen. Die Modelle geben Aufschluss über die Geschäftsprozesse, die Organisation, die geographische Verteilung sowie über die für die Geschäftsprozesse notwendigen Anwendungen, Daten und Technologien.

PRAGMATISCHE, ERGEBNISORIENTIERTE METHODEN

Chestra bzw. das Methoden-Framework ist kein wissenschaftlicher Ansatz. Die Grundlage für die inhaltliche und methodische Entwicklung bilden Erfahrungen aus dem Projektalltag. Der Einsatz von Chestra ermöglicht

- die Standardisierung der Arbeitsergebnisse,
- eine schnelle Ermittlung konkreter Ergebnisse,
- die Unterstützung des Projektteams bei der Auswahl eines geeigneten Engineering-Ansatzes,
- eine verbesserte Terminplanung im Rahmen der Projektdefinition,
- die Definition klarer Zielvorgaben sowie die Überwachung deren Einhaltung und

- die Gewährleistung der Vollständigkeit und Konsistenz der Projekte [CHES99].

Das Unternehmen wird aus sechs verschiedenen Sichten betrachtet. Dazu zählen Geschäftsprozesse, Organisationsstrukturen, Standorte, Anwendungen, Technologien und Daten. Diese Sichten werden modelliert und in den Phasen des Frameworks immer weiter verfeinert.

3.1.2.2 PHASEN VON CHESTRA

Das Methoden-Framework kann an das jeweilige Projekt angepasst werden. Dazu ist zunächst zu prüfen, welche Phase den Einstiegspunkt in das Framework bildet. Die dann zu durchlaufenden Phasen werden auf die bereits vorhandenen Arbeitsergebnisse überprüft, anschließend werden die noch zu erstellenden Ergebnisse definiert. Bei alternativen Pfaden wird die Eignung der einzelnen Alternativen evaluiert und letztendlich der für das Projekt am besten geeignete Weg bestimmt.

Das LIVE METHOD/Chestra-Modell besteht aus folgenden fünf Phasen (Abbildung 3-4):

- Implementation Vorbereitung,
- Implementation Analyse,
- Implementation Design,
- Implementation Aufbau und
- Implementation Einführung.

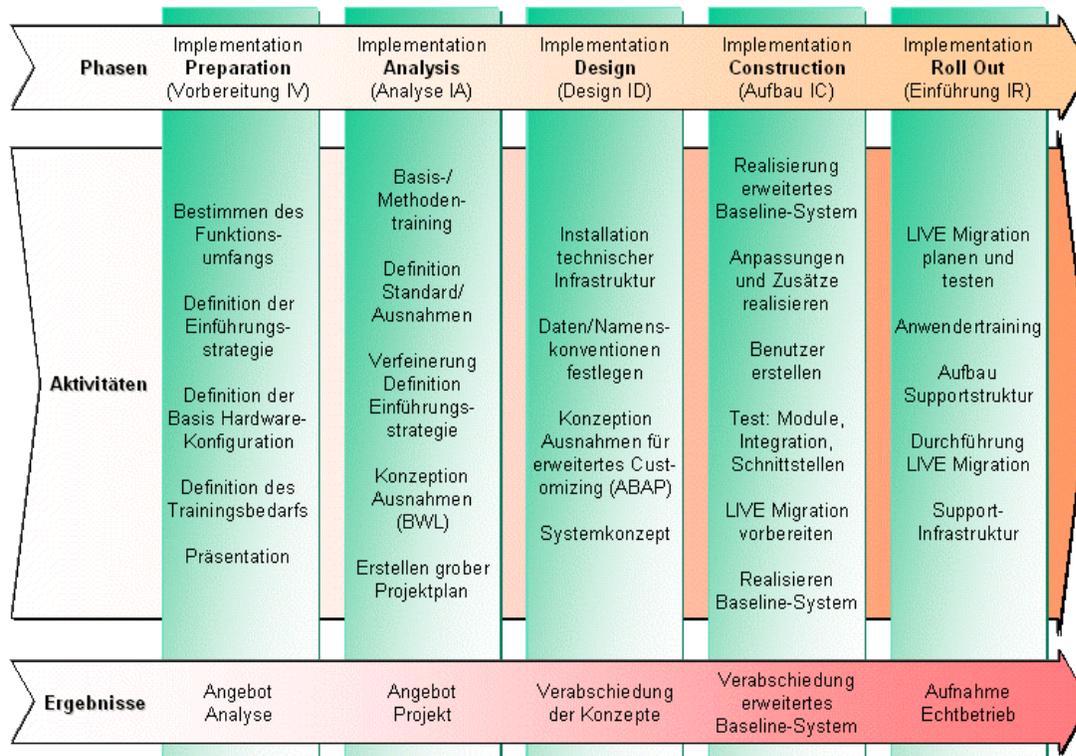


Abbildung 3-4: Aktivitäten im Implementierungsprozess von LIVE METHOD/Chestra

Diese fünf Phasen wurden speziell für die Adaption der SAP-Software angepasst. Die einzelnen Aktivitäten werden in Zusammenarbeit mit dem Anwender ausgewählt und durch die im folgenden Kapitel kurz beschriebenen LIVE Tools unterstützt.

3.1.2.3 WERKZEUGVERWENDUNG IN CHESTRA

Chestra wurde für die generelle Ausführung eines strukturellen Wandels entwickelt. Die Erweiterung bzw. Widmung des Vorgehensmodells für die Verwendung zur Adaption der SAP-Software wird u. a. durch den Einsatz der LIVE-Tools erreicht. Tabelle 3-2 gibt einen Überblick der in Chestra verwendeten Werkzeuge.

Tabelle 3-2: Übersicht der gewidmeten Werkzeuge [in Anlehnung an STRE99, S. 76-77]

Werkzeug	Beschreibung
LIVE Project	LIVE Project unterstützt alle Projektmanagementaufgaben. Die wesentlichen Eckpfeiler des Projektes (Organisation, Planung, Kommunikation und Steuerung) werden durch dieses auf MS Project basierende Werkzeug unterstützt.
LIVE KIT Structure	LIVE KIT Structure ist ein PC-basiertes Expertensystem zur schnellen und wirtschaftlichen R/3-Einführung. Die Einsatzmöglichkeiten der Software werden durch verständliche Fragen transparent. Die Auswertungskomponente ermöglicht die Zusammenstellung aller Ergebnisse in kundenindividuellen Reports. Dieses Werkzeug ist aus dem ODYSSEUS-Ansatz von HUFgard entstanden [HUF94].
LIVE KIT Power	LIVE KIT Power unterstützt die Einführung durch Analyse und Darstellung prozessrelevanter Informationen, um Ergänzungsentwicklungen zu planen, Lücken zu identifizieren und organisatorische Problemlösungen zu entwickeln. Dieses Werkzeug wurde von VOGELsANG im Rahmen des PENELOPE-Konzepts entwickelt [VOGE97].
LIVE KIT Control	LIVE KIT Control unterstützt die Einführung bei der Analyse und Darstellung von Informationen zur Berichtsadaption durch den Berichtshierarchie- bzw. Berichtobjekt-Monitor. Dieses Werkzeug wurde von WEDLICH im Rahmen des MENTOR-Konzepts entwickelt [WED97].
LIVE AG	Die LIVE AG ist eine Modellfirma, die alle SAP-Standardmodule integrativ einsetzt. Sie kann für Schulungen oder als Selbstlernmedium genutzt werden, aber genauso zu Tests neuer Funktionalitäten oder für Präsentationen eingesetzt werden. Neben dem eigentlichen System, besteht die LIVE AG aus einer umfangreichen Dokumentation der Customizingeinstellungen, Stammdaten und Prozessabläufe in Form von Fallstudien.
LIVE Migration	LIVE Migration dient der maschinellen Übernahme von Stamm- und Bewegungsdaten aus Altsystemen in das SAP-System.
LIVE KIT Process	Die wesentlichen Abläufe der R/3-Standardprozesse können in einer frühen Phase der Einführung transparent gemacht werden. Workflows können graphisch dargestellt werden. Es wird eine tabellarische Dokumentation der Anforderungen je Funktion/Aktivität erstellt.

Die einzelnen Werkzeuge finden innerhalb der Vorgehensweise an unterschiedlichen Stellen Verwendung. In der Regel handelt es sich nicht um einen punktuellen Einsatz, sondern um projektbegleitende Unterstützung. Ergebnisse des Werkzeugeinsatzes können somit in späteren Projektphasen (z. B. Re-Adaption) ebenso hilfreich sein.

Aus Abbildung 3-5 wird deutlich, an welchen Stellen eines Einführungsprojektes die beschriebenen Werkzeuge innerhalb von Chestra eingesetzt werden können.

Im Rahmen des Continuous Improvement werden die einzelnen Phasen zyklisch durchlaufen, wobei jeweils auf die Ergebnisse der entsprechenden Vorgängerabschnitte zurückgegriffen werden kann.

Die in LIVE METHOD/Chestra integrierten Werkzeuge wurden mit der Prämisse der Methodenunabhängigkeit entwickelt, das bedeutet, dass diese Tools auch im Rahmen anderer Vorgehensmodelle zum Einsatz kommen, um dort Unterstützungsarbeit zu leisten. STRELLER hat sich ausführlich mit der Einbindung der Werkzeuge aus der LIVE-Familie in ASAP beschäftigt, um exemplarisch die Flexibilität hinsichtlich ihrer methodischen Verwendbarkeit aufzuzeigen [STRE99, S. 79-82].

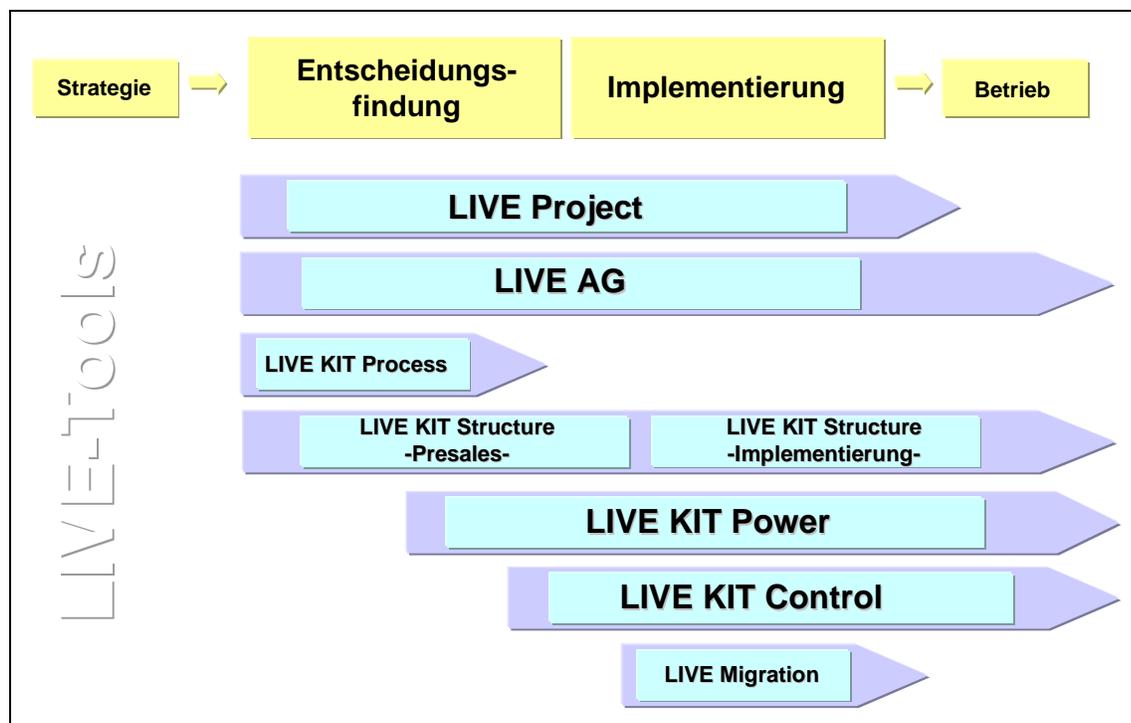


Abbildung 3-5: Einsatzzeitpunkt der integrierten Werkzeuge in Chestra

3.1.3 Sonstige Vorgehensmodelle

Die Beratungsunternehmen, die Unterstützung der Adaption der SAP-Software anbieten, haben in der Regel eigene Vorgehensweisen und Methoden entwickelt, um sich von Konkurrenzunternehmen zu unterscheiden. Das Systemhaus debis bietet beispielsweise ein durchgängiges Vorgehensmodell zur Einführung der SAP-

Software an, das verschiedene Analysewerkzeuge beinhaltet [DEBI00]. STRELLER hat über eine Befragung unter den führenden Beratungshäusern 1998 die entwickelten Vorgehensmodelle analysiert und bewertet. Das Ergebnis der Studie soll an dieser Stelle kurz zusammengefasst werden [STRE99, S. 89-91]:

- Die Hälfte der im Rahmen der Untersuchung zu den verschiedenen Methoden und Werkzeugen befragten Unternehmen gab an, eine eigene Einführungsmethode für die Implementierung einzusetzen.
- Fast jedes dieser Unternehmen kreiert eine mehr oder weniger neuartige Einführungsmethode, die zwar einen neuen Namen hat, aber doch überwiegend auf bereits existierenden Methoden beruht.
- Diesen bereits vorhandenen Methoden ist gemeinsam, dass sie entweder nur „papier-basierte Lösungen“ sind oder aus einer Mischung verschiedener, für eine effiziente SAP-Einführung nur bedingt geeigneter Werkzeuge bestehen, die keinen direkten Zugriff auf ein SAP-System haben.
- Die einzelnen Methoden setzen sich überwiegend aus Bestandteilen anderer Methoden oder Werkzeuge zusammen.
- Ein Großteil der untersuchten Methoden wurde ähnlich wie bei LIVE METHOD/Chestra ursprünglich für den kompletten Prozess des geschäftlichen Wandels als durchgängiges Consulting-Konzept entwickelt und später speziell auf die SAP-Software angepasst.
- Die zu untersuchenden Einführungsmethoden und Werkzeuge entbehren meist einer werkzeuggestützten durchgängigen Vorgehensweise. Viele Lücken werden mit Berater-Know-how geschlossen.

SCHIPP hat ebenfalls verschiedene Vorgehensmodelle für die Einführung einer Standardanwendungssoftware untersucht [SCHI99, S. 95-111]. Der Fokus dieser Analyse lag dabei mehr auf der Abbildung vorkonfigurierter Lösungen. Neben typischen Vorgehensweisen für die SAP-Software, z. B. ARIS for mySAP.com der IDS Prof. Scheer AG [IDS00], waren auch Lösungen für sogenannte Non-SAP-Software, z. B. Application Implementation Methodology der Oracle Corporation [ORAC00], Gegenstand der Betrachtung.

Die beiden Untersuchungen zeigen, dass große Affinitäten zwischen den einzelnen Vorgehensmodellen bestehen. Die Bewertung bezüglich der in dieser Arbeit be-

handelten Problemstellung beschränkt sich daher im folgenden auf ASAP und LIVE METHOD/Chestra.

3.2 Beurteilung der Vorgehensmodelle

Ziel dieses Kapitels ist die Bewertung der vorgestellten Vorgehensmodelle anhand der in Tabelle 2-1 aufgeführten Kriterien. Hauptaugenmerk liegt dabei auf der Unterstützung der Wissensnutzung, deren Bedeutung in Kapitel 2.3 behandelt wird.

Tabelle 3-3: Beurteilung der Vorgehensmodelle

Kriterium	ASAP	LIVE METHOD/Chestra
Technik		
Offenheit	Das Vorgehensmodell erlaubt keine aktive Integration unterschiedlicher Werkzeuge. Anbindungen sind nur durch passive Schnittstellen möglich.	Die offene Architektur von Chestra lässt grundsätzlich die Verwendung alternativer Werkzeuge zu. So finden neben den LIVE Tools auch die Werkzeuge aus ASAP Verwendung.
Flexibilität	ASAP als Vorgehensmodell ist grundsätzlich nicht erweiterbar. Eine Adaption ist nur inhaltlich möglich.	Chestra als Vorgehensmodell ist nicht erweiterbar. Inhaltliche Erweiterungen sind über die Werkzeuge möglich.
Informationsverwaltung	Die Informationsverwaltung basiert überwiegend auf Einzeldokumenten, lediglich die Anforderungsspezifikation im Rahmen der Q&Adb wird datenbankbasiert verwaltet.	Nahezu alle Werkzeuge im Rahmen des Modells sind datenbankbasiert (z. B. LIVE KIT Structure, LIVE KIT Control). Daneben werden auch viele Einzeldokumente verwendet (z. B. Auswertungen, Vorlagen).
Softwareabhängigkeit	ASAP ist nur für die SAP-Software einsetzbar und speziell für diese Problemstellung zugeschnitten.	Chestra ist ein allgemein gültiges Methoden-Framework und kann nicht nur für die SAP-Software eingesetzt werden.
Systemintegration	ASAP ist in der vorliegenden Version 4.6C (Stand August 2000) an verschiedenen Stellen in das SAP-System integriert. Als Beispiel seien hier direkte Transaktionsaufrufe aus der BPML oder der Aufruf des Implementation Assistant aus dem SAP-System genannt.	Chestra selbst hat aufgrund der Softwareunabhängigkeit keine Integration zu einer speziellen Software. Die Werkzeuge mit ihren gewidmeten Inhalten haben an unterschiedlichen Stellen (z. B. Transaktionsaufrufe aus LIVE KIT Power, Berichtsanzeigen aus LIVE KIT Control) aktive Integrationsschnittstellen mit der SAP-Software.

Kriterium	ASAP	LIVE METHOD/Chestra
Versionsabhängigkeit	Die einzelnen Dokumente und die Referenzstruktur (d. h. die Prozesshierarchie) beziehen sich auf bestimmte Versionsstände der SAP-Software. Es werden technische Möglichkeiten zur Migration angeboten.	Das Vorgehensmodell ist releaseunabhängig. Die eingesetzten Werkzeuge sind releaseabhängig und werden bei jedem Wechsel angepasst. Es gibt technische Möglichkeiten der Datenübernahme.
Inhalt / Methodik		
Übersichtlichkeit	Die Werkzeuginhalte sind unzureichend aufeinander abgestimmt und erschweren die Übersichtlichkeit. Die heterogene Informationsverteilung macht viele Navigationsschritte nötig.	Die heterogenen Navigationsstrukturen (z. B. prozessorientiert, funktionsorientiert), die aus den unterschiedlichen Zielsetzungen der Werkzeuge resultieren, erschweren die Transparenz.
Nachvollziehbarkeit	Die Nachvollziehbarkeit einzelner Entscheidungen ist, bedingt durch die mangelnde Übersichtlichkeit, nur sehr rudimentär gegeben. Insbesondere lassen sich die Auswirkungen von Entscheidungen werkzeugübergreifend nur schwer analysieren.	Die einzelnen Werkzeuge der LIVE Familie besitzen aktive Integrations- und Navigationsschnittstellen, die eine ausreichende Nachvollziehbarkeit sicherstellen.
Stabilität	Die Informationsstrukturen, d. h. die Verzeichnis- und Dokumentenstruktur, bleiben weitestgehend stabil. Die Navigationsstruktur unterliegt aber je nach Releasestand verwirrenden Änderungen.	Die Strukturen bleiben weitestgehend stabil. Änderungen in den Navigationsstrukturen erschweren jedoch die Nutzung bereits gesammelter Informationen.
Vergleichbarkeit	Verschiedene Accelerators unterstützen einen Vergleich unterschiedlicher Projekte bzw. Projektstände, es gibt aber kein durchgängiges Konzept für die Gegenüberstellung im Sinne des CSE.	Einzelne Werkzeuge unterstützen direkte Projektvergleiche (z. B. LIVE KIT Structure), ein durchgängiges Konzept über alle eingesetzten Werkzeuge ist aber nicht realisiert.
Vollständigkeit	ASAP deckt nahezu alle Bereiche des Adaptionprozesses ab. Einige Schwachpunkte werden durch ergänzende Werkzeuge von Fremdanbietern ausgeglichen (z. B. Berichtsadaption).	Die Werkzeugvielfalt garantiert die vollständige Abdeckung der Projektabwicklung.

Kriterium	ASAP	LIVE METHOD/Chestra
Integration	Die ASAP-eigenen Werkzeuge sind integriert. Darüber hinaus gibt es passive Schnittstellen zu gängigen Softwareprodukten, die eine Projektabwicklung unterstützen (z. B. MS Project, MS Office). Werkzeuge aus anderen Vorgehensmodellen werden nicht aktiv unterstützt, die Offenheit der Werkzeuge ermöglicht aber einen passiven Einfluss auf einige Accelerators (z. B. Q&Adb, BMLS).	Die einzelnen Werkzeuge sind integriert und besitzen in der Regel bidirektionale Schnittstellen, die einen problemlosen Informationsfluss gewährleisten. Daneben werden durch aktive oder passive Interfaces auch Werkzeuge anderer Vorgehensmodelle unterstützt (z. B. ASAP).
Versionsabhängigkeit	Die abgelegten Informationen sind jeweils versionsabhängig. Für die Nutzung alter Informationen mit einem neueren Versionsstand sind meist Konvertierungsschritte nötig, Deltainformationen werden nicht angezeigt.	Die abgelegten Informationen sind versionsabhängig. Eine Übernahme alter Projektinhalte in eine neue Werkzeugversion ist möglich, dabei wird automatisch das Delta angezeigt (vgl. Fundamentalmmodell von LIVE KIT Power).

Tabelle 3-3 beschreibt die Einordnung der beiden vorgestellten Vorgehensmodelle ASAP und Chestra bezüglich der aufgestellten Kriterien. Die Ausführungen machen deutlich, dass die Modelle grundsätzlich für die Projektabwicklung im Adaptionsumfeld geeignet sind. Durch die stärker werdende Bedeutung der Folgeprojekte und der damit verbundenen Notwendigkeit effektiv das gesammelte Wissen aus den Projektierungen zu nutzen, sind jedoch Modifikationen der Modelle notwendig, die durch die THESEUS-Methode evaluiert werden sollen. Es gilt die Schwächen der bisherigen Ansätze zu identifizieren und die Stärken der Vorgehensweisen direkt in die Entwicklung von THESEUS einfließen zu lassen. Aus den Erkenntnissen der vorangegangenen Kapitel, insbesondere aus der Einordnung bezüglich der aufgestellten Kriterien, lassen sich die Schwächen wie folgt zusammenfassen:

- Die Informationen bezüglich der unterschiedlichen Themen, die im Zuge der Softwareadaption gewonnen werden, sind über die eingesetzten Werkzeuge verteilt, ein transparenter Zugriff ist nicht möglich.
- Die Navigationsstrukturen in den Werkzeugen sind sehr heterogen, die Transparenz wird dadurch zusätzlich erschwert. Insbesondere die Zusammenhänge der einzelnen Werkzeulemente werden nicht deutlich.

- Der Aufbau und die Inhalte der Werkzeugstrukturen variieren zum Teil bei Versionsänderungen erheblich, die Nutzung des bereits gesammelten Wissens wird somit behindert.
- Die einzelnen Werkzeuge sind sehr stark an einer bestimmten Softwarelösung ausgerichtet, die Berücksichtigung alternativer Produkte wird nicht ausreichend unterstützt.
- Die Vorgehensmodelle ermöglichen keine flexible Integration von Unterstützungswerkzeugen, insbesondere die Erweiterung des bestehenden Werkzeug-Pools ist nur mit erheblichem Entwicklungsaufwand möglich.
- Die Systemintegration der einzelnen Werkzeugen variiert sehr stark, ein durchgängiges Konzept ist nicht realisiert.
- Die Zusammenhänge zwischen den Systemeinstellungen und den Ergebnissen der Anforderungsanalysen sind durch die mangelnde Systemintegration nur schwer erkennbar.
- Adaptionaufwand, der durch veränderte Rahmenbedingungen initiiert wurde, kann mit den bisherigen Strukturen nur unzureichend identifiziert und realisiert werden.

Die genannten Schwächen können durch organisatorischen und technischen Zusatzaufwand umgangen werden. Im Hinblick auf eine effiziente und anwenderfreundliche Projektabwicklung im Sinne des Continuous System Engineering sollte es aber die oberste Maxime sein, durch methodische und inhaltliche Anpassungen bzw. Erweiterungen der Modelle auf die erkannten Schwächen zu reagieren sowie entsprechende Konzepte zu deren Beseitigung zu entwerfen.

3.3 Prinzipien für eine Adaptions-Workbench

Aus den in Kapitel 3.2 identifizierten Schwächen der bisherigen Vorgehensmodelle können Prinzipien für die Gestaltung einer Adaptions-Workbench abgeleitet werden, welche die Grundlage zur Entwicklung von THESEUS liefern. Ein Ergebnis der durchgeführten Bestandsaufnahme muss die Umsetzung in einem entsprechenden Konzept bzw. Werkzeug sein, das es dem Anwender ermöglicht, die spezifischen Informationen, die sich während des Lebenszyklus einer Softwarebibliothek ansammeln, gezielt zu nutzen, zu ändern und zu ergänzen. Nur so kann ein

wissensbasiertes Adaptionsumfeld geschaffen werden, das die Vorteile des Continuous System Engineering, insbesondere die Nutzung der Informationen über den Organisations- und Informationsverarbeitungsstatus, der durch den jeweils letzten Adaptionsprozess erreicht wurde [THHU96, S. 84], ausschöpfen kann. Ziel dieses Kapitels war es daher, methodische und inhaltliche Prinzipien festzuhalten und später im Rahmen der Konzeptausprägung zu nutzen. Die abgeleiteten Prinzipien sind:

Themenorientierung

Zu einem bestimmten Thema muss es möglich sein, aus verschiedenen Werkzeugen alle relevanten Informationen gebündelt darzustellen und zu bearbeiten. Eine unabhängige Adaption-Workbench kann eine solche Aufgabe übernehmen.

Übergeordnete Navigationsstruktur

Die aus der Werkzeugvielfalt resultierenden unterschiedlichen Navigationsstrukturen müssen durch eine übergeordnete Struktur zusammengefasst werden. So können einzelne Sachverhalte über mehrere Werkzeuge hinweg transparent gemacht werden.

Stabilität

Die Wissensträger müssen über den Zeitablauf, d. h. über den Lebenszyklus einer eingesetzten Softwarelösung konstant bleiben, um die Vergleichbarkeit und Transparenz zu gewährleisten. Vor allem Versionswechsel der eingesetzten Werkzeuge dürfen keinen Einfluss auf die Adaptionselemente haben. Über Zuordnungsmechanismen können die Inhalte von der Struktur getrennt werden.

Erweiterbarkeit

Die Navigationsumgebung muss flexibel erweiterbar sein, d. h. es muss möglich sein, neue Wissensträger zu definieren und mit den vorhandenen Elementen aus der Softwarebibliothek einerseits und den Adaptionswerkzeugen andererseits zu verknüpfen, ohne dass Informationsverluste auftreten oder bestehende Funktionalitäten nachhaltig beeinträchtigt werden.

Softwareunabhängigkeit

Die strenge Ausrichtung und Widmung der Adaptionsumgebung auf eine spezifische Softwarelösung muss aufgegeben werden. Über die Workbench sollten verschiedene Softwareprodukte adaptiert und alternativ analysiert werden können. Dazu ist ein hohes Maß an inhaltlicher und technischer Flexibilität nötig.

4 THESEUS-Methode

In diesem Kapitel wird die Entwicklung einer betriebswirtschaftlich gewidmeten Adaption-Workbench vorgestellt, welche die bestehenden Zwänge der Softwareabhängigkeit einzelner Vorgehensmodelle (Kapitel 3) aufhebt und als übergreifende Arbeitsumgebung zur Adaption unterschiedlicher Softwarelösungen dient. Von der Konzeption und Architektur bis zu den wichtigsten technischen Details werden alle Probleme angesprochen.

In Handwerksbetrieben bezeichnet der Begriff Workbench eine Werkbank, die als Arbeitsumgebung für den oder die Mitarbeiter dient. Mit Hilfe dieser Werkbank können die benötigten Materialien und Werkzeuge geordnet und strukturiert aufbewahrt sowie bei Bedarf bereitgestellt werden. Das im Rahmen dieser Arbeit entwickelte Konzept ermöglicht im übertragenen Sinn ebenfalls eine zielgerichtete und bedarfsgesteuerte Bereitstellung der notwendigen Materialien - in Form von Informationen - und Werkzeugen - in Form von Softwareprodukten - für die Realisierungsphase eines Adaptionprojekts. In Anlehnung an die Werkbank kann THESEUS daher als Adaption-Workbench bezeichnet werden.

Im Gegensatz zu den bisher eingesetzten Werkzeugen, die sich auf eine konkrete Softwarelösung beziehen und nur mit erheblichem Aufwand umgewidmet werden können, sollte eine Adaption-Workbench alternativ und/oder partiell mehrere Softwarebibliotheken ansprechen und deren Adaption unterstützen können. Gerade im Hinblick auf die neueren Entwicklungen im Bereich der Standardanwendungssoftware ist eine derartige Entkopplung und Flexibilität zwingend notwendig. Die zunehmende Öffnung des Marktes in den Bereichen Electronic Commerce (e-Commerce) und die daraus resultierenden Einsatzmöglichkeiten heterogener Softwareprodukte zur Abdeckung dieser Aufgabenfelder, in Verbindung mit der verstärkten Branchenausrichtung (z. B. Industrial Solutions der SAP AG) und Bereitstellung von Speziallösungen (z. B. New Dimension-Produkte der SAP AG) induziert eine gewisse Offenheit der eingesetzten Vorgehensmodelle zur Adaption solcher Softwarelösungen. Das derzeitige Instrumentarium kann diese Forderungen nur unzureichend erfüllen und muss durch entsprechende Zusatzwerkzeuge ergänzt werden. Der Schwerpunkt der Ausführungen und konzeptionellen Ent-

wicklung liegt auf der Erweiterung des ITHAKA-Konzepts, das bereits in Kapitel 2.1.3 vorgestellt wurde.

Die THESEUS-Methode wird zunächst schematisch erläutert, dann werden ihre Philosophie und Ziele formuliert (4.1). Dabei wird auf die Neuausrichtung der Adaptionenobjekte eingegangen. Im Anschluss daran werden die allgemeinen Gestaltungsprinzipien der Methode erläutert (4.2) und schließlich der Übergang zur Definition einer neuen Navigationsstruktur in Form von Wissenspaketen aufgezeigt (4.3), die zentraler Bestandteil des Fundamentalmodells (5) sind.

4.1 Präzisierung der Aufgabenstellung

Die Notwendigkeit einer Neukonzeption zur Unterstützung einer flexiblen Adaptionstrategie basiert nicht auf rein wissenschaftlichen Überlegungen, sondern wurde vielmehr aus Erkenntnissen von durchgeführten Projekten im Rahmen von Softwareeinführungen bzw. -anpassungen abgeleitet. Die Probleme, die in derartigen Projekten immer wieder zutage getreten sind, lassen sich in drei Hauptgruppen zusammenfassen:

- **Ansteuerung mehrerer Softwarelösungen:**
Für die unterschiedlichen Problemstellungen in der Geschäftstätigkeit eines Unternehmens werden verschiedene Softwarelösungen eingesetzt. Selbst mächtige Softwarebibliotheken, wie z. B. das SAP-System, können nicht alle Geschäftsfelder abdecken und versorgen. Die Entwicklung zusätzlicher Systeme, v. a. in den Bereichen des e-Commerce, bestätigen diese These. Die Vorgehensmodelle zur Adaption sind derzeit nicht in der Lage, mehrere verschiedene Softwarelösungen zu bedienen und eine zusammenhängende Projektierung systemübergreifend durchzuführen. Dabei sind die methodischen Grundlagen durchaus softwareneutral entwickelt worden, die Umsetzung beschränkte sich aber auf eine spezielle Softwarelösung.
- **Einsatz verschiedener Vorgehensmodelle und Werkzeuge:**
Für die Adaption von Softwarelösungen werden verschiedene Vorgehensmodelle angeboten, die jeweils Vor- und Nachteile gegenüber den „Konkurrenzprodukten“ aufweisen. Eine sinnvolle Kombination der Modelle und Werkzeuge könnte die Projektarbeit beschleunigen und die Arbeitsergebnisse verbessern. Die bisherigen Tendenzen zur Öffnung der Werkzeuge über

Schnittstellen, z. B. die Übernahme des Projektumfanges aus der ASAP-Q&Adb in das Individualmodell aus dem LIVE KIT Power, reichen nicht aus, um eine effiziente Nutzung von Synergien aus unterschiedlichen Modellen bzw. Werkzeugen zu unterstützen.

- Zugriff auf bereits gesammelte Informationen bei Folgeprojekten:
Die stetige Verbesserung der eingesetzten Softwarelösung und der damit verbunden Prozessabläufe ist nur möglich, wenn ausreichende Kenntnisse über den erreichten Stand der Informationsverarbeitung in einem Unternehmen vorliegen. Diese Informationen umfassen dabei sowohl die definierten Prozessschritte, als auch die vorgenommenen Systemeinstellungen. Mit dem derzeitigen Instrumentarium ist ein transparenter Zugriff und eine flexible Kombination der vorliegenden Informationen nicht in ausreichendem Maß möglich. Ein Grund ist die mangelnde Integration der Adaptionen über die Grenzen der einzelnen Modelle bzw. Werkzeuge hinweg.

Grundsätzlich lassen sich die genannten Probleme über die Bereitstellung einer offenen, flexiblen Navigationsumgebung lösen, die eine Mixtur aus Softwarelösungen und Vorgehensmodellen zulässt. THESEUS soll die methodischen Grundlagen für die Entwicklung einer derartigen Umgebung schaffen.

Ähnlich dem griechischen Helden THESEUS, der von Ariadne ein Wollknäuel bekam, um so den Weg aus dem Labyrinth des Minotaurus bestimmen zu können, soll die hier vorgestellte Methodik einen transparenten Weg durch das umfassende aber teilweise verwirrende Wissen finden. Aufgrund dieser Affinität wurde für die entwickelte Vorgehensweise das Akronym THESEUS gewählt.

Zunächst sollen noch einmal die grundlegenden Überlegungen erläutert werden, die zur Entwicklung von THESEUS geführt haben.

Die derzeit bestehenden Ansätze sind nur bedingt geeignet, eine konsistente und im Sinne des CSE dynamische Verwaltung der Informationen aus dem Umfeld der Softwareadaption zu gewährleisten. Ein Problem dabei ist die mangelnde Stabilität der Navigationsstrukturen bzw. der einzelnen Wissensträger, z. B. eines Geschäftsprozesses. Ständige Änderungen in den verwendeten Referenzmodellen erschweren den Aufbau einer Wissensbasis und verhindern die effiziente Nutzung des gesammelten Wissens. Ein weiteres Problem ist die facettenreiche Werkzeuglandschaft, die ein Ergebnis der unterschiedlichen Zielsetzungen einzelner Adaptionen-

prozesse ist. Jedes Analysewerkzeug beinhaltet eine eigene Navigationsstruktur, wobei die Unterscheidungen aufgrund der zu behandelnden Fragestellungen durchaus berechtigt sind. Die Ausführungen zum Wissensmanagement machen deutlich, dass diese Heterogenität die Transparenz für den Anwender erschwert und Verständnisprobleme hervorruft. Es sei an dieser Stelle nochmals darauf hingewiesen, dass der Einsatz von unterschiedlichen Werkzeugen in den jeweiligen Projektphasen nicht in Frage gestellt, sondern ausdrücklich befürwortet wird. THESEUS bündelt das umfassende Wissen der Adaption und ermöglicht dem Anwender bzw. Projektteam die effiziente Nutzung und Weiterverarbeitung der Informationen. Das Ziel ist es daher, eine Sammlung von Adaptionselementen zu entwerfen, die gleichermaßen die Transparenz des erzeugten Wissens sicherstellt und darüber hinaus die von THOME und HUGGARD postulierte Phase 2 der CSE-Vorgehensweise unterstützt [THHU96, S. 88]. Das Überdenken und Erneuern von Geschäftsprozessen in kleinen Schritten wird erst durch den Zugriff auf alle relevanten Informationen der Eröffnungslösung möglich, dies zu unterstützen ist Aufgabe der hier entwickelten Konzeption. Vor allem die Aktualität der Dokumentation des „durch den jeweils letzten Adaptionsprozess erreichten Organisations- und Informationsniveaus“ [THHU96, S. 88], wird durch den hier entwickelten Ansatz gewährleistet, nur so kann eine geeignete Ausgangsbasis für die gesicherten Überlegungen zur Weiterverbesserung der Geschäftsprozesse geschaffen werden. Neben den Erkenntnissen aus dem Wissensmanagement (vgl. Kapitel 2.3) und insbesondere aus dem Hyperknowledge-Konzept (vgl. Kapitel 2.3.2.3) spielt die Entwicklung zur hypermedialgestützten Bearbeitung eine wichtige Rolle. Informationen müssen zeitnah zum Bedarf des Anwenders ermittelt werden und entsprechend der Sichtweise des Betrachters aufgebaut sein, d. h. nur die relevanten Daten werden zu einem Sachverhalt aufbereitet und zur Verfügung gestellt. Die Funktionen, Prozesse oder Organisationsstrukturen können dabei sinnvolle Informationsbasen für die Entscheidungssichten sein. Jede Sicht verlangt eine eigene flexible Aufbereitung und Bereitstellung der Informationen, was durch die bisherigen Ansätze nicht ausreichend unterstützt wird.

Die Bildung von inhaltlich und organisatorisch stabil abgrenzbaren Informationseinheiten (Wissenspakete) ermöglicht die geforderte Flexibilität. Die in sich stabile Einheit kann leicht gepflegt und flexibel eingesetzt werden. Änderungen zu einem Paket werden nur einmal in der zentralen Zuordnungsstruktur vorgenommen und

stehen an allen Einsatzgebieten zeitnah zur Verfügung. Somit werden alle Bereiche, die auf dieses Element zugreifen, direkt auf den aktuellen Stand gebracht.

Der hier entwickelte Ansatz soll dabei nicht die bestehenden Vorgehensmodelle oder Adaptionensätze ablösen oder überflüssig machen, vielmehr liegt der Fokus auf der Realisierung einer anwenderfreundlichen und im Sinne des CSE dynamischen Adaptionunterstützung, die auf bestehende Inhalte zurückgreift und diese in Abhängigkeit der bearbeiteten Problemstellung aufbereitet. Die Sammlung, Aufbereitung und datenbankbasierte Dokumentation von Wissen aus dem Umfeld der Softwareadaption sind somit die Hauptaspekte der hier behandelten Zielsetzung.

4.1.1 Umsetzungsstrategien

Für das bessere Verständnis der THESEUS-Methode und der in Kapitel 4.3 definierten Wissensträger ist es erforderlich, dass zunächst die grundsätzlichen Strategien zur Umsetzung von Adaptionsergebnissen in einer Standardanwendungssoftware unterschieden werden. Die klassischen Ansätze gehen von einer Trennung zwischen der funktions- bzw. modulatorientierten und der prozessorientierten Umsetzung aus. Diese Unterscheidung ist für die komplexe Problemstellung, die das Adaptionsumfeld und die damit verbundenen Analyseschritte bieten, nicht ausreichend. Für THESEUS wird deshalb auf den Grundgedanken des Mind Mapping, die Konzentration auf eine zentrale Idee, zurückgegriffen und eine Themenorientierung auf Basis von Wissenspaketen propagiert.

MODULATORIENTIERTE UMSETZUNG

Moderne Softwarebibliotheken sind aus betriebswirtschaftlicher Sicht modular aufgebaut, d. h. die Bibliothek ist in thematisch zusammengehörige Komponenten aufgeteilt, die jeweils einen eigenen funktionalen Bereich innerhalb eines Unternehmens abbilden. Die SAP AG entwickelte beispielsweise fünf Hauptkomponenten, namentlich:

- die drei funktionsbezogenen Komponenten Accounting, Human Resources und Logistics,
- die funktionsübergreifende Komponente Cross Application Component und

- die Systemkomponente Basic Component.

Diese Hauptkomponenten wurden gemäß den funktional orientierten Unternehmensbereichen noch weiter in die sogenannten Module unterteilt (z. B. Vertrieb, Materialwirtschaft, Produktion). Komplexe Unternehmensabläufe durchlaufen dabei jeweils einen Teilbereich aus unterschiedlichen Modulen, wobei ein integrierter Datenfluss durch den ganzheitlichen Systemansatz gegeben ist. Die vertikale Ausprägung der Applikationen wird durch horizontal liegende und integriert ablaufende Prozesse verbunden.

Modul- bzw. funktionsorientierte Ansätze gehen bei den Implementierungsbemühungen analog der im Unternehmen bestehenden Struktur vor. Die meisten Unternehmen sind funktionsorientiert, d. h. nach funktionalen Betriebsbereichen, z. B. Personalwesen, Entwicklung, Materialwirtschaft, aufgeteilt [WÖHE00, S. 178]. Gemäß der Maxime des CSE werden die Unternehmen diese Orientierung im Zuge der Einführung einer Softwarebibliothek nicht ändern, sondern die bestehenden Potenziale der Software nutzen und die im Rahmen des Betriebszwecks notwendigen Hauptfunktionen herausarbeiten [THHU96, S. 82].

Die Implementierung einer Softwarebibliothek nach funktionsorientierten Kriterien lässt die vielfältigen Möglichkeiten zur Unterstützung der unternehmensweit ablaufenden Prozesse weitgehend ungenutzt. Analog zur Aufteilung nach Adam Smith werden auch hier nur kleine Teileinheiten bzw. -funktionen umgesetzt, das Gesamtbild der Geschäftsprozesse geht dabei verloren [TEUF99, S. 26].

PROZESSORIENTIERTE UMSETZUNG

Prozessorientierte Ansätze knüpfen die Softwarebibliothek an die applikationsübergreifenden Geschäftsabläufe. Folglich werden alle Prozessbausteine und Funktionen aus den einzelnen Applikationen (=Modulen) eingeführt, die zur Abwicklung des Prozesses benötigt werden. Diese Vorgehensweise berücksichtigt also die Anforderungen aus Organisations- und Prozesssicht bezüglich einer durchgängigen und gesamtheitlichen Geschäftsprozessunterstützung und -verbesserung. Durch dieses Konzept können aufwendige Schnittstellenanalysen größtenteils vermieden werden. Im Sinne des CSE wird auch bei dieser Art der Adaptionstrategie der Fokus zunächst auf die wichtigsten Kernprozesse oder zumindest auf die in einem ersten Schritt zwingend notwendigen Funktionalitäten gelegt. Somit können beispielsweise Probleme hinsichtlich der Komplexität der zu erarbeitenden Lösung,

fehlender interner oder externer Ressourcen oder des hohen Abstimmungs- bzw. Koordinationsaufwands umgangen werden. Das Vorgehensmodell ASAP ist ein Beispiel für die prozessorientierte Umsetzungsstrategie (vgl. Kapitel 3.1.1). Aus pragmatischen Gründen wird in der Praxis häufig eine disjunkte Betrachtung der einzelnen Schritte in der Vordergrund gerückt, obwohl diese hochgradig vernetzt sind und Wechselwirkungen haben. Somit wird die hinter dem Prozessgedanken stehende Vernetztheit nur unzureichend betrachtet. Als Unterstützungswerkzeuge werden hierbei vorwiegend gewidmete Referenzmodelle eingesetzt, die den Funktionsumfang der betrachteten Softwarelösung widerspiegeln.

THEMENORIENTIERTE UMSETZUNGSSTRATEGIE

Die themenorientierte Strategie löst sich von der strengen Ausrichtung an den Geschäftsprozessen eines Unternehmens, baut jedoch auf den Erkenntnissen dieser Umsetzungsstrategie auf und verbindet sie mit den Einflüssen aus den Bereichen der Objektorientierung und des Mind Mapping.

Vorgehensmodelle zur Einführung bzw. Adaption von Softwarebibliotheken weisen eine hohe Komplexität auf, da sie die verschiedenen Adaptionsschritte miteinander koppeln. Besonders die Verknüpfung der zielgerichteten und daher heterogenen Werkzeuge stellt in Verbindung mit der geforderten Effizienz und Transparenz der Einführungsaktivitäten eine große Herausforderung dar, die durch funktions- bzw. prozessorientierte Vorgehensweisen nicht adäquat angenommen werden kann. Ein komplexes betriebswirtschaftliches Themengebiet kann nicht vollständig durch eine funktionsorientierte Analyse bearbeitet werden, da wichtige integrative Prozessinformationen fehlen. Auf der anderen Seite ist eine Prozessanalyse ohne bereichsspezifische Anforderungen nicht ausreichend. Mit dem derzeitigen Instrumentarium ist eine Mischstrategie, wie die hier entwickelte Themenorientierung, nicht möglich. Eine Betrachtung der funktional ermittelten Ergebnisse eines LIVE KIT Structure-Workshops beispielsweise macht in Bezug auf einen Kernprozess aus dem Umfeld des LIVE KIT Power wenig Sinn, da kein direkter Zusammenhang zwischen den beiden Zielelementen besteht. Im umgekehrten Fall ist es wenig aussagekräftig zu einer bestimmten Reduktionsfrage auf Ergebnisse der Kernprozessmodellierung zurückzugreifen, da beide Ansätze völlig unterschiedliche Zielsetzungen verfolgen. Die strenge Ausrichtung auf eine prozessorientierte Vorgehensweise, wie sie in ASAP umgesetzt wurde, schränkt zum einen den Einsatz möglicher Werkzeuge ein, zum anderen verringert sich der

möglicher Werkzeuge ein, zum anderen verringert sich der Aussagegehalt der gewonnenen Ergebnisse durch die fehlenden Zusammenhänge.

Ein weiterer Nachteil der bestehenden Ansätze ist die Abbildung von softwareübergreifenden Themen. Gerade im Hinblick auf die hohe Integrationsdichte der klassischen (ERP-) Systeme und der neuen e-Business-Lösungen können betriebswirtschaftliche Sachverhalte identifiziert werden, bei deren Umsetzung mehrere Softwarelösungen zu bedienen sind. Ein Beispiel soll dies verdeutlichen. Die Kooperation des ERP-Marktführers SAP AG mit dem Hersteller für e-Business-Lösungen Clarify induziert bei der Umsetzung eines integrierten CRM den Einsatz unterschiedlicher Softwareprodukte, namentlich R/3 der SAP AG als Back Office-System und das Produkt eSupport der Firma Clarify zur Abwicklung des webbasierten Kundenservice. Beide Programme tauschen über standardisierte Schnittstellen Daten aus. Die Implementierung der eSupport-Lösung macht bezüglich des Kundenservice Einstellungen im R/3-System notwendig und umgekehrt. Die prozessorientierte Umsetzungsstrategie setzt auf gewidmeten Referenzmodellen, z. B. für die Prozessgestaltung des R/3-Systems, auf und kann die Abläufe der integrierten Front Office-Software nicht abbilden. Ebenso verhält es sich mit anderen Adaptionswerkzeugen (z. B. LIVE KIT Structure), die nicht in der Lage sind, derartige übergreifende Sachverhalte zu berücksichtigen. Für die Umsetzung ist in dieser Konstellation der Einsatz unterschiedlicher Werkzeuge notwendig, was einen erheblichen Transparenzverlust zu Folge hat.

Bei der Themenorientierung wird die Trennung der verschiedenen Sichtweisen aufgehoben und ein zentrales Thema identifiziert, das mit allen relevanten Methoden sowie allen Werkzeugen aus verschiedenen Vorgehensmodellen verknüpft ist und als Referenzobjekt sämtliche Informationen zur Umsetzung des Themas in der unternehmensweiten Softwarelandschaft sammelt. In Verbindung mit den Erkenntnissen aus der Objektorientierung ergibt sich somit ein völlig neuer Ansatz der Softwareimplementierung.

Nach dem Paradigma der Objektorientierung wählt der Anwender aus einer Fülle von Objekten, die sich auf eine oder mehrere Softwarelösungen beziehen kann, diejenigen aus, die er für seine Aufgabenstellung benötigt. Die einzelnen Objekte sind gekapselt, d. h. sie beinhalten alle für diesen Typ relevanten Informationen. Die Gegenüberstellung der Prozess- und/oder Objektmodelle zur Beschreibung einer Standardanwendungssoftware zeigt, dass Prozesse im Grunde nichts anderes

sind, als eine Navigation durch bzw. Komposition von Objekten, die über Methoden und Schnittstellen miteinander kommunizieren.

Übertragen auf die Adaptionsaktivitäten eines Unternehmens bedeutet dies, dass der Anwender bei der Umsetzung der Ergebnisse nicht an die Zwänge vorgegebener Strukturen, wie sie in den bisherigen Vorgehensweisen zu finden sind, gebunden ist, sondern vielmehr dynamisch eine Projektstruktur zusammenstellen kann, die flexibel einsetz- und beliebig erweiterbar ist.

4.1.2 Stufenmodell

Die Intention von THESEUS ist die Entwicklung einer adaptierbaren Realisierungs- und Analysekomponente, um gezielt Informationen aus den Bereichen der Anforderungsanalyse, Prozess- und Berichtsmodellierung bzw. der Organisationsgestaltung für ein betriebswirtschaftliches Thema zu bündeln und die Umsetzung in einer Softwarelösung zu unterstützen. Das Konzept gliedert sich in mehrere Stufen, die iterativ durchlaufen werden und jeweils zum Ausbau der Wissensbasis beitragen. Im einzelnen besteht das Modell aus folgenden Stufen (Abbildung 4-1):

1. Definition des Projekts und der Wissenspakete:

Die Realisierungsaktivitäten werden in einem eigenständigen Projekt zusammengefasst. Es wird bestimmt, aufgrund welcher Informationsquellen die Umsetzung der Ergebnisse vorgenommen wird (z. B. Anforderungsanalyse mit LIVE KIT Structure, Umfang des ASAP-Projekts). Weiterhin wird das Fundamentalmodell um spezielle, noch nicht vorhandene Wissenspakete erweitert, sofern dies notwendig ist.

2. Identifikation der relevanten Wissenspakete:

Die zur Verfügung stehenden Informationsquellen werden als Grundlage für die Identifikation der zur Auswahl stehenden Pakete herangezogen. Somit wird der Realisierungsrahmen eingegrenzt und vorbereitet. Es ist auch denkbar, dass die Realisierung ohne den Bezug zu irgendwelchen Informationsquellen durchgeführt werden soll oder dass zu einem bestimmten Thema (z. B. e-Commerce) keine Informationsquellen vorliegen. In diesem Fall werden aus der bestehenden Sammlung die gewünschten Pakete manuell ausgewählt und in den Projektumfang übernommen.

3. Komposition der Projektstruktur:

Aus den verbliebenen Wissenspaketen wird über eine beliebige Hierarchisierung eine kundenindividuelle Projektstruktur aufgebaut. Auf unterster Ebene werden die einzelnen Wissenspakete eingegliedert, die als eigentliche Navigationselemente für die weiteren Schritte dienen. Diese Komposition ist der fundamentale Unterschied zu den bestehenden Werkzeugen und ermöglicht eine flexible und offene Gestaltung der Projektabwicklung in dieser späten Phase der Softwareadaption. Bisher war es lediglich möglich, eine bestehende Struktur (z. B. Prozesshierarchie) auf die notwendigen Elemente zu reduzieren, eine Änderung der Strukturierung war nicht möglich. Für die Definition kann auf bestehende Referenzstrukturen zurückgegriffen werden, die bereits ausgeprägte Hierarchiestufen beinhalten und an die unternehmensindividuellen Anforderungen anpassbar sind. Beispiele für derartige Referenzstrukturen sind die Unterscheidung nach den eingesetzten Systemen (SAP R/3, SAP-CRM, Siebel, Oracle etc.), Organisationseinheiten (Standorte, Werke, Buchungskreise etc.) oder Geschäftsprozessen (Gerätewesen, Auftragsabwicklung etc.).

4. Ermittlung des Projekt-Aufwands:

Der Aufwand zur Umsetzung der selektierten Wissenspakete wird unter Zuhilfenahme der ermittelten Anforderungsergebnisse (z. B. aus dem LIVE KIT Structure) für jedes Paket ermittelt und dient in Summe als Grundlage für die Projektkalkulation.

5. Konfiguration des Systems und Dokumentation der Einstellungen:

Das System wird auf Basis der festgelegten Projektinstanzen mit den zugeordneten Wissenspaketen konfiguriert und eingestellt. Die vorgenommene Parametrisierung bzw. die gepflegten Tabelleneinträge werden mit direktem Bezug zu einem Wissenspaket dokumentiert. An dieser Stelle zeigt sich die hohe Flexibilität des hier entwickelten Ansatzes. Während die bestehenden Werkzeuge durch den Softwarebezug jeweils nur eine Möglichkeit der Systemkonfiguration unterstützen konnten, ist es nun möglich, verschiedene Einstellungswerkzeuge anzusprechen. Ein Beispiel soll diesen Vorteil verdeutlichen. Für das SAP-System wird das Customizing mit Hilfe des Einführungsleitfadens durchgeführt. Die Werkzeuge zur Unterstützung der Realisierung hatten daher immer einen direkten Bezug zu diesem Einführungs-

leitfaden. Die Berücksichtigung anderer Softwarelösungen (z. B. Intershop der INTERSHOP Software Entwicklungs GmbH) war nicht möglich, da dieses System nicht über einen IMG parametrisierbar ist, sondern über die sogenannten Pipelines auf die Anwenderbedürfnisse abgestimmt wird [INTE00]. Bei einem parallelen Einsatz der beiden Softwarelösungen mussten somit auch unterschiedliche Werkzeuge in der Realisierungsphase verwendet werden, die wiederum nicht integriert waren. THESEUS ermöglicht über die softwareunabhängigen Wissenspakete im Zusammenspiel mit einem offenen Zuordnungsmechanismus die Anbindung beliebiger Softwarebibliotheken bzw. Customizingwerkzeuge innerhalb der homogenen Adaptions-Workbench.

6. Abnahme des erstellten Systems:

Nach der Fertigstellung des Systems und Beendigung verschiedener Test-szenarien schließt die Abnahme des Systems den Adaptionsprozess ab. Für die Ausprägungen der Wissenspakete bedeutet dieser Schritt, dass keine Änderungen an den Paketen mehr vorgenommen werden und somit der aktuelle Stand als eigenständige Version „eingefroren“ wird. Das aufgebaute Wissen aus Paketstruktur und -inhalt dient als Ausgangspunkt und Grundlage für die nachfolgenden Projekte, die sich mit der weiteren Anpassung bzw. Weiterentwicklung des Systems beschäftigen.

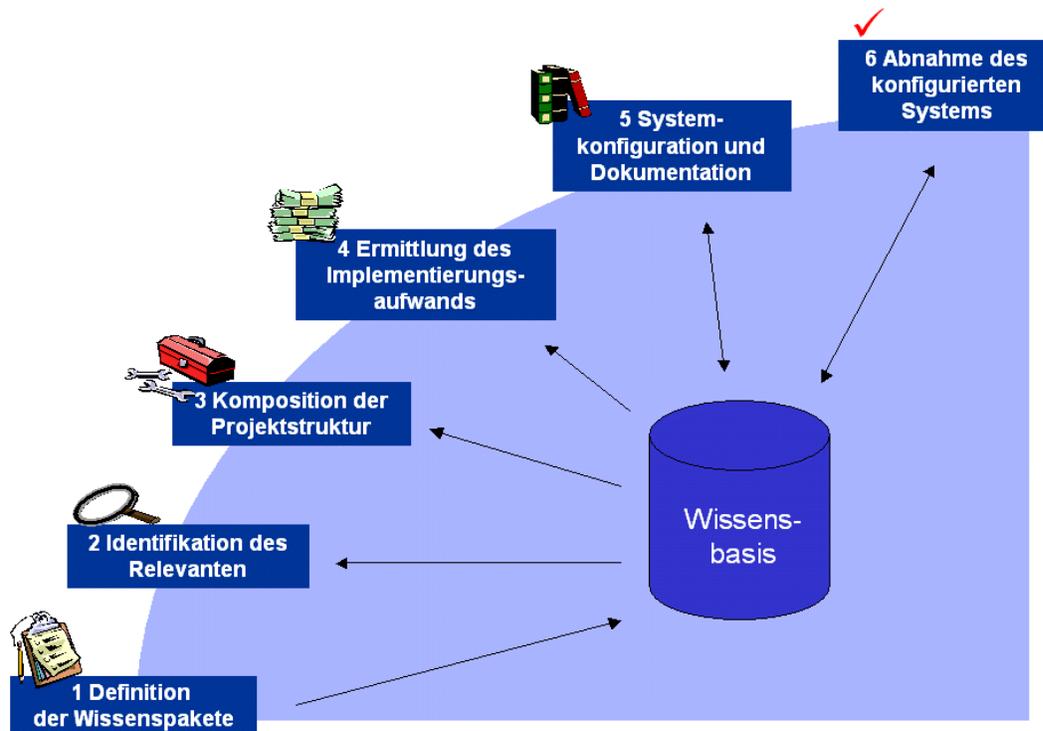


Abbildung 4-1: Stufenkonzept THESEUS

Diese Phasen sind elementarer Bestandteil der THESEUS-Methode. Für die Unterstützung des Continuous System Engineering werden diese Stufen wiederholt durchlaufen, wobei jeweils auf die Informationen aus der Wissensbasis zurückgegriffen wird.

4.1.3 Architektur

Die Abwicklung der Realisierungsphase, wie sie in Kapitel 4.1.2 beschrieben wurde, kann nur unterstützt werden, wenn die einzelnen Komponenten des Konzepts voneinander getrennt sind. Als Komponenten werden in diesem Zusammenhang

- die Informationsquellen,
- der Zuordnungsmechanismus und
- die operative Projektierungsumgebung

unterschieden. Diese Separation führt zu einer dreistufigen Architektur, die im folgenden näher betrachtet wird.

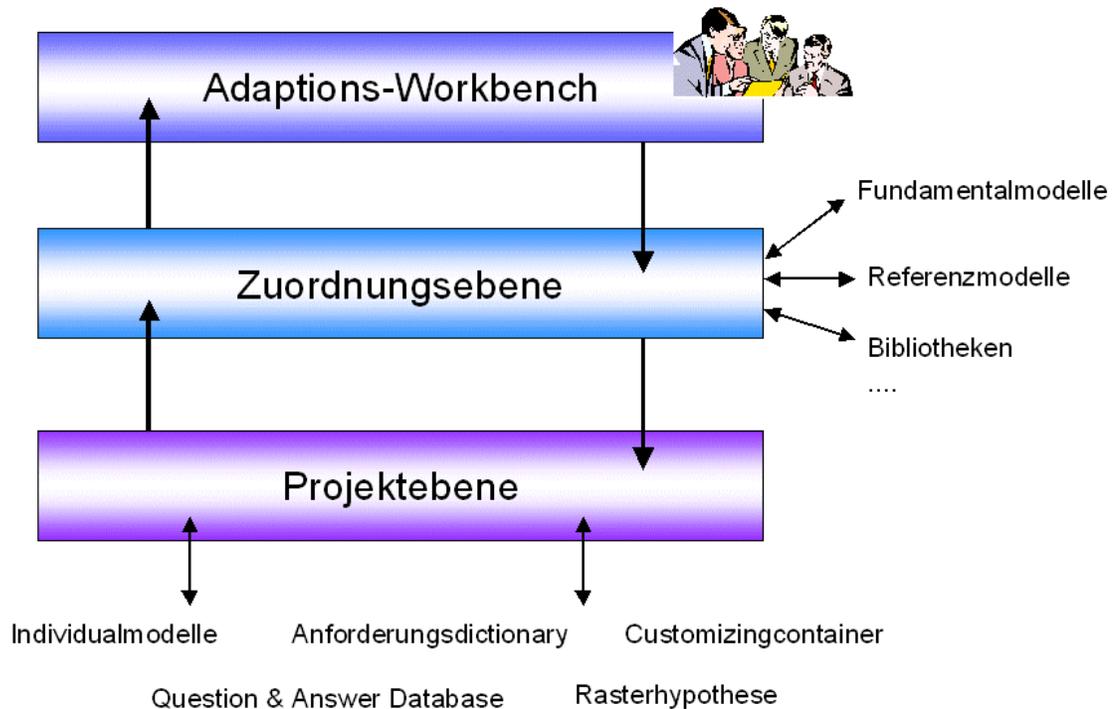


Abbildung 4-2: THESEUS-Architektur

Abbildung 4-2 zeigt die unterschiedlichen Ebenen des THESEUS-Modells. Es wird zwischen der Projekt- und Zuordnungsebene sowie der eigentlichen Adaptionen-Workbench unterschieden.

PROJEKTEBENE

Auf Projektebene werden alle Informationen gespeichert, die in direktem Zusammenhang mit dem jeweiligen Adaptionenvorhaben stehen. Über Schnittstellen wird der Bezug zu den eingehenden Datenquellen aus den unterschiedlichen Werkzeugen (z. B. LIVE KIT Power) hergestellt und die erforderlichen Informationen werden abgerufen. Es erfolgt keine redundante Ablage der Analyseergebnisse. Die vorgenommenen Systemeinstellungen werden ebenfalls in der Projektebene mit eindeutigem Bezug zu den Konfigurationstransaktionen dokumentiert. In der Projektebene werden somit alle relevanten Informationen bezüglich eines Projekts gespeichert.

ZUORDNUNGSEBENE

Die Zuordnungsebene ist der zentrale Bestandteil von THESEUS. Sie beinhaltet alle Verknüpfungen, die zwischen den Wissenspaketen und den eingesetzten bzw.

betrachteten Werkzeugen bestehen. Die Zuordnungen werden über reine Assoziierungstabellen gepflegt, d. h. es erfolgt keine redundante Ablage der Informationen aus den eingehenden Datenquellen. Gerade in dieser Ebene ist eine offene und flexible Datenstruktur notwendig, um die geforderte Unabhängigkeit der Workbench zu gewährleisten. Die zu integrierenden Werkzeuge müssen verschiedene Voraussetzungen erfüllen, um den Einsatz von THESEUS zu ermöglichen.

1. Die Datenspeicherung des Werkzeugs muss in strukturierter bzw. formatierter Form erfolgen (vgl. Kapitel 2.3.2.2), um die Selektion und Aufbereitung entscheidungsrelevanter Informationen sicherzustellen. Diese Voraussetzung ist durch Datenbanken in der Regel gewährleistet, es sind aber auch andere Formen der strukturierten Ablage, z. B. Textdateien mit einem fest definierten Datensatzaufbau, möglich.
2. Auf die Datenbasis muss ein Zugriff über standardisierte Schnittstellen möglich sein. Die datenbankunabhängige Technologie ODBC (Open Database Connectivity) beispielsweise erfüllt diese Forderung. ODBC ist eine Programmierschnittstelle, mit der Anwendungen auf Daten in Datenbankmanagementsystemen zugreifen können, die SQL (Structured Query Language) als Standardsprache für den Datenzugriff unterstützen [HILL97, S. 50].
3. Die einzelnen Elemente aus der Datenbasis des Werkzeugs müssen eindeutig identifizier- und erreichbar sein, um die Zuordnung der Objekte zu den jeweiligen Wissenspaketen zu unterstützen. Datenbankbasierte Werkzeuge erfüllen diese Voraussetzung durch die eindeutige Vergabe von Primärschlüsseln.
4. Das Werkzeug muss in der Lage sein, die gesammelten Informationen in Abhängigkeit eines Projekts zu speichern und im Bedarfsfall abzurufen. Diese projektspezifische Ablage der Elementausprägungen ist für die Festlegung der Anforderungen bezüglich eines Wissenspakets von essentieller Bedeutung.

ADAPTIONS-WORKBENCH

Diese Ebene regelt die eigentliche Interaktion zwischen Anwender und der Projektdatenbank. Das Front-End übernimmt alle Navigations- und Manipulationsschritte, die zur Bearbeitung der behandelten Problemstellung notwendig sind. Die

Aufgabenschwerpunkte in dieser Ebene liegen auf einer transparenten Bereitstellung des gesammelten Wissens und der übersichtlichen Darstellung der angeforderten Informationen.

4.2 Modellgestaltung

THESEUS ermöglicht als Monitoring-Ansatz durch die verschiedenen Detaillierungsebenen dem Anwender einen schnellen Überblick über alle relevanten Aspekte und Informationen zu einem bestimmten Sachverhalt, ohne eine Beschränkung der verwendeten Methoden oder Werkzeuge. Über die unabhängige und offene Arbeitsumgebung kann der Anwender verschiedene Softwarelösungen in die Implementierungsschritte mit einbeziehen und so die vorgefundene Systemlandschaft gemäß den von HUFGARD entwickelten Adaptionierungsgrundsätzen [HUF94, S. 115-116] entsprechend den Anforderungen gestalten.

In die Gestaltungsprinzipien fließen die in Kapitel 2 dargestellten Grundlagen und die aus dem Vergleich der beiden Vorgehensmodelle LIVE Tools/Chestra und ASAP gewonnenen Erkenntnisse (Kapitel 3.3) ein.

4.2.1 Modellintention und Modellierungsansatz

THESEUS ist mit dem Ziel der Unterstützung der schnellen und kostengünstigen Realisierung der Analyseergebnisse entwickelt worden. Anregungen wurden von den beschriebenen Grundlagen, v. a. aus den relativ jungen Bereichen des Wissensmanagements sowie des Hyperknowledge, übernommen und unter dem Gesichtspunkt der Vermeidung von Schwächen bestehender Ansätze eingesetzt.

Mit THESEUS werden adaptionsspezifische Informationen mit Hilfe der Wissenspaketstruktur gebündelt veranschaulicht und durch Dokumentationsmöglichkeiten ergänzt. Die Integration wird durch den direkten Zugriff auf das Umsetzungswerkzeug der Softwarelösung realisiert.

Folgende Zielsetzungen zeichnen THESEUS als wesentliche Erweiterung des ITHAKA-Konzepts aus:

- Visualisierung aller adaptionrelevanten Informationen
Die gebündelte Darstellung aller für die Umsetzung eines betriebswirtschaftlichen Themas relevanten Teilbereiche aus der Analyse- bzw. Modellie-

rungsphase gewährleistet eine konsistente und schnelle Konfiguration des Systems. Im Gegensatz zu den bestehenden Ansätzen werden aufwendige Dokumentationsrecherchen über mehrere Werkzeuge hinweg vermieden.

- **Systematische Unterstützung der Systemkonfiguration**
Die Ausrichtung an gekapselten Wissenspaketen ermöglicht es, systematisch die Parametereinstellung bzw. Tabelleneinträge im Zielsystem vorzunehmen, wobei die Vollständigkeit durch die Themenorientierung sichergestellt wird. Bereits bearbeitete Pakete können als Vorlage für ähnliche Aufgabenstellungen dienen, somit kann gezielt auf Wissen zurückgegriffen werden, das in vorangegangenen Projekten abgelegt wurde.
- **Flexibilität in der Projektgestaltung**
Die Komposition der Projektstruktur für den Teilbereich der Realisierungsphase schafft Freiraum für die individuelle Gestaltung der Tätigkeitsabfolge. Die strenge Ausrichtung an vorgegebenen Strukturen, wie sie in anderen Modellen zu finden ist, wird durch die generischen Wissenspakete obsolet, da alle Abhängigkeiten in einem Paket berücksichtigt sind und daher eine freie Zusammenstellung einzelner Aufgabenblöcke für die Umsetzung ermöglicht wird. Dem Anwender eröffnet sich so die Möglichkeit, seinen organisatorischen und/oder inhaltlichen Anforderungen an die Projektierungsabfolge gerecht zu werden, ohne einen Informationsverlust hinnehmen zu müssen.
- **Software- und Werkzeugoffenheit**
Die Trennung der einzelnen Ebenen lässt die Zuordnung verschiedener Softwarelösungen auf der einen und unterschiedlicher Adaptionswerkzeuge auf der anderen Seite offen. Somit können neue thematische Ausrichtungen des Unternehmens, z. B. der Einsatz integrierter e-Commerce-Lösungen, miteinbezogen und dadurch die Wissensbasis um genau diese Informationen erweitert werden. Zudem wird es möglich, die Schwächen einzelner Ansätze für die Unterstützung der Softwareadaption durch die Anbindung besserer Werkzeuge zu umgehen und so das Qualitätsniveau der Adaption zu erhöhen.

4.2.2 Softwarebezug

Das Ebenenmodell von THESEUS ist software- und vorgehensmodell-neutral konzipiert worden, d. h. es wurde bei der Entwicklung der Prinzipien bewusst darauf geachtet, die hier vorgestellte Methode ohne spezifische Widmung zu einer Softwarebibliothek auszuprägen. Da THESEUS aber vornehmlich das Ziel hat, das ITHAKA-Konzept um eine integrierte, aber gleichzeitig offene Realisierungskomponente zu erweitern, wurde das Fundamentalmodell mit der SAP-Softwarebibliothek verknüpft und stellt aus diesem Grund ein Abbild dieser Lösung dar. Die Zuordnungen bezüglich der Wissenspakete präsentieren in den einzelnen Monitoren jeweils konkrete adaptionsrelevante Aspekte der Softwarebibliothek.

Zudem greift THESEUS auf die Ergebnisse zeitlich vorgeschalteter Analyse- bzw. Modellierungsschritte zurück. Diese als Informationsquellen bezeichneten Werkzeuge sind ebenfalls im Fundamentalmodell berücksichtigt, so dass auch hier eine Verknüpfung besteht. Im einzelnen werden folgende Methoden betrachtet

- ODYSSEUS,
- PENELOPE und
- MENTOR.

Neben diesen Konzepten sind zudem OLYMP [BÄTZ99], MEDEA [MEHL98], SPARTA [SCHI99] und OKEANOS [WALZ00] in der Zuordnungslogik des Fundamentalmodells enthalten.

4.2.3 Darstellung

Im Rahmen des hier entwickelten Ansatzes wird ein mehrstufiges Konzept verfolgt, das auf den verschiedenen Ebenen jeweils einen unterschiedlichen Detaillierungsgrad aufweist. Somit werden von Stufe zu Stufe die Informationen präzisiert und gleichzeitig der Abstraktionsgrad verringert.

Der Aussagegehalt geht bei dieser Komplexitätsminderung nicht verloren, da jeder Teilbereich eine eigene Problemstellung hat und sich an dieser orientiert. Durch die Trennung der Darstellungssichten in eine Navigationssicht und eine operative Sicht werden in den einzelnen Monitoren jeweils nur die zu einem durch den Anwender ausgewählten Wissenspaket gehörenden Informationen und Inhalte näher

betrachtet, so dass ein intransparenter Informations-Mix vermieden wird. Die Disaggregation erfolgt dabei von einer hohen Ebene, z. B. der Komposition der Projektinstanzen, bis hin zu einzelnen Tabelleneinträgen für die Umsetzung eines konkreten Wissenspakets.

Gegenstand der folgenden Betrachtungen ist die Entwicklung eines Fundamentalmodells als zentraler Bestandteil der THESEUS-Methode. Dieses Fundamentalmodell beschreibt - in Analogie zum Begriff der Fundamentalbibliothek - die Umsetzungs-komponenten in ihrer Gesamtheit, wie sie bei der Realisierung der Adaptionsergebnisse in Bezug auf eine betriebswirtschaftliche Softwarebibliothek vorkommen können. Dieses Fundamentalmodell besteht aus einer Sammlung vordefinierter Wissenspakete, die den Funktionsumfang des Systems widerspiegeln und den zugeordneten Inhalten aus den verknüpften Werkzeugen.

Durch das Monitor-Konzept, d. h. die spezifizierte Darstellung der relevanten Inhalte, wird der Aussagegehalt konkretisiert und die Transparenz für den Anwender erhöht.

4.2.4 Werkzeugunterstützung

Der THESEUS-Ansatz beschränkt sich bewusst auf die Entwicklung softwareorientierter Modelle, so dass die Anzahl der Werkzeuge und die Komplexität überschaubar bleiben. Damit wird es ermöglicht, mit einfachen Mitteln eine aktive und flexible Realisierung der Adaptionsergebnisse zu gewährleisten.

Die Zuordnungsebene gestattet eine flexible Werkzeuganbindung. Das intuitiv zu bedienende Werkzeug erlaubt dem Anwender auf bestehende Inhalte aus dem Fundamentalmodell zurückzugreifen oder individuelle Erweiterungen im Sinne von zusätzlichen Paketen bzw. Zuordnungen vorzunehmen. Die Gestaltung des Werkzeugs wird in den Kapiteln 5 und 6 detailliert vorgestellt. Hierbei wird u. a. darauf eingegangen, wie die Inhalte der einzelnen Modellebenen erarbeitet und präsentiert werden. Im Anschluss daran wird der Weg vom Fundamental- zum Individualmodell gezeigt. Diese aktive Komponente besteht zum einen aus der Widmung bzw. Komposition der Projektstruktur durch die einfließenden Informationsquellen (z. B. die Rasterhypothese) und zum anderen aus der anpassenden Modellierung. Dabei wird auch die logische Interaktion zwischen den Informationsquellen und THESEUS betrachtet.

4.3 Wissenspakete

Als zentraler Wissensträger wird das Wissenspaket (WiPak) definiert. Ein Wissenspaket ist eine in sich abgeschlossene Sammlung von zusammenhängenden Aktivitäten im Rahmen der Implementierung einer betriebswirtschaftlichen Softwarebibliothek. Ähnlich einer Knowledge Map aus dem Bereich des Wissensmanagements wird auch in der hier entwickelten Methode von zentralen Objekten ausgegangen, die alle relevanten Kenntnisse sammeln und verbinden. Das WiPak dient als system- und projektspezifischer Wissensträger für alle Informationen, die bei der Realisierung von Softwarefunktionalitäten ermittelt werden. Durch die Verbindungen der einzelnen Pakete entsteht eine Wissensbasis für den Bereich der Softwarebibliothek, der u. a. folgende Aufgaben zukommen [TEUF99, S. 161]:

- Allgemein zugängliche Informationsquelle,
- Konsistentes Wissensmanagement (Versionsverwaltung, Lokalisierung etc.),
- Redundanzfreiheit der Objekte und
- Zugriff auf alle relevanten Informationen zu einem Sachverhalt.

Die bisher vorherrschende Doktrin der zielgerichteten Navigationsstruktur wird aufgegeben und durch die Themenorientierung ersetzt. Dies bedeutet, dass die Softwareimplementierung nicht mehr einer vorgegebenen Struktur, z. B. in Form einer Prozesshierarchie, folgt, sondern vielmehr eine flexible Kombination von in sich abgeschlossenen Themengebieten genutzt wird. Die gekapselten Pakete erlauben eine dynamische Projektorganisation und eine Wiederverwendung des gesammelten Wissens. Über eine regelbasierte Zuordnungslogik werden den einzelnen Paketen die Elemente aus den unterschiedlichen Werkzeugen (z. B. LIVE KIT Power) und Bibliotheksbereichen (z. B. Stammdaten- und Prozessinformationen) zugewiesen, die zur vollständigen Abbildung des Pakets im System nötig sind. Die Inhalte und Informationen über konkrete Ausprägungen und Anforderungen werden auch weiterhin durch den Einsatz der Spezialwerkzeuge ermittelt, lediglich die Aufbereitung und Nutzung dieser Informationen wird über die zentralen Wissenspakete abgewickelt.

Für eine genauere Beschreibung der Funktionen zur Verwaltung von Wissen mit THESEUS ist es wichtig, das Konzept des Kontextes zu verstehen. Jedes Informationsobjekt in Form eines Pakets besteht aus genau einem logischen und mindes-

tens einem physischen Objekt. Ein logisches Objekt kann auf mehrere physische Objekte verweisen, die jeweils eine andere Version derselben Informationen beinhalten. Beispielsweise kann ein Dokument zur Abbildung der Systemeinstellungen in unterschiedlichen Sprachen existieren und releaseabhängige Informationen enthalten. Alle physischen Objekte werden über die Wissenspakete gebündelt, obgleich sie originär zu einem anderen Kontext gehören.

Damit eine flexible Strukturierung der einzelnen WiPaks möglich wird, müssen die Pakete standardisiert werden. Um die Transparenz für den Anwender zu erhöhen und das Vorgehen im Umgang mit den Paketen zu erleichtern, müssen die WiPaks zunächst klassifiziert werden. Ziel ist dabei u. a. die Aufbereitung der Komplexität des Untersuchungsgegenstands für eine leichtere Verständlichkeit. Diese Klassifizierung wird in den folgenden Kapiteln erläutert, da sie die Ausgangsbasis für die weiteren Betrachtungen, insbesondere der Integrationsaspekte bildet.

4.3.1 Typisierung

Über das von MEHLICH entwickelte MEDEA-Konzept ist es möglich, aufgrund verschiedener Merkmalsausprägungen ein Anwendersegment zu bestimmen [MEHL98, S. 265-266]. In Abhängigkeit des Anwendersegments können dann die Wissenspakete identifiziert werden, die im Rahmen der Softwareeinführung relevant sind. Die Informationen bzw. Ausprägungen, die zu einem Wissenspaket hinterlegt werden müssen, sind ebenfalls abhängig von der Auswahl des Anwendersegments, d. h. ein WiPak wird mit Bezug zu einem Anwendersegment definiert und ausgeprägt. Dieser Zusammenhang soll im Folgenden als Widmung bezeichnet werden. Demgegenüber entstehen aber auch WiPaks, die unabhängig vom Anwendersegment Gültigkeit haben. Derartige Pakete werden als generisch bezeichnet. Diese generischen Pakete sind abhängig von der einzuführenden Softwarebibliothek und finden u. a. in Projekten ohne vorherige Rasterhypothese Anwendung. Abbildung 4-3 verdeutlicht noch einmal den Zusammenhang zwischen MEDEA, SPARTA und der Widmung einzelner Pakete.

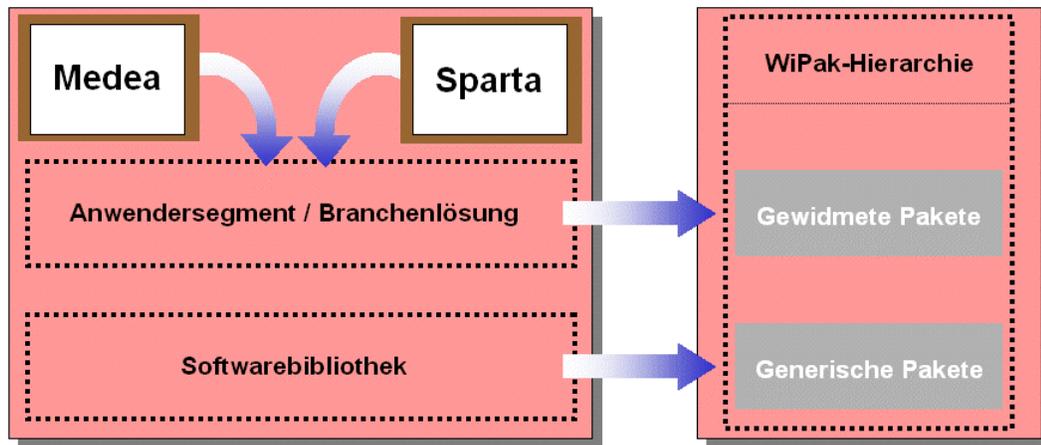


Abbildung 4-3: Widmung der Wissenspakete durch MEDEA

4.3.2 Charakterisierung

Neben der Widmung spielt die Art des Wissenspakets eine entscheidende Rolle für die Ausprägungen. Der Charakter eines WiPaks bestimmt u. a. die Position in der Paketstruktur, das Verhältnis zu anderen Wissenspaketen und die Integration in die Werkzeuglandschaft des ITHAKA-Ansatzes. Die verschiedenen Pakete werden in sieben Kategorien eingeteilt, die jeweils unterschiedliche Funktionen determinieren:

- Organisation,
- Prozess,
- Stammdaten,
- Berichte.
- Funktion,
- Basis,
- Schnittstellen und

Die Einteilung der Pakete in die sieben Kategorien wurde gewählt, da diese Aspekte für die Implementierung von Softwarebibliotheken zwingend notwendig sind. Ein Grundsatz bei der Definition der Pakete ist daher, dass es zu jeder Kategorie mindestens ein WiPak geben muss. Jedes Wissenspaket gehört dabei jeweils genau einer Kategorie an. Die Art des Pakets bestimmt vor allem die Interaktionen mit anderen Werkzeugen, so sind Organisations-WiPaks beispielsweise eng verbunden mit Elementen aus dem von BÄTZ entwickelten OLYMP-Konzept [BÄTZ99]. Prozess- und Funktionspakete vereinen dagegen eher Elemente aus dem PENELOPE-Ansatz von VOGELANG [VOGE97].

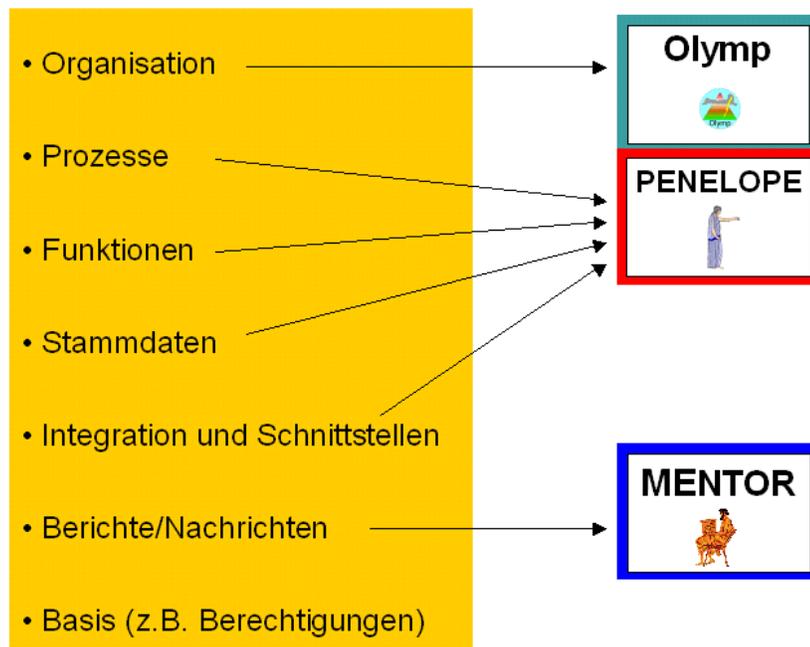


Abbildung 4-4: Charakterisierung der Wissenspakete

Abbildung 4-4 verdeutlicht den Zusammenhang zwischen der Art eines Wissenspakets und seiner Verknüpfung mit anderen Elementen des ITHAKA-Konzepts.

Die Interaktion der Pakete ist dabei nicht auf die Elemente des ITHAKA-Ansatzes beschränkt. Die Flexibilität der neuen Struktur erlaubt es, eine Vielzahl unterschiedlicher Werkzeuge anzubinden und damit als umfassender Wissensträger zu fungieren. Die einzelnen Paketarten werden im folgenden näher vorgestellt und durch Beispiele veranschaulicht.

ORGANISATIONSPAKETE

Organisationspakete umfassen alle Aktivitäten, die nötig sind, um die bestehende Struktur des Unternehmens konsistent in der Softwarebibliothek abzubilden. In Verbindung mit OLYMP [BÄTZ99] ist es möglich, die Aspekte der dynamischen Organisationsgestaltung vollständig aufzuzeigen und eine einfache Navigation durch die verschiedenen Ebenen der Organisationsmodellierung zu erleichtern. Die Organisationsstruktur wird in der Regel zu Beginn eines Adaptionprojekts festgelegt, da sie starke Auswirkungen auf die späteren Prozessabläufe hat. Als Beispiel kann an dieser Stelle die Organisation der Lagerverwaltung als eigenständiges Paket genannt werden. Über Verbindungen zu den Organisationsmonitoren von OLYMP können die Anforderungen an die Lagerstruktur anwenderfreundlich ge-

bündelt und für die Adaption der Softwarebibliothek in diesem Bereich aufbereitet werden.

STAMMDATENPAKETE

Stammdatenpakete bilden die relevanten Einführungsaktivitäten für die Pflege der Daten ab, die sich im Zeitablauf nicht verändern. Dazu gehören beispielsweise der Materialstamm oder die Arbeitspläne der Produktion. Diese Pakete sind meist generischer Art, da sie unabhängig von dem jeweiligen Anwendersegment benötigt werden. Lediglich die Ausprägung der Pakete kann von Segment zu Segment variieren. Ein weiteres Merkmal dieser Pakete ist die vielfältige Integration der Stammdaten. Eine eindeutige Zuordnung zu Unternehmensprozessbereichen ist in den meisten Fällen nicht nötig. So wird der Materialstamm zum Beispiel sowohl im Bereich der Materialwirtschaft für eine Vielzahl von Prozessabläufen benötigt, daneben greift aber auch der Bereich des Vertriebs auf die Informationen im Materialstamm zurück, um die Prozesse abzubilden. Aspekte der Stammdatenpflege, die nur in einem bestimmten Prozess relevant sind, werden nicht als eigenständige Stammdatenpakete abgelegt. Derartige Aspekte werden innerhalb des jeweiligen Prozesspakets subsumiert und dort verwaltet. So ist gewährleistet, dass alle Bereiche auf die generischen Stammdatenpakete zugreifen können und unnötige Paketverknüpfungen vermieden werden.

Weitere Beispiele für Stammdatenpakete sind die Pflege Equipmentstammsätze aus dem Bereich der Instandhaltung und die Definition von Konten innerhalb der Finanzbuchhaltung.

PROZESSPAKETE

Innerhalb eines Prozesspakets werden alle direkt zu diesem Prozess gehörenden Aspekte subsumiert. Bei der Entwicklung von Prozess-WiPaks ist darauf zu achten, dass generische Bestandteile des Prozesses, d. h. Informationen, die durch eigenständige Pakete abgebildet sind (z. B. Stammdaten, Berichte), nicht im Prozesspaket selbst sondern durch eine Paketverknüpfung realisiert werden. Das Prozesspaket sammelt alle Stammdaten und Funktionen, die für den Ablauf unbedingt erforderlich sind und nur dort verwendet werden. Häufig kommt es vor, dass ein Prozess-WiPak in mehreren Anforderungssegmenten eingesetzt wird, diese Konstellation ist besonders im Konzernumfeld häufig anzutreffen. Aus diesem Grund

müssen die Prozesspakete mit der Anzahl der Segmente multipliziert werden können.

Ein Prozesspaket ist beispielsweise die vollständige Umsetzung der Geräteablesung aus dem Anwendersegment der Versorgungsindustrie. Innerhalb dieses Pakets werden alle Anforderungen des Unternehmens an die Abwicklung der Geräteablesung abgefragt und anschließend im System abgebildet.

FUNKTIONSPAKETE

Funktionspakete bilden komplexe Funktionen der Softwarebibliothek ab, die wiederum mit anderen Wissenspaketen verknüpft sind. Die Inhalte der Funktionspakete müssen gewährleisten, dass die Funktion komplett abgehandelt werden kann. Meist werden Funktionspakete mit den Prozesspaketen verknüpft, die auf diese komplexen Funktionen zurückgreifen. Besonders für Prozesse mit vielen Varianten empfiehlt es sich, die Funktionen, die in allen Varianten verwendet werden, als eigenständige Pakete abzubilden und dann an den entsprechenden Stellen zu verknüpfen.

Als Beispiel für eine generische Funktion kann die Materialverfügbarkeitsprüfung genannt werden. Diese Funktion wird in vielen Prozessen der Fertigungs- und Instandhaltungsauftragsbearbeitung in der gleichen Weise durchgeführt. Eine redundante Pflege dieses Sachverhalts kann somit vermieden werden.

SCHNITTSTELLENPAKETE

Schnittstellenpakete bilden die Interaktion zwischen zwei Wissenspaketen ab und dienen der Evaluierung der Anforderungen an den Informationsfluss. Verbindungen zu Elementen des Schnittstellenmonitors der PENELOPE sind offensichtlich. Vor allem die genaue Definition der auszutauschenden Dateninhalte und -strukturen ist Gegenstand derartiger Paketdefinitionen. Die Anbindung von externen Systemen zur Durchführung von Qualitätsprüfungen an die SAP-Software soll an dieser Stelle als Beispiel genannt werden. Die Definition der zu prüfenden Materialspezifikationen muss sowohl aus Sicht des SAP-Systems als auch aus dem Blickwinkel des Fremdsystems abgestimmt werden.

NACHRICHTEN- BZW. BERICHTSPAKETE

Nachrichten- bzw. Berichtspakete umfassen alle Aspekte zur Implementierung einzelner Berichtsobjekte. In diesem Bereich herrscht eine große Affinität zu dem von WEDLICH entwickelten MENTOR-Konzept [WED97]. Den Paketen werden dabei bestehende Berichtshierarchien oder direkt einzelne Berichtsobjekte zugeordnet und so die Navigation durch das Modell gesteuert.

Analysen der Offenen-Posten-Bearbeitung aus dem Bereich der Debitorenbuchhaltung sind ein solches in sich abgeschlossenes Berichtspaket, dessen Anforderungen gezielt über die Integration einzelner Berichte umgesetzt werden können.

BASISPAKETE

Basispakete dienen in erster Linie dazu, die Grundfunktionen der Softwarebibliothek zu implementieren. In diesen Bereich fällt beispielsweise die Ausprägung des Berechtigungs- oder die Gestaltung des Transportwesens für die Systemlandschaft des Unternehmens.

4.3.3 Allgemeine Eigenschaften

Zur Definition und Abgrenzung der einzelnen WiPaks ist es notwendig, bestimmte Eigenschaften zu identifizieren, die für ein Wissenspaket Geltung haben. Derartige Eigenschaften sind teilweise abhängig vom Charakter eines Pakets (vgl. Kapitel 4.3.2). Es gibt darüber hinaus jedoch einige Merkmale, die allgemeine Gültigkeit besitzen.

UNABHÄNGIGKEIT

Ein Wissenspaket sollte weitestgehend unabhängig sein, d. h. bei der Definition eines Pakets muss auf möglichst wenige Abhängigkeiten zu anderen Paketen geachtet werden. Dadurch ist gewährleistet, dass eine komfortable Navigation über alle Wissenspakete hinweg durchgeführt und die Bearbeitung der einzelnen Pakete im Rahmen der Realisierungsphase eines Projekts entkoppelt werden kann.

VOLLSTÄNDIGKEIT

Ein wesentliches Merkmal der Wissenspakete ist die Vollständigkeit. Es muss gewährleistet sein, dass durch das Paket der komplette betriebswirtschaftliche Sach-

verhalt abgebildet ist, ohne Lücken in der Adaption auftreten zu lassen. Dies impliziert, dass alle relevanten Informationen, z. B. alle Reduktionsfragen aus dem ODYSSEUS-Umfeld und alle Kernprozesse aus PENELOPE, die direkt zur Implementierung dieses Pakets nötig sind, vollständig angesprochen und bearbeitet werden. Besonders wichtig ist die vollständige Abbildung der Customizingaktivitäten, die für ein Paket Gültigkeit haben.

VORGÄNGER / NACHFOLGER

Jedes Wissenspaket kann einen Vorgänger und/oder einen Nachfolger besitzen. Ziel dieser Eigenschaft ist es, die Projektabfolge (insbesondere die Organisation der Umsetzung) zu steuern und nachvollziehbar zu machen. Ausprägungen des Vorgängerpakets haben dabei Auswirkungen auf das Nachfolgerpaket. Über Vorgängerpakete können Bestandteile des Nachfolgerpakets reduziert oder vorbelegt werden. Die Vorgänger-/Nachfolgereigenschaft bezieht sich dabei nur auf eine direkte Beziehung der Pakete, indirekte Verflechtungen werden nicht berücksichtigt, da diese in der Struktur der Wissenspakete festgelegt sind.

HISTORIE

Zur konsequenten Umsetzung des CSE mit Hilfe der Wissenspakete ist es nötig, dass alle Informationen mit einem Zeitstempel versehen sind und somit Änderungen jederzeit nachvollzogen werden können. Diese Eigenschaft erhöht die Transparenz und den Nutzen der WiPaks gegenüber den bisher bestehenden Ansätzen.

STABILITÄT

Ein Wissenspaket ändert sich im Zeitablauf nicht, d. h. es bleibt stabil. Die Dynamik wird durch die sich ändernden Inhalte und Zuordnungen des Pakets gewährleistet. Im Lebenszyklus der Bibliothek bleibt die Bedeutung und der Einsatz eines Wissenspakets konstant, um die Vergleichbarkeit verschiedener Lösungen zu gewährleisten. Durch das hier entwickelte Stufenkonzept sowie die damit verbundene Trennung von Wissenspaket und zugeordnetem Adaptionswerkzeug sind die Voraussetzungen für eine konzeptionelle und inhaltliche Stabilität geschaffen worden.

4.3.4 Attribute eines Wissenspakets

Wissenspakete stellen die unterste Ebene dar. Sie enthalten konkrete Handlungsanweisungen bzw. dokumentieren im Detail die vorgenommenen Tätigkeiten bzw. Systemeinstellungen. Zu diesem Zweck muss jedes WiPak über gewisse Attribute (nicht Zuordnungen) verfügen. Tabelle 4-1 zeigt eine Auflistung der wichtigsten Attribute, die einem Paket zugewiesen werden müssen.

Tabelle 4-1: Attribute eines Wissenspakets (Kopfdaten)

Attribut	Eigenschaften
Nummer	Jedes Wissenspaket erhält eine Nummer, aus der die Anordnung in der Sequenz ermittelt werden kann.
Bezeichnung	Die Bezeichnung dient einer schnellen Identifikation, wobei man sich an einer vorhandenen Referenzliste orientiert. Diese Referenzliste umfasst Begriffe wie die SAP-Organisationselemente, die Unternehmensorganisationselemente oder auch Customizing-Aktivitäten.
Beschreibung	In diesem Feld erfolgt die Beschreibung der Aktivitäten.
Klassifikation	Die Art des Pakets wird gemäß den definierten Typen aus Kapitel 4.3.2 zugeordnet, um gefilterte Auswertungen und Entwicklungen zu ermöglichen.
Verantwortliche/Beteiligte	Dem Paket werden Personen, Personengruppen oder Rollen zugewiesen, die sich für die Bearbeitung verantwortlich zeichnen.
Vorgänger/Nachfolger	Über diese beiden Felder werden die Abhängigkeiten der Pakete definiert. Jeweils ein Vorgänger und/oder ein Nachfolger sind möglich.
Versionskennzeichen	Jedes Paket verfügt über ein Versionskennzeichen, durch das die Entwicklungsstufen unterschieden werden können.
Fachbereichs-ID	Dem Paket wird ein generischer Fachbereich (Paket-Owner) zugeordnet.
Zusätzliche Attribute	Definition von weiteren Attribute in Abhängigkeit der Erfahrungen mit dem Prototypen.

4.3.5 Zuordnungen

Im folgenden werden mögliche Zuordnungen zu einem Wissenspaket vorgestellt, die eine Nutzung von THESEUS als Erweiterung des ITHAKA-Konzepts zulassen. Auf Zuordnungen weiterer Werkzeuge wird aus Gründen der Übersichtlichkeit verzichtet, die methodische Einordnung ist davon nicht betroffen.

Zunächst werden einige der in Kapitel 3 vorgestellten Werkzeuge auf ihre Eignung bezüglich eines integrierten Einsatzes mit THESEUS geprüft (Tabelle 4-2), bevor dann auf die im Fundamentalmodell abgebildeten Assoziationen näher eingegangen wird. Die Prüfung erfolgt anhand der in Kapitel 4.1.3 genannten Kriterien.

Tabelle 4-2: Leistungsmerkmale ausgewählter Adaptionswerkzeuge

Werkzeug	Beurteilung
LIVE KIT Structure	<ul style="list-style-type: none"> • Das Regelwerk des LIVE KIT Structure ist in einer Datenbank gespeichert (Development Data Dictionary, DDD). Alle Informationen über die Werte der einzelnen Parameter werden in einer datenbankähnlichen Struktur, d. h. in formatierten Dateien abgelegt (Online Data Dictionary, ODD). • Auf die Dateien des ODD kann über standardisierte Schnittstellen zugegriffen werden. • Die einzelnen Elemente aus dem Fundamentalmodell (z. B. Fragen, Profile, Fachbereiche) sind eindeutig anhand einer Parameterkennung identifizierbar. • Zur Speicherung der projektspezifischen Werte eines Parameters wird ein separates Verzeichnis angelegt, in dem alle relevanten Informationen mit eindeutigem Projektbezug zu finden sind. Zusätzlich wird für jedes Projekt eine Datei (check.txt) geschrieben, die in formatierter Form alle Parameter des DDD mit dem jeweils ermittelten Wert enthält.
LIVE KIT Power & Control	<ul style="list-style-type: none"> • Das Werkzeug basiert auf einer zentralen Datenbank (power.mdb). • Die Datenbank unterstützt einen ODBC-Zugriff. • Die einzelnen Elemente aus dem Fundamentalmodell sind eindeutig anhand einer ID identifizierbar. • Über die anpassende Referenzmodellierung wird das Fundamentalmodell in ein projektspezifisches Individualmodell überführt. Dabei werden die Ausprägungen der Elemente (aktiv/inaktiv) in eigenständigen Tabellen hinterlegt.
Rasterhypothese	<ul style="list-style-type: none"> • Die betriebswirtschaftlichen Raster sind in einer Datenbank abgelegt. • Die Datenbank unterstützt einen ODBC-Zugriff. • Die einzelnen Elemente (z. B. Merkmale, Merkmalsausprägungen) sind eindeutig anhand einer ID identifizierbar. • Die Rasterhypothese wird für ein spezielles Projekt erstellt und enthält in Abhängigkeit des Projekts die Ausprägungen der verschiedenen Merkmale über mehrere Raster hinweg.

Reverse Business Assistant	<ul style="list-style-type: none"> • Der Reverse Business Assistant greift auf die zentrale RBE-Datenbank zu, in der alle zu evaluierenden Sets und Extrakte abgelegt sind. • Die Datenbank unterstützt einen ODBC-Zugriff. • Die einzelnen RBE-Sets sind in der Datenbank anhand einer ID eindeutig identifizierbar. • Die Bewertung der einzelnen RBE-Sets erfolgt in Abhängigkeit eines kundenindividuellen Extrakts des produktiven SAP-Systems. In der Datenbank werden die Ergebnisse in getrennten Tabellenbereichen gespeichert und sind somit eindeutig abgrenzbar.
Q&Adb	<ul style="list-style-type: none"> • Das Werkzeug basiert auf einer zentralen Datenbank (qadb.mdb). • Die Datenbank unterstützt einen ODBC-Zugriff. • Die einzelnen Elemente sind in der Q&Adb anhand einer ID eindeutig identifizierbar. • Jedes ASAP-Projekt enthält eine fest definierte Verzeichnisstruktur. Die Q&Adb wird in einem eigenen Verzeichnis abgelegt und enthält alle projektspezifischen Informationen. Es ist nicht möglich, in einem Projekt mehrere Q&Adb zu verwenden.
Concept Check Tool	<ul style="list-style-type: none"> • Das Werkzeug basiert auf einer zentralen Datenbank (outline.mdb). • Die Datenbank unterstützt einen ODBC-Zugriff. • Die einzelnen Elemente sind in der Datenbank anhand einer ID eindeutig identifizierbar. • Jedes ASAP-Projekt enthält eine fest definierte Verzeichnisstruktur. Das Concept Check Tool wird in einem eigenen Verzeichnis abgelegt und enthält alle projektspezifischen Informationen.

Anhand der Leistungsmerkmale in Tabelle 4-2 wird deutlich, dass die betrachteten Werkzeuge die Voraussetzungen für einen integrierten Einsatz mit THESEUS erfüllen und somit für die Aufnahme in das Fundamentalmodell geeignet sind.

Tabelle 4-3 enthält ausgewählte Zuordnungen aus den Werkzeugen des ITHAKA-Konzepts, die zum Aufbau des Fundamentalmodells dienen.

Tabelle 4-3: Zuordnungen eines Wissenspakets

Art der Zuordnung	Beschreibung
LIVE KIT Structure	Die Zuordnung erfolgt auf Fragen bzw. Profilebene. Dabei spielt es keine Rolle, aus welchen Fachbereichen oder Komponenten die einzelnen Fragen bzw. Profile stammen (vgl. Kapitel 7.2.2).
LIVE KIT Power	Die Zuordnung erfolgt auf Beleg bzw. Schnittstellenebene. Dabei spielt es keine Rolle, aus welchen Kernprozessen die einzelnen Elemente stammen (vgl. Kapitel 7.2.3).
LIVE KIT Control	Dem Wissenspaket werden ein oder mehrere Berichte zugeordnet. Dabei spielt es keine Rolle, aus welchen Kernprozessen die Berichte stammen.
Rasterhypothese	Einzelne Merkmalsausprägungen aus dem MEDEA-Umfeld werden dem Wissenspaket zugeordnet und ermöglichen so die Nutzung der Rasterhypothese als Grundlage für den THESEUS-Einsatz (vgl. Kapitel 7.2.1).
Reverse Business Engineering	Den Wissenspaketen werden RBE-Sets zugeordnet. Über diese Verbindungen ist es möglich, auf Basis einer RBE-Analyse Informationen über bearbeitete oder benötigte Wissenspakete zu erlangen und so ein effektives CSE zu unterstützen (vgl. Kapitel 7.2.4).
Customizing	Die Zuordnung erfolgt auf Ebene der IMG-Bausteine. Entscheidend ist vor allem die Vollständigkeit. Es muss gewährleistet sein, dass nach dem Bearbeiten der zugeordneten Customizingaktivitäten das Wissenspaket in der richtigen Ausprägung im System verfü- und nutzbar ist.
Konfiguration	Dem Wissenspaket werden einzelne Business Configuration Sets zugeordnet. Dabei sind in diesem Zusammenhang nur die direkten Zuordnungen gemeint, es besteht zudem die Möglichkeit einer indirekten Zuweisung von BC-Sets über die betriebswirtschaftlichen Profile des Anforderungsnavigators (vgl. Kapitel 6.3.5).
Zuordnungsdokumentation	Ähnlich der Anbindung von Business Configuration Sets, werden einem Wissenspaket Zuordnungsdokumentationen zugeordnet, die detaillierte Beschreibungen der Customizingeinstellungen enthalten. Auch hier sind neben den direkten Zuordnungen indirekte über die Profile möglich.
Aufwand	Die Zuordnung einzelner Dienstleistungsvariablen, die über den Anforderungsnavigator bewertet werden, ermöglicht die Aufwandsermittlung zur Einführung eines Wissenspakets.
Online-Dokumentation	Zur Beschreibung des Inhalts bzw. der Bedeutung eines Wissenspakets, werden Zuordnungen von Teilen der Online Dokumentation der Softwarebibliothek vorgenommen.

4.3.6 Prüffälle

Die Wissenspakete bilden den Kern der THESEUS-Methode, die das gesamte Wissen, welches zur Umsetzung eines Themas in der produktiven Softwarelösung nötig ist, in sich vereinen. Pro Wissenspaket wird ein Prüffall definiert, der mittels einer heuristischen Vorgehensweise ermittelt, ob das betrachtete Paket im jeweiligen Anwendungsfall einen aktiven oder inaktiven Status aufweist. Ziel dabei ist die automatische Identifikation der relevanten Umsetzungsschritte im Rahmen der Softwareadaptation. Zentraler Punkt der hier beschriebenen Vorgehensweise ist, dass alle Informationen, die notwendig sind, um die Relevanz oder Nicht-Relevanz eines Pakets festzustellen, an jedem einzelnen Wissenspaket hängen. Demnach muss für jedes Paket ein Analyse-Set definiert werden, das eine Zusammenstellung der zu prüfenden Regeln liefert. Tabelle 4-4 gibt die Status wieder, die dabei von den Wissenspaketen eingenommen werden können. Die Unterscheidung in die beiden Status „relevant“ und „nicht relevant“ ist nicht ausreichend, da in manchen Fällen die Informationsquellen keine eindeutige Relevanz bestimmen können. Durch die Mehrfachzuordnungen von Reduktionsfragen oder Prozessbelegen zu unterschiedlichen Paketen kann es möglich sein, dass Teilaspekte des Themas im Anforderungskatalog des Unternehmens zutreffen, die Umsetzung aber nicht Gegenstand der Betrachtung sein soll.

Tabelle 4-4: Status der Wissenspakete nach dem Einlesen der Informationsquellen

Relevant	Das Wissenspaket wurde aufgrund der Prüfungen als sicher (Wahrscheinlichkeit von 100%) relevant identifiziert.
wahrscheinlich relevant	Das Wissenspaket ist nur mit einer Wahrscheinlichkeit von 60% relevant, d. h. bestimmte Prüfungen weisen zwar auf eine Aktivität hin, es müssen allerdings noch weitere Untersuchungen stattfinden, um die Aussage zu verifizieren.
wahrscheinlich nicht relevant	Das Wissenspaket ist nur mit einer Wahrscheinlichkeit von 60% nicht relevant, d. h. bestimmte Prüfungen weisen zwar auf eine Nicht-Relevanz hin, es müssen allerdings noch weitere Untersuchungen stattfinden, um die Aussage zu verifizieren.
nicht relevant	Das Wissenspaket wurde aufgrund der Prüfungen als sicher (Wahrscheinlichkeit von 100%) nicht relevant identifiziert.
indifferent	Aufgrund mangelnder Informationen oder widersprüchlicher Ergebnisse kann keine Aussage bzgl. der Relevanz des untersuchten Elements getroffen werden.

Die Prüfschritte für die Feststellung der Paketrelevanz beziehen sich auf die in der Zuordnungsebene hinterlegten Informationen bezüglich eines Wissenspakets. In Abhängigkeit der eingelesenen Informationsquelle (z. B. Auswertungsdateien aus dem Anforderungsnavigator) werden die zugeordneten Elemente eines Pakets geprüft. In der ODYSSEUS-Checkliste stehen beispielsweise für alle Fragen bzw. Profile die gegebenen Antworten. Wurde eine zugeordnete Frage während der Anforderungsanalyse mit „Ja“ beantwortet, so lässt dies auf die Relevanz des Wissenspakets schließen. Innerhalb eines Prüffalls werden alle zugeordneten Elemente zu einem Wissenspaket auf ihren Wert aus der Informationsquelle geprüft. Sind alle zugewiesenen Objekte mit einer positiven Antwort evaluiert, ist das Wissenspaket sicher relevant. Über Schwellwerte und Gewichtung der einzelnen Elemente wird festgelegt, welche Prozentzahl der Antworten bzw. Ergebnisse zum Status „wahrscheinlich relevant“ usw. führt. Wichtig in diesem Zusammenhang ist, dass die Gewichte der zugeordneten Elemente in Abhängigkeit des jeweils betrachteten Pakets hinterlegt werden. So wird gewährleistet, dass ein Objekt vorkommensabhängig die Relevanzentscheidung des betriebswirtschaftlichen Themas beeinflusst. Sind alle Ergebnisse negativ bewertet, ist das Paket nicht relevant. Lassen die Werte keine Statusvergabe zu, wird der Status auf „indifferent“ gesetzt, eine manuelle Entscheidung erfolgt durch den Anwender.

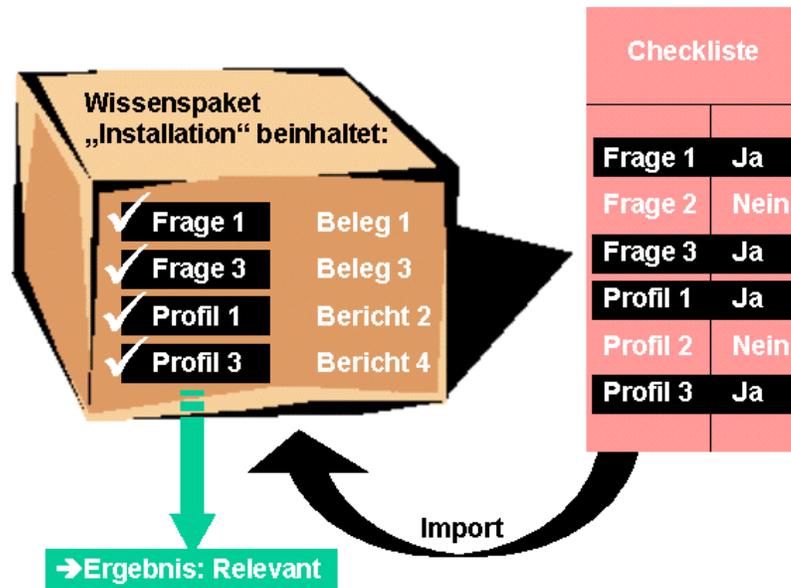


Abbildung 4-5: Ablauf der Relevanzentscheidung bei Zugriff auf eine Checkliste

Abbildung 4-5 verdeutlicht nochmals schematisch den Ablauf der Relevanzentscheidung aufgrund der Ergebnisse einer Anforderungsanalyse. Die Abwicklung beim Bezug auf andere Informationsquellen, wie z. B. einer Rasterhypothese oder einem modellierten Individualmodell des LIVE KIT Power, verläuft auf die gleiche Weise.

4.3.7 Lebenszyklus

Wissenspakete durchlaufen einen gewissen Lebenszyklus. Während des Einsatzes einer Softwarebibliothek sammelt das WiPak zunehmend Informationen, d. h. der Informationsgehalt eines Pakets nimmt im Zeitverlauf stetig zu. Wissenspakete werden vorwiegend in der Realisierungsphase eingesetzt, um das Customizing zu unterstützen. Durch die Verwendung in Folgeprojekten wächst der Informationsgehalt durch neue Anforderungen oder geänderte Systemeinstellungen weiter an. Der Lebenszyklus kann in folgende Phasen eingeteilt werden:

- Entwicklung,
- Business Blue Print,
- Realisierung und
- Re-Adaption.

Die Phasen sind auf das klassische Vorgehensmodell zur Einführung einer Standardanwendungssoftware, wie es bei STRELLER beschrieben ist [STRE99, S. 198-197], abgestimmt. Zusätzlich wurde die Entwicklungsphase aufgenommen, da sie die Definition eines Pakets und die Ablage in der regelbasierten Zuordnungsdatenbank darstellt. Während dieser Phase werden die einzelnen Pakete inhaltlich definiert, klassifiziert und die Einordnung in die Paketstruktur bestimmt. Darüber hinaus werden die für die Realisierung des Pakets nötigen Zuordnungen aus dem ITHAKA-Umfeld (z. B. betriebswirtschaftliche Profile aus ODYSSEUS, Kernprozesse aus PENELOPE) vorgenommen. Diese Phase ist projektunabhängig und dient zum Aufbau bzw. zur Ergänzung des Fundamentalmodells (Kapitel 5). Die weiteren Phasen sind dagegen projektabhängig, da die einzelnen Pakete als Wissensträger für konkrete Projekte eingesetzt werden.

Während der Business Blue Print Phase werden die Anforderungen bezüglich der Softwareausprägung durch das Unternehmen festgelegt. In dieser Phase kommen die unterschiedliche Werkzeuge (z. B. LIVE KIT Structure, LIVE KIT Power) zum Einsatz. Über die Zuordnung der Werkzeugelemente, können die für die Umsetzung relevanten Pakete identifiziert und gemäß dem Projektaufbau komponiert werden. Somit lassen sich die gesammelten Informationen ausgehend von einem Paket abrufen und nachvollziehen.

Nach Abschluss der Analysephase folgt die Umsetzung der Anforderungen durch die Konfiguration des Systems. Hierbei dient die Paketstruktur als zentrale Navigationshilfe für das Projektteam. Zu einem Wissenspaket werden Informationen über die einzustellenden Systemparameter aufgezeigt, die dann durch das Projektteam direkt in das System überführt werden müssen. Dabei spielt die neue Technik der Business Configuration Sets (BC-Sets) zur transportierbaren Speicherung von inhaltlich konsistenten Parametersätzen eine zentrale Rolle [SAP00, BC-CUS]. Durch die Zuordnung einzelner BC-Sets zu betriebswirtschaftlichen Profilen aus dem ODYSSEUS-Umfeld können die benötigten Parametersätze für ein Wissenspaket identifiziert und über Schnittstellen direkt in das SAP-System importiert werden. Die Implementierung wird durch diese Art der Systemkonfiguration erheblich beschleunigt. Somit ist es nun erstmals möglich, automatisch Systemeinstellungen aus einem Umsetzungswerkzeug vorzunehmen. Zudem können über diese BC-Sets die durchgeführten Customizingeinstellungen komfortabel mit Bezug zu einem Paket dokumentiert werden. In Kapitel 6.3.5 ist die Verwendung der BC-Sets in der A-

daptions-Workbench näher beschrieben. Neben der automatischen Systemkonfiguration werden aber auch manuelle Customizingeinstellungen durchgeführt, die mit Bezug zu einem WiPak beschrieben werden. Somit wächst gerade in dieser Phase der Informationsgehalt des WiPaks durch die Dokumentation der Systemkonfiguration an.

Im Rahmen des Continuous System Engineering wird die Softwarebibliothek an sich ändernde Umweltbedingungen angepasst. Im Zuge dieser Re-Adaption werden erneut Anforderungsanalysen mit den entsprechenden Werkzeugen durchgeführt, wobei auf die Ergebnisse der Einführungsprojekte zurückgegriffen wird. Für die Wissenspakete bedeutet dies, dass neben den Informationen des Ursprungsprojekts auch die Informationen aus dem Folgeprojekt gesammelt und verarbeitet werden. Über Änderungsprotokolle wird gewährleistet, dass jederzeit auf die ursprünglichen Informationen zurückgegriffen werden kann. Der Zyklus der einzelnen Phasen wiederholt sich dabei ständig, d. h. nach der erneuten Business Blue Print-Phase schließt sich eine weitere Realisierungsphase an, deren Customizingeinstellungen wiederum dokumentiert wird. Entscheidend ist in diesem Zyklus, dass immer ausgehend von denselben Paketen auf alle Informationen zugegriffen wird. Sämtliche Änderungen sind nachvollziehbar und somit für den Anwender in höchstem Maß transparent.

Abbildung 4-6 verdeutlicht noch einmal die einzelnen Phasen und Bestandteile des Lebenszyklus.

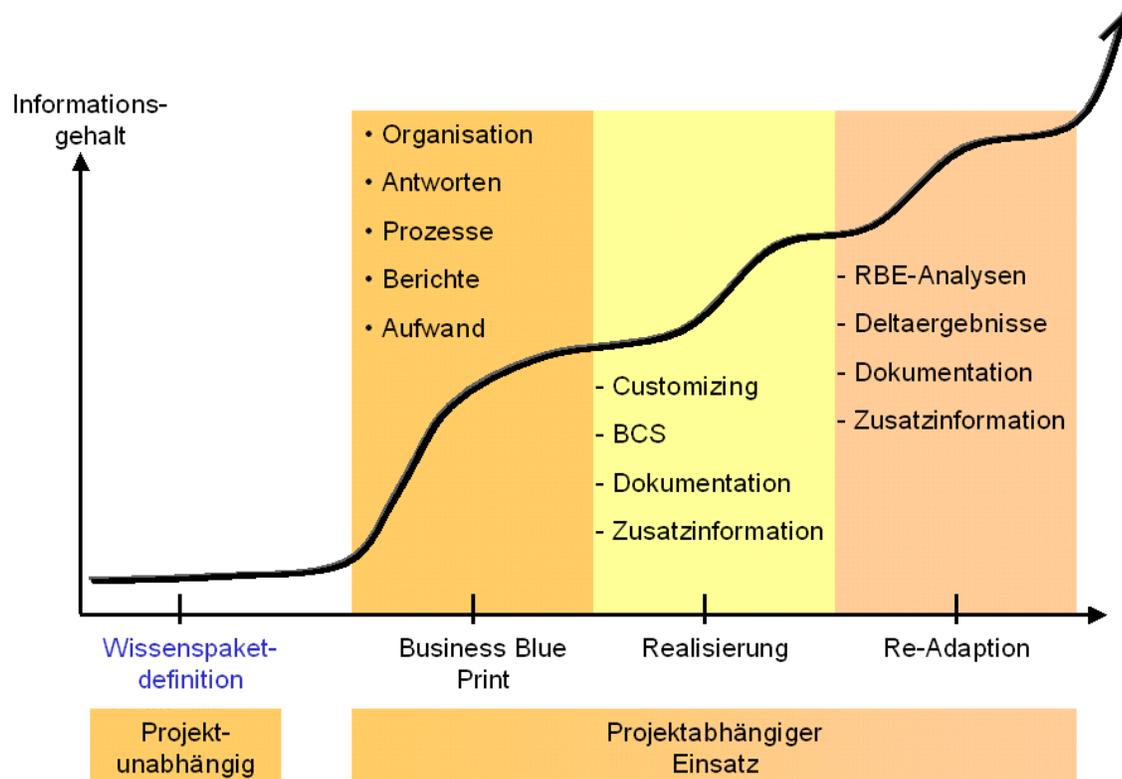


Abbildung 4-6: Lebenszyklus eines Wissenspakets

4.3.8 Informationsquellen

Für die Gestaltung und Definition der Wissenspakete lassen sich viele Informationsquellen heranziehen. Die folgenden Ausführungen beziehen sich vorwiegend auf Datenquellen, die in direktem Zusammenhang mit der SAP-Software stehen. Die Idee und Zielsetzung der themenorientierten Pakete basiert dagegen auf der Unabhängigkeit von Softwareprodukten und Werkzeuganbietern, daher können auch andere Quellen aus dem Bereich der betrieblichen Informationsverarbeitung sowie der Adaptionunterstützung herangezogen werden, die sich mit Unternehmensmodellen und -prozessen bzw. Ausprägungen bestimmter Softwarelösungen beschäftigen. Die jeweiligen Quellen bezogen auf die SAP-Software sollen zunächst vorgestellt und auf Ihren Informationsgehalt für die Wissenspakete untersucht werden.

PROZESSHIERARCHIE

Die Prozesshierarchie zeigt gezielt den Funktionsumfang und den betriebswirtschaftlichen Inhalt der SAP-Software, sie geht dabei auch auf verschiedene Varianten einzelner Prozesse ein. Diese Struktur liefert wichtige Informationen für die Definition der Wissenspakete, insbesondere können integrierte Prozesse identifiziert und evaluiert werden. Über die Zuordnung der Systemfunktionen innerhalb eines Prozesses, werden generische Funktionspakete erkannt und eigenständig modelliert.

KOMPONENTENHIERARCHIE

Über die Komponentenhierarchie werden die verschiedenen Funktionsbereiche der Softwarebibliothek abgebildet. Durch die starke funktionale Ausrichtung kann diese Struktur vorwiegend zur groben Einordnung einzelner Pakete genutzt werden. Die Zuordnung der Prozesse zu einzelnen Komponenten der SAP-Software zeigt in Verbindung mit der Prozesshierarchie die Zugehörigkeit der zu definierenden Pakete an.

ONLINE DOKUMENTATION

Die Online Dokumentation der SAP-Software ist eine HTML-basierte Sammlung von Beschreibungen der Funktionalitäten einzelner Module. Die Dokumentation ist in verschiedene Kapitel eingeteilt, die sich an der Komponentenhierarchie orientiert. Für die Gestaltung der Wissenspakete bietet die Online-Dokumentation wertvolle Informationen, da nicht nur komplexe Sachverhalte vollständig beschrieben, sondern auch Schnittstellen- und Integrationsaspekte erläutert werden. Besonders im Hinblick auf die Vollständigkeit können so wichtige Hinweise auf die einzubeziehenden Aspekte der Pakete gesammelt werden [SAP00a]. Abbildung 4-7 zeigt einen Ausschnitt aus der Dokumentationsumgebung des SAP-Systems.

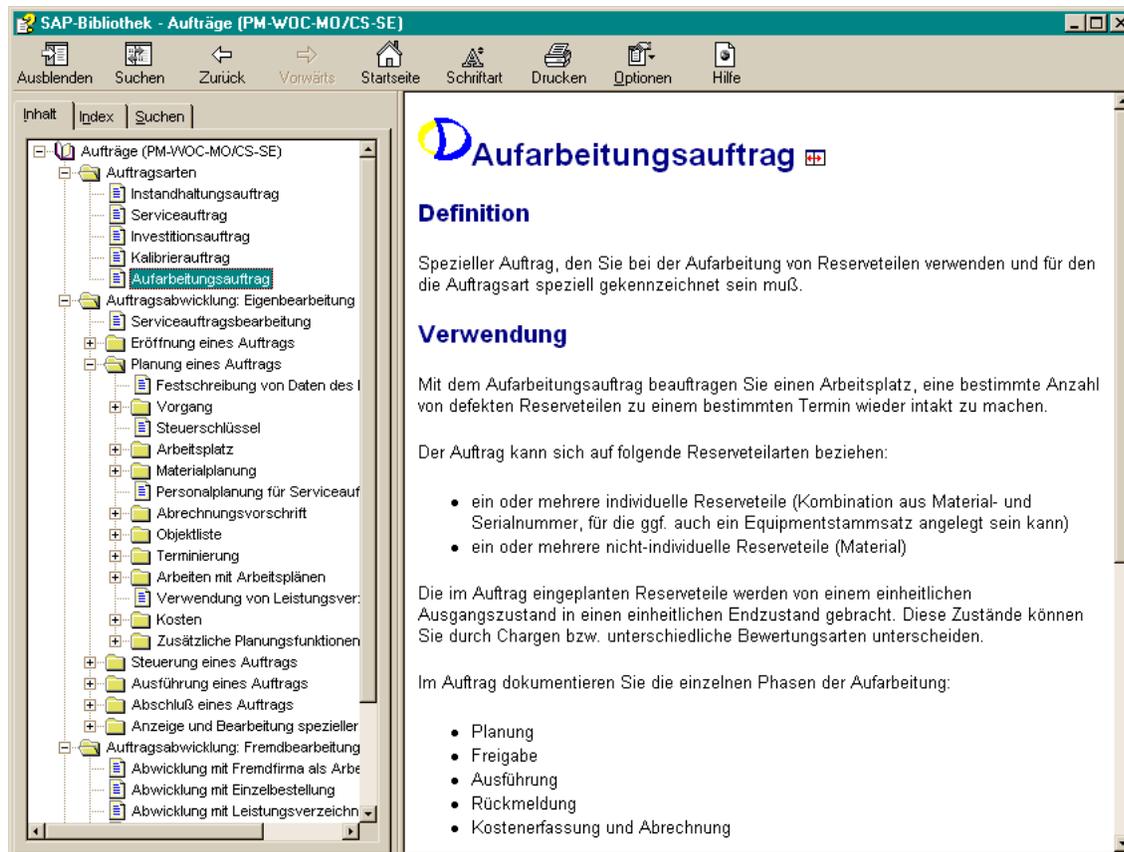


Abbildung 4-7: Online-Dokumentation (Ausschnitt) [SAP00a, PM-WOC-MO/CS-SE]

EINFÜHRUNGSLEITFADEN

Der Einführungsleitfaden (IMG) listet alle Aktivitäten auf, die nötig sind, um die Parameter der SAP-Software so einzustellen, dass die Anforderungen des Unternehmens abgedeckt werden. Der IMG ist ebenfalls an der Komponentenhierarchie orientiert und beschränkt sich hauptsächlich auf eine knappe Implementierungsbeschreibung. Zur Definition einzelner WiPaks liefert der IMG nur wenige Informationen, da er zu technisch ausgerichtet ist. Zur Integration der technischen Aspekte eines Wissenspakets im Rahmen der Einführungswerkzeuge dagegen bildet der IMG die Basis. Abbildung 4-8 zeigt beispielhaft einen Teilbereich des Customizing für den Bereich der Instandhaltung.

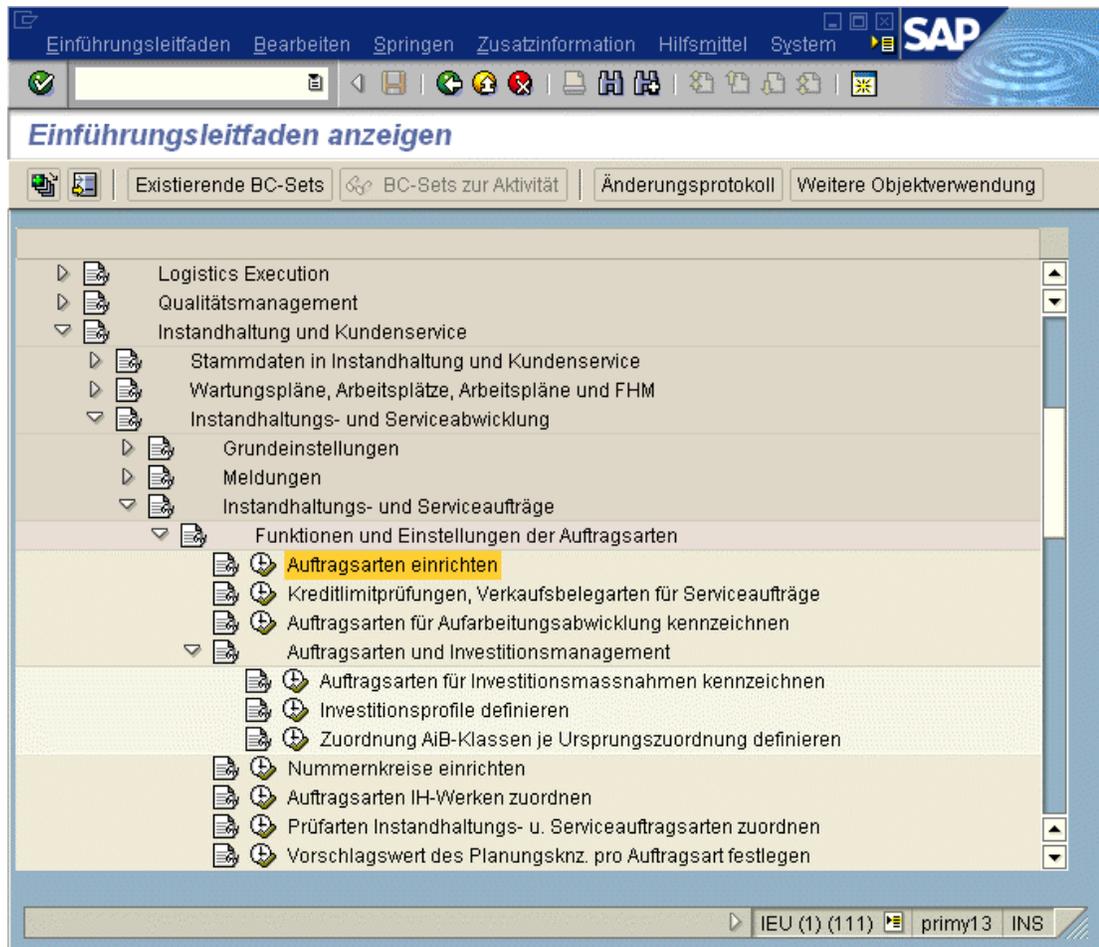


Abbildung 4-8: Einführungsleitfaden des SAP-Systems (Ausschnitt)

EREIGNISPROZESSKETTEN (EPK)

Die ereignisgesteuerten Prozessketten eines Referenzmodells, wie sie beispielsweise in dem von SCHEER entwickelten ARIS-Ansatz verwendet werden [SCHE97], beschreiben detailliert den inhaltlichen und zeitlichen Ablauf von Ereignissen und Funktionen innerhalb eines Geschäftsprozesses.

Die EPK werden ergänzend zur Prozesssicht eingesetzt und definieren einzelne Prozesse genauer. Der Informationsgehalt für die Definition der Wissenspakete ist vergleichbar mit dem der Prozesshierarchie. Vorteil der EPK gegenüber der Prozesshierarchie ist die Visualisierung der Sachverhalte, welche die Identifikation abgeschlossener Einheiten erleichtert. Abbildung 4-9 zeigt einen Ausschnitt aus dem Prozessmodell für den Gesamtprognoselauf des Bereichs Logistik Planung.

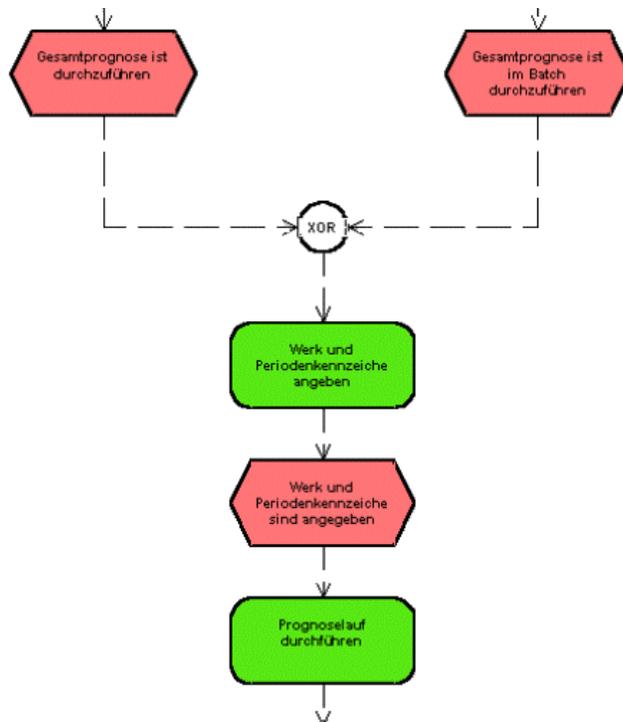


Abbildung 4-9: Ereignis-Prozess-Kette (Ausschnitt)

SOLUTION MAPS

Über die branchenabhängigen Solution Maps werden typische Geschäftsbereiche in einer mehrstufigen Hierarchie abgebildet. Die Solution Maps dienen der Visualisierung, Planung und Umsetzung integrativer Geschäftsprozesse innerhalb eines Unternehmens. Der interessierte Anwender navigiert dabei zunächst auf einer hypertextbasierten Übersichtskarte durch die Hauptprozessbereiche des Unternehmens. Im nächsten Schritt erhält er detaillierte Informationen zu den kritischen Prozessen innerhalb eines bestimmten Geschäftsbereichs [SAP00s]. Abbildung 4-10 zeigt einen Ausschnitt aus der ersten Ebene der Solution Map für das Anwendersegment Versorgungsindustrie (SAP Utilities).

Der Informationsgehalt für die Entwicklung der Wissenspakete ist sehr hoch, da die einzelnen Prozessgruppen bzw. Geschäftsbereiche in sich abgeschlossene Elemente sind. Die bereits vorhandene Hierarchisierung der einzelnen Objekte erleichtert dabei die Zuordnung der Elemente untereinander und gibt wichtige Hinweise auf Zusammenhänge zwischen möglichen Paketen. Solution Maps existieren nur für Branchen, eine generische Map gibt es nicht. Aus diesem Grund können

die Solution Maps vorwiegend zur Definition gewidmeter Wissenspakete herangezogen werden

SAP Utilities - Edition 2000

Enterprise Management	Strategic Enterprise Management		Business Intelligence & Data Warehousing		Managerial Accounting
Customer Relationship Management	Customer Service	Market Research & Analysis	Product / Brand Marketing	Marketing Programs	
Generation	Engineering & Construction			Production	Plant Maintenance
Transmission & Distribution	Engineering & Construction		Operations Management	Maintenance & Work Management	
Installation Services	Sales Cycle & Billing		Connection & Installation Management		Service Management
Energy Service	Sales Cycle	Service Agreement		Special Customer Services	
Energy Trading	Energy Data Management		Traders Workbench	Risk Management	Wholesale

Abbildung 4-10: Solution Map Utilities (Ausschnitt)

SITZMANN hat sich ausführlich mit dem Konzept und dem Aufbau der Solution Maps befasst [SITZ00, S. 74-99]. Er hat insbesondere einen Kriterienkatalog für die Bewertung dieses Ansatzes bezüglich der Eignung als Unterstützungswerkzeug bei der Implementierung einer Softwarebibliothek aufgestellt. Die Beurteilung zeigt, dass die Solution Maps vor allem in den Bereichen Anforderungsdokumentation, Strukturierung und Kontrolle des Adaptionprozesses Schwächen aufweisen, die einen isolierten Einsatz verhindern [SITZ00, S. 82-88].

Die Siebel Systems Inc. bietet ebenfalls typische Geschäftsvorfälle aus dem Bereich des Electronic Business in Form von Solution Maps an [SIEB00]. Diese Maps können vor allem für die Definition von Wissenspaketen aus dem e-Commerce herangezogen werden, da die Geschäftsprozesse softwareneutral gestaltet sind.

KNOWLEDGE MAPS

Knowledge Maps (KM) vereinen die Wissensdarstellung von Text und Grafik, um so basierend auf der Mind Mapping-Methode von TONY BUZAN das geistige Po-

tenzial des Anwenders voll auszuschöpfen [TEUF99, S. 39]. Dabei sind die KM die Anwendung der Mind Mapping-Methode auf betriebswirtschaftliches Wissen bezüglich eines konkreten Sachverhalts. Alle relevanten Themen werden für diesen Sachverhalt visualisiert und strukturiert. Mit Hilfe der KM kann auch bezüglich der Adaption einer betriebswirtschaftlichen Softwarebibliothek eine Wissensbasis aufgebaut werden, die als zentrale Informationsquelle für die Adaptionenprojekte dient. Ähnlich dem hier entwickelten Ansatz der Wissenspakete wird ein zentraler Sachverhalt als Ausgangspunkt für die notwendigen Informationsabfragen bzgl. der Anforderungen an das Softwareprodukt definiert. Die einzelnen Sachverhalte sind über Vorgänger- und Nachfolgerbeziehungen miteinander verknüpft. Im Umfeld der SAP-Software wurden durch die SAP AG seit der Version 4.5B Knowledge Maps auf einer separaten CD ausgeliefert, die über das MindManager-Tool der Firma Marketsoft in das Adaptionenprojekt eingebunden werden können [TEUF99, S. 187]. Die Maps weisen eine hohe Affinität zu den in dieser Arbeit entwickelten Wissenspaketen auf. Die Grundidee der beiden Ansätze ist durchaus vergleichbar, wobei die Wissenspakete einen anderen Projektfokus haben, als die klassischen Knowledge Maps der SAP AG. Die Knowledge Maps sollen in erster Linie die Prozessanalyse als Basis eines zentralen Wissensmanagements unterstützen. Für die Entwicklung der Wissenspakete und die einzelnen betriebswirtschaftlichen Inhalte bieten die Knowledge Maps eine geeignete Grundlage.

Anhang B enthält eine Beschreibung der gepflegten Wissenspakete aus dem Fundamentalmodell für das Anwendersegment der Versorgungsindustrie. Zu jedem WiPak sind die zugeordneten Elemente aus den Werkzeugen LIVE KIT Structure (Parameter-ID) und LIVE KIT Power (Prozessbeleg-ID) aufgelistet. Zusätzlich werden die verknüpften Kalkulationsvariablen zur Berechnung des Aufwands gezeigt. Aus Gründen der Übersichtlichkeit wurde auf die Darstellung weiterer Zuordnungselemente, z. B. der benötigten IMG-Bausteine zur Realisierung des Pakets, verzichtet.

5 THESEUS-Fundamentalmodell

Das THESEUS-Fundamentalmodell besteht grundsätzlich aus den in Kapitel 4.1.2 beschriebenen Stufen, in denen mehrere Monitore zum Einsatz kommen. Das Fundamentalmodell wurde so konzipiert, dass es zur Klärung der folgenden Fragen bezüglich eines Wissenspakets beiträgt.

- Welchen betriebswirtschaftlichen Inhalt besitzt das Paket?
- Welche betriebswirtschaftlichen Anforderungen gibt es bezüglich der Paketausprägung?
- Welche Softwarelösungen werden für die Umsetzung benötigt?
- Wie müssen die Softwarelösungen eingestellt werden?
- Welcher Aufwand entsteht bei der Implementierung?
- Welcher Zusammenhang besteht zu anderen Paketen?

Zunächst wird die Monitorbeschreibung in ihren Grundzügen erklärt, um die Intention des Modells zu vermitteln (5.1). Anschließend werden die monitorübergreifenden Gestaltungsprinzipien erläutert (5.2), bevor dann die einzelnen Monitore im Detail vorgestellt werden (5.3 bis 5.5).

5.1 Aufbau der Monitorbeschreibung

In den folgenden Kapiteln werden die Inhalte und Elemente, die zur Erreichung der Zielsetzungen benötigt werden, erläutert, bevor dann im einzelnen auf die Bestandteile der Monitore eingegangen wird. Ausgehend von der zentralen Fragestellung, wird durch Aufzeigen der Gestaltungsprinzipien und Modellelemente die Intention verdeutlicht. Anhand von Beispielen wird der Aufbau der einzelnen Monitore anschließend anschaulich dargestellt, bevor in Kapitel 6 näher auf die technische Umsetzung eingegangen wird.

5.1.1 Fragestellung

Die einzelnen Monitore sind so gestaltet, dass sie sich jeweils mit einer zentralen Fragestellung befassen. Die Hauptaussage dient der Beantwortung dieser Frage,

daher wird zu Beginn der Beschreibung näher auf die Fragestellung eingegangen und diese explizit aufgeführt, um die Zielsetzung zu verdeutlichen, die der Monitor verfolgt. Die Frage selbst ist dabei dem jeweiligen Kontext angepasst und bezieht sich direkt auf einen speziellen Sachverhalt, zum Beispiel auf das aktuell bearbeitete Wissenspaket. So beginnt der Customizingmonitor für das Paket der Geräteverwaltung z. B. mit der Frage „Welche Customizingeinstellungen müssen für die Abbildung der Geräteverwaltung vorgenommen werden?“.

5.1.2 Gestaltung und Aussage

Dieser Bereich umfasst die betriebswirtschaftlichen Grundlagen für die Zusammenstellung und Darstellung der Monitorinhalte. Dazu werden zunächst die betriebswirtschaftlich relevanten Begriffe erläutert und in das Themenumfeld eingeordnet. Anschließend wird der grundsätzliche Aufbau dargestellt und näher auf die Interaktionsschnittstellen zwischen THESEUS und dem Anwender eingegangen.

5.1.3 Modellelemente

Das Fundamentalmodell der THESEUS-Methode und das daraus spezifizierte Individualmodell in Form der evaluierten bzw. bearbeiteten Wissenspakete werden durch eine eigene Symbolik visualisiert. Die Symbole, die für die einzelnen Monitore entwickelt wurden, sind am Ende der Monitorbeschreibung aufgeführt. Mit Hilfe der durchgängigen und einheitlichen Symbolik wird eine transparente Navigationsumgebung geschaffen, die es dem Anwender ermöglicht, einzelne Sachverhalte bzw. Werkzeugfunktionalitäten monitorübergreifend zu identifizieren und sich somit schnell in die Themenstellung einzuarbeiten. Für die Darstellung der assoziierten Sachverhalte bzw. Elemente aus den zugeordneten Werkzeugen wird die Symbolik verwendet, die in der jeweiligen Realisierungskomponente implementiert ist. So werden zur Darstellung der Prozessbelege aus PENELOPE die Symbole verwendet, die von VOGELANG entwickelt wurden [VOGE97, S. 124], um Verwirrungen und Neuorientierungen des Anwenders zu vermeiden. Auf die Erläuterung der externen Symbole wird in der Monitorbeschreibung verzichtet, da sie nicht elementarer Bestandteil der Betrachtungen sind. Es wird auf die jeweils angegebene Quelle verwiesen, die nähere Einblicke in die Bedeutung und Aussage der verwendeten Symbole liefert.

5.1.4 Beispiel

Als Anschauungsobjekt und für die graphischen Beispiele der einzelnen Monitore wird das Anwendersegment der Versorgungsindustrie herangezogen. Es wurde bewusst ein Bereich ausgewählt, der im Sinne des SPARTA-Konzepts als eigenständiges Segment spezifiziert und in den einzelnen Werkzeugen abgebildet ist, weil dadurch die Einbeziehung der von SCHIPP entwickelten Rasterhypothese [SCHI99, S. 129-138] möglich ist und so die Intention des hier entwickelten Ansatzes anschaulich wird. An dieser Stelle soll nun auf die grundsätzlichen Inhalte des Anwendersegments der Versorgungsindustrie eingegangen werden, da diese zum besseren Verständnis notwendig sind.

Zum Anwendersegment der Versorgungsindustrie (Utilities) zählen alle Unternehmen, die an der Generierung, Übertragung und Verteilung jeglicher Art von Energie beteiligt sind. Neben den klassischen Energiearten (Sparten) Strom und Wasser zählen auch neuere Ressourcen, wie z. B. Wind und Sonne, zu dem hier behandelten Spektrum.

In diesem Bereich werden zunehmend integrierte Gesamtlösungen in Form von betriebswirtschaftlichen Softwarebibliotheken eingesetzt, die um segmentspezifische Funktionalitäten, wie z. B. die Verbrauchsabrechnung oder das Gerätemanagement, erweitert sind. Der Marktführer auf dem Gebiet der betrieblichen Standardanwendungssoftware, die SAP AG aus Walldorf, hat ebenfalls die branchenneutrale R/3-Software um eine Branchenkomponente für die Versorgungsindustrie ergänzt, um den Anforderungen dieses Segments gerecht zu werden. Bezüglich der Adaption einer solchen erweiterten Softwarebibliothek bedeutet dies, dass die bestehenden Vorgehensmodelle an die neuen Inhalte angepasst werden müssen. SCHIPP hat sich im Rahmen der Entwicklung des SPARTA-Konzepts mit dieser Thematik ausführlich befasst [SCHI99, S. 139-155].

Über die Werkzeuge LIVE KIT Structure, LIVE KIT Power und LIVE KIT Control wird die regelbasierte Anforderungsnavigation aus unterschiedlichen Blickwinkeln ermöglicht. Für alle drei Werkzeuge wurden Anpassungen im Sinne der Vorkonfiguration nach dem SPARTA-Konzept vorgenommen [SCHI99, S. 139-153].

5.1.5 Technische Umsetzung

Dieser Teil der Beschreibung bezieht sich auf die technische Implementierung des Fundamentalmodells, speziell auf die Ablage in einer Datenbank. Die Ausführungen dienen dem besseren Verständnis der Zusammenhänge zwischen den einzelnen Monitoren und zwischen den einzelnen Elementen untereinander. Gerade im Hinblick auf die Integration und Zuordnung der heterogenen Werkzeuginhalte ist diese Darstellung unbedingt erforderlich, da sie verdeutlicht, welche Elemente aus dem Datenmodell des zugewiesenen Werkzeugs in Relation zu den Wissenspaketen stehen. Die Zuordnungsdatenbank muss flexibel genug gestaltet sein, um eine asynchrone Releasestrategie zu unterstützen, bei der auf aufwendige Konvertierungsschritte zwischen den betroffenen Datenbanken verzichtet werden kann.

Neben dem zur Abbildung eines Monitors relevanten Ausschnitt aus dem Datenmodell werden die einzelnen Tabellen mit ihren Feldern kurz beschrieben.

Die Darstellung der Teilaspekte des Gesamtdatenmodells wurde über das Datenbankverwaltungsprogramm SQL Server 7.0 der Firma Microsoft realisiert. Dabei wird jeweils der betroffene Ausschnitt aus der Beziehungs-Ansicht dargestellt. Die Ansicht ist auf die für den beschriebenen Monitor relevanten Tabellen beschränkt, wobei alle anderen Tabellen ausgeblendet sind. Das vollständige Datenmodell mit allen Tabellen und Beziehungen ist im Anhang D dargestellt.

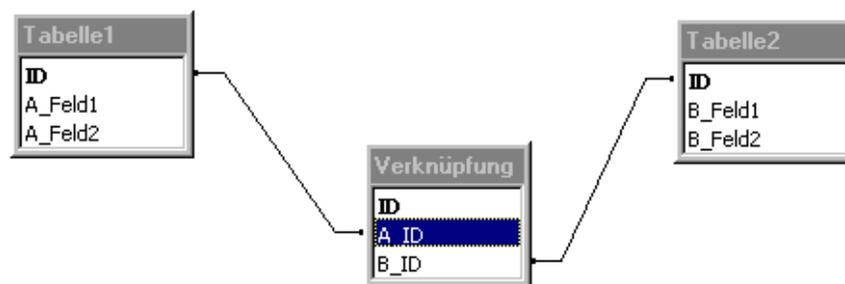


Abbildung 5-1: Darstellungselemente des Monitor-Datenmodells

Abbildung 5-1 zeigt die Bestandteile der Monitor-Datenmodelle. Jedes Rechteck entspricht einer Tabelle, die durch einen eindeutigen Namen identifiziert wird und aus einem oder mehreren Feldern besteht. Die fett gedruckten Felder sind die Schlüssel der Tabellen. Zur Sicherstellung der referentiellen Integrität sind die Tabellen über Beziehungen (meist 1:n-Beziehungen) miteinander verknüpft. Die Dar-

stellung komplexer n:m-Beziehungen wird über zusätzliche Verknüpfungstabellen, die als Hilfskonstruktion dienen, umgesetzt.

Die technische Umsetzung für die einzelnen Monitore ist in Kapitel 6.4.2 detailliert beschrieben.

5.2 Allgemeine Gestaltungsprinzipien

Die transparente und anwenderfreundliche Darstellung der relevanten Informationen zu einem Wissenspaket ist eine der Hauptaufgaben des im Rahmen dieser Arbeit entwickelten Werkzeugs. Zur Umsetzung der THESEUS-Methode wird auf die Monitoring-Technik, die bereits bei der Realisierung von PENELOPE und MENTOR eingesetzt wurde, zurückgegriffen. Zielsetzung der Gestaltung ist die übersichtliche Darstellung vielfältiger Informationen in einem Monitor. Die Workbench ist in mehrere Anzeigesegmente unterteilt. Diese Frame-Technik wird unter anderem in allen gängigen Browsern zur Visualisierung der Inhalte aus dem World Wide Web verwendet und kann als allgemein anerkannter Standard angesehen werden. Mit Hilfe der Frames kann der Anzeigebereich des Browsers in verschiedene, frei definierbare Segmente aufgeteilt werden. Jedes Segment kann eigene Inhalte enthalten und erhöht die Flexibilität bezüglich Navigation und Visualisierung einzelner Sachverhalte. Frames sind über Verweise miteinander verbunden, das bedeutet, dass über einen Link kontextsensitiv Elemente aufgerufen werden können, die dann in einem anderen Segment angezeigt werden. Diese Technik ermöglicht es, die spezifischen Eigenschaften der Bildschirmanzeige konsequent zu nutzen und Informationen hypertextuell (d. h. nicht linear) auszuarbeiten. Abbildung 5-2 zeigt schematisch den Aufbau der Adaption-Workbench zur Umsetzung der THESEUS-Methode.

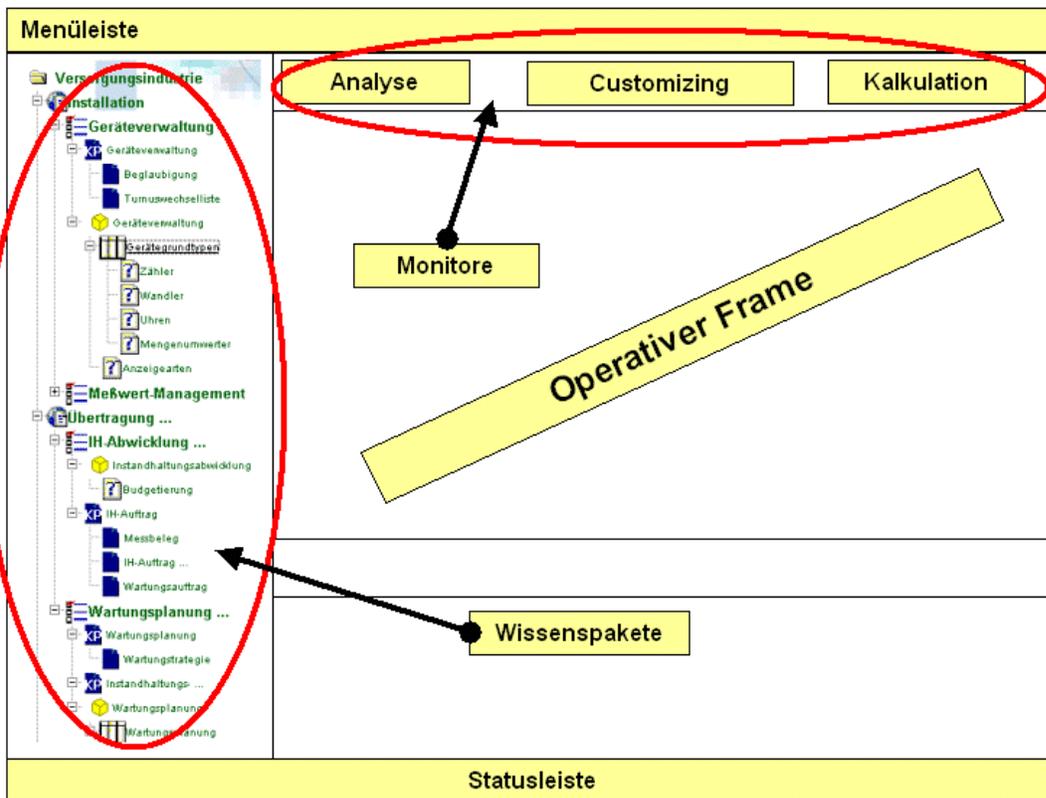


Abbildung 5-2: Schematischer Grundaufbau der Adaption-Workbench

Die gesamte Workbench ist in mehrere Frames unterteilt, deren Inhalt in den folgenden Kapiteln näher dargestellt wird.

NAVIGATIONSBEREICH

Auf der linken Seite der Workbench befindet sich der Navigationsbereich, der monitorübergreifend die Navigation entlang der Wissenspakete ermöglicht. Aufgabe dieses Frames ist die Anzeige der zu bearbeitenden Pakete, die aufgrund der Projektstruktur zur Verfügung stehen. Die Komposition der Projektstruktur und der Pakete folgt dabei den in Kapitel 4.1.3 erläuterten Konventionen.

OPERATIVER FRAME

Auf der rechten Seite befindet sich der operative Frame, dessen Inhalt je nach Monitorauswahl variiert. Ziel dieses Bereichs ist die kontextsensitive und detaillierte Darstellung des zu einem Wissenspaket gehörenden Inhalts, z. B. der Customizingaktivitäten, die zur Implementierung eines Pakets nötig sind. In Abhängigkeit der Monitorauswahl sind einige Frames weiter unterteilt, um

Monitorauswahl sind einige Frames weiter unterteilt, um beispielsweise die zugeordneten HTML-Dokumente anzeigen und bearbeiten zu können.

MENÜLEISTE/STATUSLEISTE

Zur Steuerung und Bearbeitung zusätzlicher Transaktionen, die nicht in direktem Zusammenhang mit einzelnen Wissenspaketen stehen, wird auf die von Microsoft Office-Produkten bekannten Menüleisten zurückgegriffen. Die Verwendung dieser standardisierten Bedienelemente erhöht die intuitive Bedienbarkeit des Werkzeugs und trägt somit maßgeblich zur Akzeptanz bei. Aufgaben der Menüleiste sind u. a.

- die Steuerung der Datenbankverknüpfungen,
- die Lizenzierung,
- die Sprachauswahl und
- die individuelle Gestaltung der Anzeigooptionen (z. B. Auflistung der noch zu bearbeitenden Wissenspakete).

Zur besseren Übersicht und zur leichteren Identifikation wichtiger Informationen befindet sich in der unteren Hälfte des Monitors eine Statuszeile, die beispielsweise Aufschluss über das gerade bearbeitete Paket oder das letzte Änderungsdatum einer Customizingeinstellung gibt. Ebenfalls Gegenstand der Statuszeile sind Informationen über das jeweils zugeordnete Objekt, z. B. die Element-ID eines Kernprozesses aus dem Datenmodell der PENELOPE.

Für die Umsetzung der THESEUS-Methode werden drei Monitore benötigt. Der Analyse-Monitor zeigt die im Rahmen eines Projekts evaluierten Ergebnisse bezüglich eines Wissenspakets aus den jeweils zugeordneten Werkzeugen an. Der Customizing-Monitor stellt die notwendigen Transaktionen und deren Einstellungen dar, die zur Umsetzung eines Pakets nötig sind. Der Kalkulations-Monitor unterstützt den Projektleiter bei der Aufwandsermittlung und -kontrolle. Die Ansteuerung der einzelnen Monitore erfolgt über die Registerkarten-Technik, das bedeutet, dass am oberen Rand des operativen Frames unterschiedliche Kartentitel angezeigt werden, die den Sprung in den dazugehörigen Monitor ermöglichen (vgl. Abbildung 5-2). Der angezeigte Inhalt ist dabei immer anhängig von der Paketauswahl im Navigationsbereich. Ein Wechsel des Pakets hat somit direkt eine Aktualisierung des Monitorinhalts zur Folge.

In den folgenden Kapiteln werden die einzelnen Monitore näher beschrieben. Anhang C enthält zusätzlich Ausschnitte aus dem entwickelten Prototyp zur Umsetzung von THESEUS. Diese Beispiele beziehen sich auf die Abwicklung der Projektverwaltung und die Auswahl geeigneter Datenquellen für die Identifikation der relevanten Wissenspakete.

5.3 Analyse-Monitor

Der Analyse-Monitor soll im Rahmen der themenorientierten Umsetzung folgende Aufgaben erfüllen:

- Aufzeigen der evaluierten Wissenspakete aufgrund einer Rasterhypothese,
- Darstellung der gewidmeten Paketinhalte, die bereits in den vorkonfigurierten Systemen enthalten sind,
- Visualisierung der zugeordneten Werkzeugelemente (z. B. Kernprozesse, Fachbereiche, Reduktionsfragen) und deren Ausprägungen,
- Verdeutlichung der Zusammenhänge zwischen Wissenspaket und Workshopergebnis im Sinne einer Erklärungskomponente sowie
- Darstellung der neu zu bearbeitenden Wissenspakete und deren Inhalt, die aufgrund von Umweltänderungen (z. B. Anforderungswechsel, Softwareupdate) evaluiert wurden bzw. zur Verfügung stehen.

Im Gegensatz zur bisherigen Vorgehensweise muss der Anwender zur Analyse bzw. Anzeige der relevanten Informationen bezüglich eines Themas nicht mehr in verschiedenen Werkzeugen navigieren, sondern konzentriert alle Aspekte über ein Wissenspaket. Die zentrale Fragestellung, zu deren Lösung der Analyse-Monitor beitragen soll, lautet demnach:

Welche Informationen liegen bezüglich eines Wissenspakets vor ?

Die Herkunft dieser Informationen kann dabei sehr heterogen sein. Zum einen sind im Fundamentalmodell gewidmete Informationen aus dem Umfeld der Anwendersegmente hinterlegt, zum anderen wurden Ergebnisse über Workshops der integrierten Werkzeuge (z. B. LIVE KIT Power) ermittelt und in der Projektdatenbank abgelegt. Über die Paketzusammenfassungen der einzelnen Inhalte können die durchzuführenden Änderungen an den Einstellungen der Softwarebibliothek

transparent gemacht werden, ohne dass der Zugriff über mehrere Werkzeuge nötig wird.

Die Aufgaben, die mit Hilfe des Monitors bewältigt werden können, sind vielfältig. Über die Visualisierung der zentralen Themeninhalte ist sowohl die Unterstützung der Softwareeinführung, als auch die Reorganisation bestehender Bibliothekselemente im Sinne des Continuous System Engineering möglich. Der Analyse-Monitor gibt u. a. Antworten auf folgende Fragestellungen:

- Welche Wissenspakete sind zur Umsetzung der Anforderungen nötig ?
- Welche Anforderungen haben sich im Zeitablauf verändert ?
- In welcher Form haben sich die Anforderungen geändert ?
- Welche neuen Möglichkeiten bietet die Softwarebibliothek, d. h. welche zusätzlichen Pakete stehen zu Verfügung ?

5.3.1 Gestaltung des Analyse-Monitors

Der Analyse-Monitor fasst alle Informationen zu einem Wissenspaket zusammen. Im Rahmen dieser Arbeit wird vornehmlich die Integration von THESEUS zu den bereits entwickelten Werkzeugen des ITHAKA-Ansatzes betrachtet. Der Monitor ist jedoch so flexibel, beliebige Inhalte bezüglich eines Pakets zu visualisieren, da für die Darstellung jeweils auf die entwickelte Symbolik und Technik des zugeordneten Werkzeugs zurückgegriffen wird. So können beispielsweise zu einem Wissenspaket neben den Reduktionsfragen des LIVE KIT Structure gleichermaßen die evaluierten Fragen der Question & Answer Database aus dem ASAP-Vorgehensmodell mit den gespeicherten Antworten aufgeführt werden.

Für die Aufgabenstellung, die mit Hilfe des Analyse-Monitors gelöst werden soll, muss die Funktionalität bzw. Darstellung des Navigationsbereichs erweitert werden. Die reine Auflistung des Masterprojektbereichs, der Unternehmensprozessbereiche und der zu bearbeitenden Wissenspakete reicht nicht aus, um zu ermitteln, welche zugeordneten Elemente aus anderen Werkzeugen betroffen sind. Um dies zu erreichen, werden weitere Ebenen unterhalb eines Pakets eingeführt, die beispielsweise aufzeigen, welche Fachbereichskomponenten und Reduktionsfragen aus dem LIVE KIT Structure Informationen zur Ausprägung eines Pakets liefern.

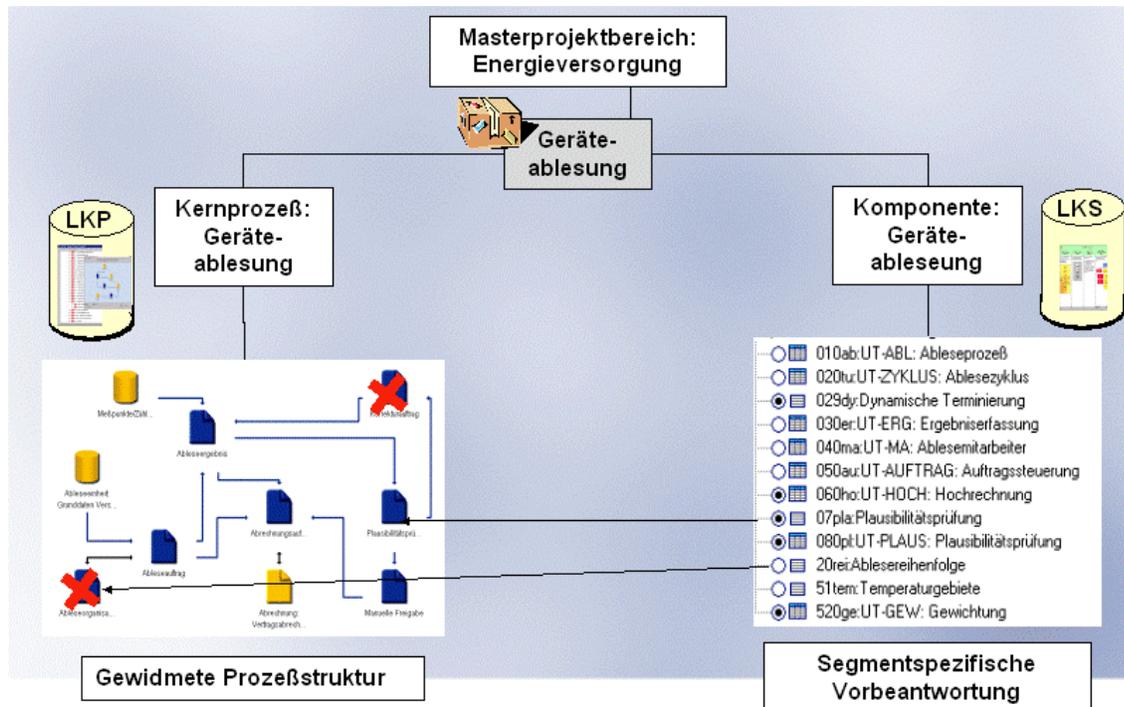


Abbildung 5-3: Detailinformationen zu einem Wissenspaket

Abbildung 5-3 zeigt die zusätzlichen Ebenen am Beispiel des Pakets zur Geräteableseung. Zur Evaluierung der Ausprägung sind Reduktionsfragen aus der Komponente Geräteableseung (LIVE KIT Structure) und der Kernprozess (LIVE KIT Power) konjugiert.

5.3.2 Beispiel eines Analyse-Monitors

Das Beispiel in Abbildung 5-4 zeigt den Analyse-Monitor in einem typischen Einführungsprojekt aus dem Anwendersegment der Versorgungsindustrie. Die hier aufgezeigte Struktur dient auch in den folgenden Kapiteln als Grundlage für die Darstellung des Customizing- bzw. Kalkulationsmonitors.

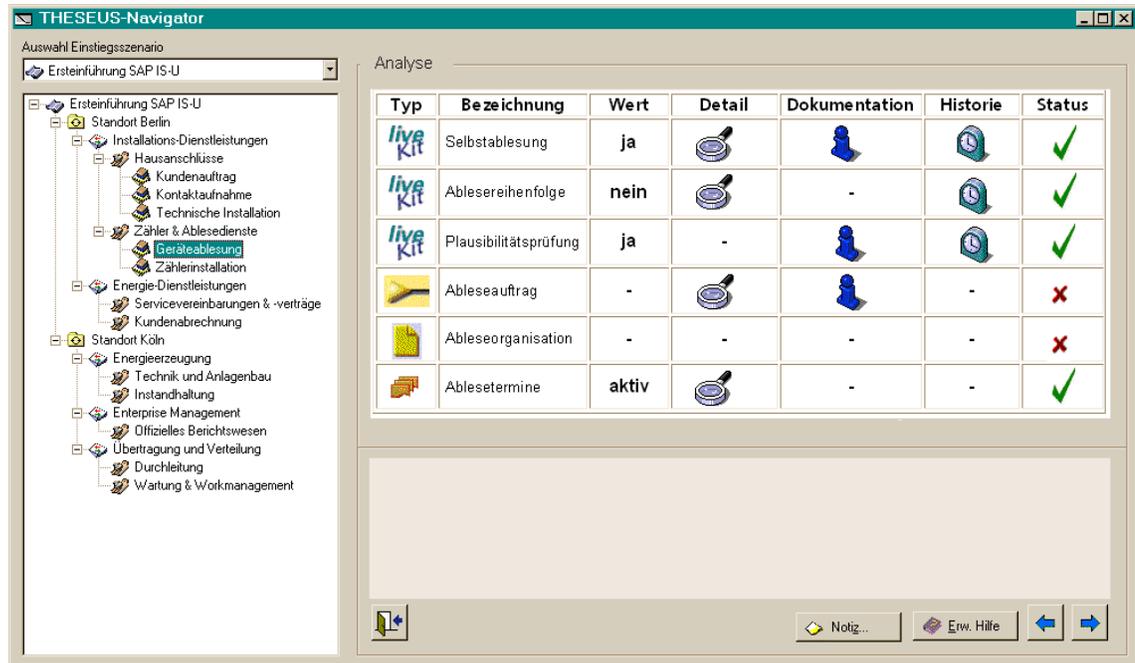


Abbildung 5-4: Beispiel eines Analysemonitors

Über eine Dropdownliste wird das zu bearbeitende Projekt ausgewählt. In dem vorliegenden Beispiel handelt es sich um die Einführung des SAP IS-U-Systems. Nach dem Import der Informationsquellen ist aus den als relevant identifizierten Wissenspaketen eine individuelle Projektstruktur erstellt worden, die im linken Frame als Navigationsumgebung dient. Die operativen Wissenspakete (z. B. Kundenauftrag, Technische Installation und Geräteablesung) bilden jeweils den Abschluss eines Hierarchiebaums.

Zu jedem Wissenspaket (hier: Geräteablesung) sind im rechten Frame die über die Zuordnungstabelle verknüpften Elemente aus den vorgelagerten Adaptionswerkzeugen aufgelistet. Das Beispiel zeigt Elemente aus den Werkzeugen LIVE KIT Structure, LIVE KIT Power und LIVE KIT Control, die jeweils durch eigenständige Symbole gekennzeichnet sind. Zu jedem Element ist in der Statusspalte erkennbar, ob das zugeordnete Element in der zugrundeliegenden Informationsquelle (z. B. power.mdb) identifiziert und analysiert wurde. Im vorliegenden Fall konnte für die LIVE KIT Power Elemente, beispielsweise die Schnittstelle <Ableseauftrag>, kein Wert evaluiert werden, da offensichtlich diese Informationsquelle nicht zur Verfügung stand. Die Elemente aus dem Anforderungsnavigator, z. B. <Selbstablesung>, sind demgegenüber mit einem Wert belegt und angezeigt.

Über die Spalten <Detail> und <Dokumentation> können zusätzliche Informationen zu den jeweiligen Objekten abgefragt werden. Dabei handelt es sich um tiefergehende Informationen des Elements (Detail), beispielsweise die Darstellung des zugehörigen Kernprozesses in Form eines Belegmonitors des LIVE KIT Power oder der Aufruf der entsprechenden Reduktionsfrage aus dem LIVE KIT Structure über den Mediaviewer. Weiterhin kann in die Dokumentationsumgebung des entsprechenden Werkzeugs verzweigt werden, um die dort abgelegten Informationen, z. B. Hilfeaufruf oder Beschreibungstexte, zu nutzen (Dokumentation).

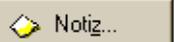
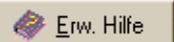
Die Historie liefert einen Überblick über die Wertentwicklung bzw. den Änderungsverlauf des betrachteten Objekts in Bezug auf das aktuell gewählte Wissenspaket. Diese Funktionalität spielt vor allem bei Aufbauprojekten eine Rolle, bei denen mit einer bestehenden Paketstruktur gearbeitet wird.

Zu jedem Wissenspaket besteht die Möglichkeit, eine Notiz in Bezug auf die Analyse zu erfassen und auf diese Weise zusätzliche Informationen zu pflegen. Zudem kann die im Fundamentalmodell hinterlegte Dokumentation über den Inhalt und die Bedeutung eines Wissenspakets aufgerufen werden.

Tabelle 5-1 fasst die Gestaltungselemente zusammen. In den nachfolgend beschriebenen Monitoren wird zum Teil auf die hier erklärten Symbole zurückgegriffen, um die intuitive Bedienbarkeit des Werkzeugs zu erleichtern.

Tabelle 5-1: Gestaltungselemente im Analyse-Monitor

Symbol	Beschreibung
	Das zugeordnete Element ist eine Reduktionsfrage oder ein betriebswirtschaftliches Profil aus dem Anforderungsnavigator LIVE KIT Structure.
	Das zugeordnete Element ist ein Prozessbeleg aus dem Prozessmodellierungswerkzeug LIVE KIT Power.
	Das zugeordnete Element ist eine Schnittstelle aus dem Prozessmodellierungswerkzeug LIVE KIT Power.
	Das zugeordnete Element ist ein Bericht aus dem Berichtsmodellierungswerkzeug LIVE KIT Control.
	Das zugeordnete Element wurde in der eingelesenen Informationsquelle identifiziert. Für dieses Objekt konnte ein Wert evaluiert werden.
	Die Datenquelle enthielt keine Informationen über das zugeordnete Element.

Symbol	Beschreibung
	Zu dem betrachteten Element liegen Detailinformationen aus der Datenquelle vor.
	In der Informationsquelle wurde eine weiterführende Dokumentation zu dem jeweiligen Objekt identifiziert, die aufgerufen werden kann.
	Historieninformationen zeigen die Änderungen bzw. Wertentwicklung des Objekts.
 Notiz...	Notizpflege zu dem aktuell bearbeiteten Wissenspaket.
 Erw. Hilfe	Aufruf der erweiterten Hilfe zu einem Wissenspaket, d. h. Navigation zur hinterlegten Paketdokumentation aus dem Fundamentalmodell.
	Diese beiden Elemente ermöglichen eine schnelle und zielgerichtete Navigation zu dem Vorgänger bzw. Nachfolger des aktuell betrachteten Wissenspakets.
	Dieses Symbol ermöglicht den Ausstieg aus der Transaktion, z. B. Rücksprung in das Hauptmenü des Werkzeugs.

5.4 Customizing-Monitor

Der Analyse-Monitor dient vorwiegend der Visualisierung der gewonnenen Erkenntnisse aus den Workshops zur Spezifikation der Anforderungen eines Unternehmens an die Softwarebibliothek. Dabei werden die verschiedenen Analyseziele durch den Einsatz integrierter Konzepte evaluiert. Die Methoden bzw. die daraus entwickelten Werkzeuge unterstützen aufgrund ihrer Zielsetzungen hauptsächlich die Sammlung von Anforderungen, die ein Unternehmen an die Gestaltung bzw. den Einsatz einer Software besitzt. Die eigentliche Realisierung erfolgt über die Änderung der Systemparameter (Customizing). Der Customizing-Monitor stellt die Verbindung zwischen der Informationsgewinnung und der Realisierungskomponente einer Softwarebibliothek her, indem er aufzeigt und dokumentiert, welche Systemparameter aufgrund der spezifizierten Anforderungen in welcher Weise zu pflegen sind bzw. bereits gepflegt wurden.

Der Customizing-Monitor trägt also wesentlich zur Beantwortung folgender Frage bei:

Welche Systemeinstellungen sind zur Umsetzung eines Wissenspakets nötig?

Eine besondere Bedeutung kommt der richtigen Aufbereitung und Speicherung der Informationen zu. In den bisherigen Ansätzen wurden die notwendigen Systemeinstellungen in Form der zu bearbeitenden Bausteine des Einführungsleitfadens dem Anwender bzw. dem Projektteam nicht transparent genug aufgezeigt und zur Verfügung gestellt. An dieser Stelle soll nochmals kurz auf die Formen der Umsetzungsunterstützung bezüglich der Customizingtransaktionen eingegangen werden, die in den bisherigen Vorgehensmodellen zu finden sind. Die geschilderte Problematik lässt sich an folgenden Punkten verdeutlichen:

1. Aufgrund der Anforderungsanalysen durch unterschiedliche Werkzeuge werden lediglich die notwendigen Customizing-Bausteine identifiziert.
2. Nicht benötigte Bausteine werden in der Softwarebibliothek ausgeblendet, ohne dass deutlich wird, welche Anforderung die Reduktion initiiert hat.
3. Empfehlungen für konkrete Customizingeinstellungen, die vorzunehmen sind, werden nur für die technischen betriebswirtschaftlichen Profile in Form von statischen Dokumentationen angeboten. Eine Übertragung der Voreinstellungen in Form von Tabelleneinträgen (Pre-Customizing) ist nicht möglich.
4. Die Dokumentation der Systemeinstellungen erfolgt nicht datenbankbasiert, sondern wird in statischen Dokumenten oder der Softwarebibliothek selbst vorgenommen.
5. Systemeinstellungen werden meistens direkt am Baustein dokumentiert, ohne dass der Bezug zu einem Thema (z. B. einem Geschäftsprozess) ersichtlich ist. Die Nachvollziehbarkeit kann hierbei nur selten gewährleistet werden.
6. Bei Anforderungsänderungen (z. B. durch Releasewechsel) kann nur auf den Analysewerkzeugen des Vorgängerprojekts aufgesetzt werden, der Rückgriff auf die vorgenommenen Customizingeinstellungen und der sich daraus ergebenden Konsequenzen ist durch den fehlenden Themenbezug der Dokumentation nur schwer durchführbar.

Mit Hilfe des Customizing-Monitors ist es möglich, diese Problematik zu entschärfen und eine transparente Umsetzung der Anforderungen in der Softwarebiblio-

thek anhand der Systemeinstellungen zu gewährleisten. Der Monitor erfüllt zu diesem Zweck mehrere Aufgaben:

- Er zeigt die im Rahmen eines Projekts zu bearbeitenden Wissenspakete.
- Er listet alle Customizingaktivitäten auf, die zur Umsetzung eines konkreten Pakets in der gewünschten Ausprägung nötig sind.
- Er zeigt bei Software- oder Anforderungsänderungen die neu zu bearbeitenden Transaktionen bzw. die neu zur Verfügung stehenden Alternativen auf.
- Er dokumentiert die vorgenommenen Einstellungen immer in Bezug zu einem Wissenspaket.
- Er macht Änderungen über eine Historienverwaltung transparent.
- Er stellt fertige Parameter-Sets zur Verfügung, die typische Lösungen aus dem Einstellungs-Portfolio abbilden und eine automatische Generierung von Baseline-Systemen ermöglichen.

Zentraler Betrachtungsgegenstand ist immer ein konkretes Wissenspaket. Alle Einstellungen bzw. Analysen werden für ein Paket vorgenommen und auch mit Bezug zu diesem dokumentiert. Dadurch können Änderungen leicht nachvollzogen und bewertet werden. An dieser Stelle sei nochmals auf die Vorteile der Themenorientierung verwiesen, die in Kapitel 4.1.1 diskutiert wurden.

5.4.1 Gestaltung des Customizing-Monitors

Ausgangspunkt der Betrachtungen ist in diesem Monitor die Navigationsstruktur im linken Frame der Workbench. Nach der Auswahl eines Wissenspakets werden im operativen Bereich die Elemente angezeigt, die zur Umsetzung des Pakets notwendig sind. Die Darstellung hängt dabei von der zu bedienenden Softwarelösung ab, d. h. im Fall der SAP-Software wird der relevante Ausschnitt aus dem Einführungsleitfaden mit den jeweiligen Customizingtransaktionen angezeigt. Für die Angliederung einer anderen anderen Systems wird auf die entsprechenden Einfüh-

rungswerkzeuge verwiesen. Die folgenden Ausführungen beziehen sich auf die SAP-Software, andere Lösungen werden nicht betrachtet.

Zur Erhöhung der Transparenz werden verschiedene Sichten auf den Einführungsleitfaden angeboten. So ist es zum einen möglich, den gesamten IMG mit allen Bausteinen aufzulisten und die für die Realisierung des Pakets relevanten Transaktionen zu markieren, zum anderen können nur die relevanten Bausteine mit ihren jeweiligen Oberknoten aufgelistet werden. Nicht relevante Bausteine auf gleicher Ebene oder nicht relevante Customizingabschnitte werden ausgeblendet. Die Reduktion auf die wirklich notwendigen Transaktionen beschleunigt die Orientierung des Anwenders und verringert die Gefahr von Falscheingaben. Der Aufbau des Customizingbaums erfolgt dabei dynamisch aufgrund der Informationen, die in der Datenbank zu einem Wissenspaket abgelegt sind. Zu einem Wissenspaket sind in der Zuordnungsebene die notwendigen Bausteine hinterlegt. Bei Aufruf des Monitors werden aus dem im Fundamentalmodell gespeicherten kompletten Einführungsleitfaden die relevanten Elemente anhand der ID herausgefiltert und mit einem Relevanzkennzeichen versehen. Da im Fundamentalmodell mehrere IMG-Versionen parallel gehalten werden können, sind releaseübergreifende Customizingprojekte durchführbar, ohne dass Informationsverluste auftreten oder redundante Einstellungen vorgenommen werden müssen. Zudem werden Deltaanalysen bezüglich des Einführungsleitfadens erleichtert.

Zu jedem Baustein wird ein eigenständiges und versioniertes Dokument in einer Datenbankstruktur abgelegt, das die genaue Beschreibung der vorgenommenen Systemeinstellungen enthält. Die Mehrfachverwendung eines Pakets bzw. eines Customizingbausteins induziert, dass bei der Erfassung einer bausteinspezifischen Dokumentation auf vorhandene Dokumente hingewiesen wird und diese entweder als Referenz oder als Kopiervorlage eingebunden werden können. Durch diese Dokumentationsweise wird es möglich, einen direkten Bezug zwischen den IMG-Bausteinen sowie der jeweiligen Dokumentation herzustellen und dabei redundante Pflegeaktivitäten zu vermeiden. Analog werden die im Fundamentalmodell abgelegten Dokumentationen der betriebswirtschaftlichen Profile bei der Bearbeitung einer Customizingtransaktion als Referenz angeboten.

Betriebswirtschaftliche Profile des Anforderungsnavigators werden als Business Configuration Sets (BC-Sets), d. h. in Form von fertiggestellten Tabellensätzen, im Fundamentalmodell abgelegt und sind einem oder mehreren IMG-Bausteinen zu-

gewiesen. Über das Anforderungsdictionary kann evaluiert werden, welche Profile ausgewählt wurden und somit für die Konfiguration des Baselinesystems relevant sind. Diese BC-Sets werden bei der Bearbeitung eines Wissenspakets identifiziert und über eine Schnittstelle direkt an das System übergeben. Der Aufbau und die Gestaltung von BC-Sets werden in Kapitel 6.3.5 näher beschrieben. Die Dokumentation derartiger Tabellensätze ist ebenfalls im Fundamentalmodell mit Bezug zu den Customizingtransaktionen abgelegt. Die Existenz derartiger Informationen wird am jeweiligen Baustein angezeigt und kann bei dessen Bearbeitung genutzt werden.

Die einzelnen Dokumente werden als HTML-Files in einer festen Verzeichnisstruktur abgelegt, um den Einsatz über das Inter- bzw. Intranet zu gewährleisten. Die Möglichkeiten, die Editoren (z. B. Front Page der Firma Microsoft) dabei bieten, lassen einen uneingeschränkten Einsatz dieser Technologie zu. Über die in der zentralen Datenbank verwalteten Verweise, werden die Dokumente identifiziert und zur Verfügung gestellt.

Jedes Element des Einführungsleitfadens kann mit einer Statusverwaltung überwacht und koordiniert werden. Tabelle 5-2 zeigt die möglichen Status eines IMG-Bausteins.

Tabelle 5-2: Status der Customizingaktivitäten

Offen	Die Customizingaktivität ist als relevant gekennzeichnet, wurde aber noch nicht bearbeitet.
In Bearbeitung	Das Customizing bezüglich dieses Bausteins wurde begonnen, ist aber noch nicht abgeschlossen.
Abgeschlossen	Die Systemeinstellungen wurden bezüglich dieses Baustein vollständig gemäß den ermittelten Anforderungen vorgenommen und sind abgeschlossen.
Zurückgestellt	Die Customizingaktivität ist als relevant gekennzeichnet und wurde bereits in die Planung der Systempflege miteinbezogen. Sie wird jedoch erst zu einem späteren Zeitpunkt bearbeitet.

Die Kennzeichnung des erreichten bzw. vergebenen Status erfolgt durch eine Markierung im Tree des Einführungsleitfadens. Diese Markierung ist nur dann aktiv, wenn das entsprechende Wissenspaket ausgewählt wurde. Die Markierungen bezüglich desselben Bausteins in Verbindung mit einem anderen Paket sind davon unabhängig. Alle Änderungen des Status werden in der Datenbank anhand eines

Zeitstempels unter Angabe des Benutzernamens gespeichert, so dass jederzeit eine vollständige Nachvollziehbarkeit und Projektüberwachung gewährleistet ist.

5.4.2 Beispiel eines Customizing-Monitors

Der in Abbildung 5-5 dargestellte Monitor zeigt einen Auszug der notwendigen Bausteine aus dem Einführungsleitfaden der SAP-Software für die Umsetzung der Geräteablesung. Als Pflegesicht wurde die Abbildung des gesamten IMG mit der Kennzeichnung der relevanten Aktivitäten gewählt.

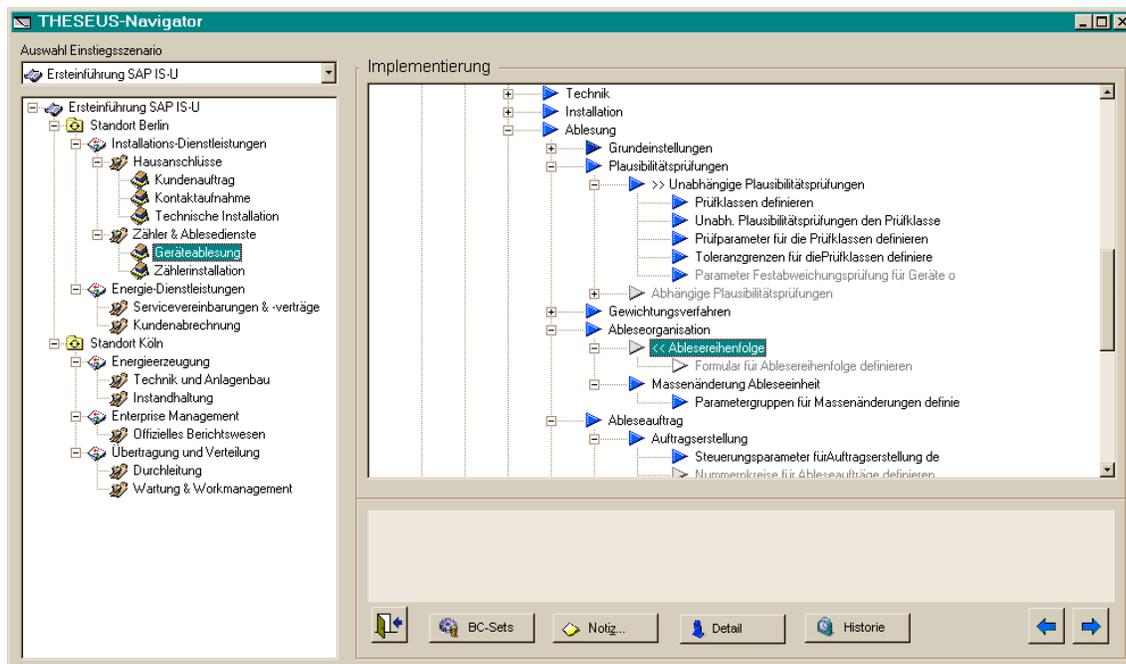


Abbildung 5-5: Beispiel eines Customizing-Monitors

Die hell dargestellten IMG-Bausteine, z. B. <Formular für Ablesereihenfolge definieren>, sind für die Bearbeitung des Pakets nicht relevant. Die Legende in Abbildung 5-6 verdeutlicht die Entscheidungsfindung für das Relevanzkennzeichen. Die Aktivität <Grundeinstellungen> muss beispielsweise verpflichtend durchgeführt werden und stellt eine sogenannte Always-Transaktion dar.

Über ein spezielles Symbol wird die Existenz von BC-Sets angezeigt. Diese Sets können sich zum einen auf das Wissenspaket im ganzen oder auf einen speziellen IMG-Baustein aus dem Fundamentalmodell beziehen. Weitere Dialoge ermögli-

chen die genauere Analyse bzw. Pflege der vordefinierten Customizingeinstellungen.

Die Pflege der projektspezifischen Systemeinstellungen in Bezug zu einem Baustein bzw. einer Systemtabelle wird mit Hilfe eines HTML-Editors durchgeführt. Dabei kann auf Vorlagen zurückgegriffen werden, die ebenfalls im Fundamentalmodell abgelegt sind.

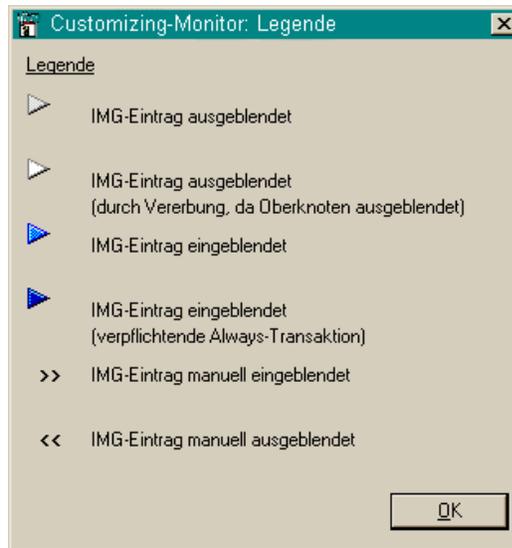


Abbildung 5-6: Legende des Customizing-Monitors

In Tabelle 5-3 sind die Symbole des Customizing-Monitors zusammengefasst. Auf eine Beschreibung der Symbole, die in der gleichen Weise im Analyse-Monitor zum Einsatz kommen, wird an dieser Stelle verzichtet.

Tabelle 5-3: Gestaltungselemente im Customizing-Monitor

Symbol	Beschreibung
 Detail	Aufruf des HTML-Editors zur Pflege der projektspezifischen Dokumentation der System- bzw. Tabelleneinstellungen zu einem IMG-Baustein. Für die Pflege kann in einem weiteren Dialogfenster auf Vorlagedokumente zurückgegriffen werden.
 BC-Sets	Zu dem betrachteten Element existiert ein BC-Set, das für die Vorkonfiguration des Systems genutzt werden kann. Weitere Dialogfenster ermöglichen neben der Nutzung vorhandener Sets auch die Einbindung individueller Sets, um beispielsweise die Mehrfachverwendung von unternehmensspezifischen Systemeinstellungen zu unterstützen.
 Historie	Aufruf der Historie zu einem IMG-Baustein. Die protokollierten Änderungen beziehen sich sowohl auf Statusinformationen als auch auf Modifikationen in den entsprechenden Dokumenten.

5.5 Kalkulations-Monitor

Die Hauptaufgabe des Kalkulations-Monitors ist die Unterstützung der Projektleitung bei der Kostenermittlung und -kontrolle im Rahmen der Softwareadaption. STRELLER hat sich bei der Entwicklung des PANDORA-Konzepts bereits mit der Unterstützung der Projektkalkulation in den einzelnen Vorgehensmodellen, insbesondere ASAP und LIVE METHOD/CHESTRA, befasst und einige Schwächen dieser Ansätze skizziert [STRE99, S. 60-94]. Aus diesen Erkenntnissen wurden Anforderungen abgeleitet, die maßgeblichen Anteil an der Gestaltung und Funktionalität des hier entwickelten Monitors hatten. Zunächst wird noch einmal kurz auf die bisherige Abdeckung der Projektkalkulation eingegangen, bevor dann die Aufgaben und Zielsetzungen des Kalkulations-Monitors näher erläutert werden.

ACCELERATED SAP

Im Vorgehensmodell ASAP der SAP AG werden verschiedene Möglichkeiten der Projektkalkulation unterstützt. Die einfache Variante stellt einen Budgetplan in Form einer Microsoft Excel-Liste dar, der die typischen Elemente eines SAP Projekts beinhaltet. Es stehen vorgefertigte Budgetpläne für Einführungs- und Folgeprojekte zur Verfügung, die je nach Einsatzszenario als Grundlage für die Projektplanung herangezogen werden können. Diese Budgetpläne sind nicht in die anderen Werkzeuge des Vorgehensmodells integriert, d. h. es findet kein Datenaustausch zwischen den einzelnen Dokumenten oder Werkzeugen statt. Der Budgetplan dient lediglich zur Festlegung des allgemeinen Projektmaßstabs. Als weitere Variante ist in ASAP das SAP-interne Werkzeug „Project Estimator“ enthalten, das in der Projektvorbereitungsphase bei der Planung der zu erwartenden Kosten eines Einführungs- bzw. Folgeprojekts hilft [SAP00, SV-ASA]. Dabei finden sowohl der Projektumfang als auch verschiedene Risikofaktoren und die Komplexität der Geschäftsprozesse Berücksichtigung. Mit dem Werkzeug ist es möglich,

- erforderliche externe und interne Ressourcen zu bestimmen,
- den Zeitrahmen der Implementierung abzustecken und
- eine erste Kostenschätzung vorzunehmen.

Auf Grundlage der Ergebnisse können folgende Dokumente erstellt werden:

- Das Umfangsdokument, welches übergeordnete, benutzerdefinierte Sichten des Projektumfangs und die entsprechend festgelegten Werke, Standorte, Vertriebswege und Unternehmenseinheiten enthält.
- Der Projektplan, der sich in Arbeitsplan, Budgetplan und Ressourcenplan untergliedert und während des Einführungsprojekts ständig zu überprüfen und fortzuschreiben ist.

Der Project Estimator ist in die anderen Werkzeuge integriert, so wird beispielsweise der Projektumfang, der im Rahmen der Question & Answer Database festgelegt wird, als Grundlage für die Schätzung des Zeitbedarfs herangezogen. Eine vollständige Unterstützung der Projektkalkulation in dem Sinne, dass einzelne Anforderungsausprägungen den Einführungsaufwand determinieren, ist jedoch nicht gegeben.

LIVE METHOD / CHESTRA

Das Vorgehensmodell der Siemens Business Services GmbH & Co. KG unterstützt die Projektkalkulation durch die Bereitstellung eigener Vorlagen zur Erstellung eines kostenorientierten Projektplans. Dieser Ansatz ist mit den Möglichkeiten aus dem ASAP-Umfeld vergleichbar. Auch in diesem Fall besteht eine nur unzureichende Integration in die übrigen Werkzeuge des Vorgehensmodells. Ein Datenaustausch bzw. eine Übernahme von Kalkulationswerten findet nicht statt. Im Gegensatz zu ASAP verfügt dieses Vorgehensmodell mit dem LIVE KIT Structure jedoch über ein Werkzeug, das die Aufwandsermittlung aktiv unterstützt.

HUFGARD hat im Rahmen der ODYSSEUS-Entwicklung ein Konzept entwickelt, das es ermöglicht, aus den Anforderungen eines Workshops durch Konfigurationsvariablen Aufwandszahlen zu generieren [HUF94]. In diesem Konzept wird für jedes SAP-Modul (z. B. Vertrieb) ein Pool an Dienstleistungsvariablen definiert, die alle Bereiche (z. B. Stammdaten, Serialnummernverwaltung) des Customizing abdecken. Diese Variablen wiederum bestehen aus ein oder mehreren Bausteinen, die in der Regelbasis abgelegt und mit den jeweils betroffenen Reduktions- oder Strategiefragen bzw. betriebswirtschaftlichen Profilen verknüpft sind. Das Zustimmung oder Ablehnen einer Anforderung im Zuge des Workshops steuert den Wert eines Dienstleistungsbausteins. Alle Bausteine zusammen ergeben schließlich den Gesamtwert für eine Dienstleistungsvariable. Über die Auswertungskomponente des Anforderungsnavigators ist es letztlich möglich, den ermittelten Umset-

zungsaufwand pro Modul zu errechnen und als Microsoft Excel-Liste zur Verfügung zu stellen. Abbildung 5-7 zeigt den Aufbau der Aufwandsberechnung im Anforderungsnavigator LIVE KIT Structure. Für das Modul der Versorgungsindustrie (UT) werden in der Detailsicht die einzelnen Dienstleistungsvariablen aufgelistet (DLUT01 bis DLUT50), für die jeweils ein Gewichtungsfaktor aus den Antworten der Analyse ermittelt wurden (Spalte Gewicht). Das Produkt aus dem Gewichtungsfaktor und dem festgelegten Basiswert ergibt schließlich den Aufwand, der zur Umsetzung der Customizingeinstellungen aufgrund der festgestellten Anforderungen notwendig ist.

R/3 Modul	Aufwand (in PT)									Aufwand [h]	
	SBS	Kunde	DL-Schlüssel	Basis [h]	Gewicht	Richtwert [h]	Faktor		SBS	Kunde	
MM Materialwirtschaft	Einführungspaket										
VWM Lagerwirtschaft											
PP Produktionsplanung	UT-Versorgungsindustrie		DLUT			37			112,90	0,00	
PI Prozessindustrie	Grundfunktionen		DLUT01	4	1,35	0 min.	1,00		5,40	0,00	
PM Instandhaltung	Portionierung und Terminierung		DLUT02	4	0,5	0 min.	1,00		2,00	0,00	
PS Projektsystem	Stammdaten IS-U		DLUT03	10	1,25	0	1,00		12,50	0,00	
QM Qualitätsmanagement	Technische Geräteverwaltung		DLUT04	20	1	0	1,00		20,00	0,00	
SD Vertrieb	Geräteinstallation		DLUT05	8	0,25	2	1,00		2,00	0,00	
FI Finanzbuchhaltung	Geräteablesung		DLUT06	20	1,75	35 min.	1,00		35,00	0,00	
BK Banking	Geräteprüfung		DLUT07	12	0,5	0 min.	1,00		6,00	0,00	
CO Controlling	Vertragsabrechnung		DLUT08	20	0,75	0	1,00		15,00	0,00	
PC Produktkosten Controlling	Fakturierung IS-U		DLUT09	12	1,25	0	1,00		15,00	0,00	
AA Anlagewirtschaft	Kundenservice		DLUT10	10	0	0 min.	1,00		0,00	0,00	
IM Investitionsmanagement	Informationssystem		DLUT50	24	0	0	1,00		0,00	0,00	
TR Treasury	6,06	0,00									
FM Haushaltsmanagement	0,00	0,00									
TM Treasurymanagement	0,00	0,00									
HR Personalwirtschaft	14,25	0,00									
RT Retail	0,00	0,00									
UT Versorgungsindustrie	14,11	0,00									
VK Vertragskontokorrent	0,00	0,00									
Sonderaufwand	0,00	0,00									
Summe	161,77	0,00									

Gewichtungsfaktoren aus Anforderungsanalyse

Abbildung 5-7: Aufwandsermittlung des LIVE KIT Structure

Aus der Abbildung ist zu erkennen, dass sich die Aufwandszahlen in diesem Modell lediglich auf die Durchführung der notwendigen Customizingeinstellungen pro SAP-Modul beziehen. In der Aufstellung nicht enthalten sind zusätzliche Aufwände, die beispielsweise für die Pflege der Stammdaten oder die Berücksichtigung von Fremdsystemen entstehen. Zudem werden die Aufwände pro SAP-Modul ermittelt und ausgewiesen, eine Zuordnung zu übergreifenden Themenstellungen ist nicht möglich. Aufgrund der starken Fokussierung auf das Customizing und die mangelnde Erklärungskomponente ist der Aussagegehalt für eine transparente Projektkalkulation eher gering. Die Aufwandsberechnung kann in dieser Form lediglich als Basis der Kalkulation dienen.

5.5.1 Gestaltung des Kalkulations-Monitors

Aus den Ausführungen der vorangegangenen Kapitel wird deutlich, dass eine rein modulatorientierte Aufwandsbestimmung keine echte Grundlage für eine Unterstützung der Projektkalkulation bildet. Gerade die Orientierung an funktionsübergreifenden Wissenspaketen ermöglicht eine detaillierte und umfassende Aufwandsermittlung auf Basis der evaluierten Anforderungen. Der Kalkulationsmonitor stellt alle Funktionalitäten zur Verfügung, die eine transparente und adaptierbare Kalkulation zulassen. Ein großes Problem war bisher die statische Verwaltung und Ablage der Informationen in verschiedenen Tabellenkalkulationsprogrammen. Eine lückenlose und nachvollziehbare Änderungsanalyse war nur bedingt möglich. Mit Hilfe des hier beschriebenen Monitors werden die Aufwandsbestandteile eines Wissenspakets einzeln unter Berücksichtigung von hierarchischen Abhängigkeiten dargestellt.

Für die Ermittlung des Aufwands können verschiedene Daten als Informationsquellen herangezogen werden. Über die Zuordnungsebene von THESEUS werden die einzelnen Bestandteile analog zur Verknüpfung der Werkzeugelemente zugeordnet. Somit ist auch die hier abgebildete Aufwandsberechnung offen gegenüber unterschiedlichen Informationsquellen. Zu einem Paket werden im operativen Frame die zugewiesenen Elemente mit dem aus den Informationsquellen ermittelten Werten aufgelistet. Zusätzlich hat der Anwender die Möglichkeit, neue Aufwandszeilen manuell einzufügen und somit nicht evaluierte Bestandteile (z. B. Schulungsbedarf) zu berücksichtigen. Für bestehende Elemente kann der Anwender die ermittelten Werte ändern und über eine Notizfunktion zusätzliche Informationen hinterlegen. Die Existenz einer Notiz wird wiederum über eine Markierung gekennzeichnet. Die Darstellung der einzelnen Zeilen ist hierarchisch gegliedert und ermöglicht so einen schnellen Überblick. Die erste Ebene bildet der betroffene Fachbereich (z. B. UT), darunter befinden sich die Hauptknoten der Aufwandskalkulation (z. B. UT01, UT02). Zu jedem dieser Hauptknoten werden auf der untersten Stufe die einzelnen Entitäten angezeigt, deren Wertermittlung aus den Informationsquellen möglich war. Die Neuordnung von Aufwandselementen durch den Anwender ist auf jeder Stufe möglich und wird in der Datenbank abgelegt.

Für eine effektive Projektkontrolle ist es unabdingbar den Projektfortschritt und die Einhaltung des zur Verfügung stehenden Budgets zu überwachen. In der bisherigen Form war es nicht möglich, neben den ermittelten Soll-Aufwandszahlen die tatsächlich angefallenen Kosten für die Umsetzung im Rahmen des Customizing zu speichern und den Planwerten gegenüberzustellen. THESEUS erlaubt zu jedem Oberknoten (z. B. UT01, UT02) die Erfassung des Ist-Aufwands. Somit kann über eine Differenzspalte kritisch analysiert werden, welcher Anteil des prognostizierten Aufwands bereits abgedeckt ist bzw. wie viel Restaufwand noch zu erwarten ist.

Die Statusvergabe und Änderungsverfolgung wird in Analogie zum Customizingmonitor durchgeführt.

5.5.2 Beispiel eines Kalkulations-Monitors

Für die Geräteablesung wurden im Fundamentalmodell drei Dienstleistungskomponenten gepflegt (DLUT02, DLUT03 und DLUT05), die im Kalkulationsmonitor sequentiell aufgelistet werden (Abbildung 5-8). Jede Dienstleistungsvariable besteht wiederum aus mehreren Bestandteilen, die im unteren Frame dargestellt sind. Im vorliegenden Beispiel unterteilt sich die Aufwandsberechnung der Portionierung (DLUT02) in die Komponenten Sperrgründe (DLUT02_01), Außendienststeuerung (DLUT02_02) und Reihenfolgeplanung (DLUT02_03).

Kalkulation

Schlüssel	Bezeichnung	Plan (MT)	Ist (MT)	Differenz	Status
DLUT02	Portionierung	2,5	-	0,5	✓
DLUT03	Terminsteuerung	1,5	-	1,5	✓
DLUT05	Ablesesteuerung	2	-	2	✓

Einzelwerte

Schlüssel	Bezeichnung	Basis	Gewicht	Faktor	Aufwand (Std)
DLUT02_01	Sperrgründe	8	0,5	1	4
DLUT02_02	Außendienststeuerung	10	1	1	10
DLUT02_03	Reihenfolgeplanung	4	1,5	1	6

Abbildung 5-8: Beispiel eines Kalkulations-Monitors

Die importierten Kennzahlen werden zu jeder Variable in der Spalte Gewicht angezeigt. Die Berechnung erfolgt durch die Multiplikation des Basiswerts mit dem Aufwandsgewicht und einem beliebigen Faktor, der durch den Projektleiter zur individuellen Steuerung der Kalkulationsbasis vergeben wird. Die Spalte <Aufwand (Std)> zeigt pro Komponente das Ergebnis der Multiplikation an. Im oberen Frame werden die auf diese Weise ermittelten Aufwandszahlen für jede Dienstleistungsvariable in Summe ausgewiesen und als Planwert in die Kalkulation übernommen (Spalte <Plan (MT)>). Da die Abrechnungen und Kalkulationen in Adaptionsprojekten in der Regel in der Einheit Manntage durchgeführt werden, erfolgt eine Umrechnung der Stundensummen der einzelnen Komponenten in diese Einheit. Im vorliegenden Beispiel wurde ein Umrechnungsfaktor von acht zugrunde gelegt, dieser Faktor kann durch den Projektleiter geändert werden. Für das gezeigte Beispiel errechnet sich demnach der Planwert für die Portionierung (DLUT02) aus folgender Formel:

$$\text{Plan (MT)} = (4 + 10 + 6) / 8 = 2,5 \text{ (MT)}$$

Für die Umsetzung der Portionierung wird aufgrund der Anforderungsanalyse mit einem Aufwand von 2,5 Manntagen kalkuliert. Für das komplette Wissenspaket ergibt sich ein Planwert von 6 Manntagen.

Zur Erfassung der tatsächlich benötigten Manntage im Rahmen der Realisierung eines Wissenspakets steht dem Projektleiter die Spalte <IST (MT)> zur Verfügung. Zu jeder Dienstleistungsvariable besteht die Möglichkeit, einen Istwert zu erfassen und somit einen Abgleich mit den geplanten Aufwandszahlen durchzuführen. Die Spalte <Differenz> zeigt die Abweichungen bzw. das noch nicht ausgeschöpfte Zeitkontingent auf.

Analog zur Statusanzeige im Analyse-Monitor wird für jede Dienstleistungsvariable über ein Symbol angezeigt, ob zu den in der Zuordnungstabelle hinterlegten Aufwandsvariablen aus der Datenquelle Informationen gelesen werden konnten oder nicht.

Als weitere Manipulationsfunktionen stehen die in Tabelle 5-4 aufgeführten Möglichkeiten zur Verfügung.

Tabelle 5-4: Gestaltungselemente im Kalkulations-Monitor

Symbol	Beschreibung
 Neu...	Anlegen einer neuen projektspezifischen Aufwandskomponente, die nicht im Fundamentalmodell enthalten, aber für die Projektabwicklung notwendig ist. Diese Funktion steht sowohl auf der Ebene der Dienstleistungsvariablen, als auch der einzelnen Variablenbestandteile zur Verfügung und ermöglicht dem Projektleiter die Erweiterung der Projektkalkulation. Die Einträge werden mit direktem Bezug zu einem Projekt und einem Wissenspaket vorgenommen.
 Ändern...	Die dargestellten Aufwandselemente können mit Historienkennzeichen geändert werden, sofern der Anwender die benötigte Berechtigung besitzt.
 Historie	Über die Historienfunktion ist es möglich, alle Manipulationen bzw. Veränderungen an den Aufwandskennzahlen eines Wissenspakets nachzuvollziehen. Neben dem Benutzer und dem Änderungsdatum wird zusätzlich die inhaltliche Manipulation angezeigt.
 Berechnen	Nach der Durchführung von Manipulationen bzw. Erweiterungen erfolgt eine Neuberechnung der Kalkulationswerte.
 Stundenzahl ...	Zur Umrechnung der Stundensummen in die Projekteinheit Manntage kann über diese Funktion der Faktor angepasst werden. Der Defaultwert ist acht Stunden pro Manntag.

6 THESEUS-Umsetzung

Inhalt dieses Kapitels ist die Darstellung der inhaltlichen und funktionalen Anforderungen an das Modellierungswerkzeug (6.1 und 6.2) und daran anschließend die Erläuterung der Anpassungsmethodik des Fundamentalmodells bis zur Ausgestaltung des Individualmodells (6.3). Zudem wird vorgestellt, wie auf Basis des Individualmodells ein lauffähiges Baseline-System erstellt und im erforderlichen Maß dokumentiert werden kann. Eine besondere Bedeutung kommt dabei der automatischen Systemkonfiguration durch die Business Configuration Sets (BC-Sets) zu, die in einem eigenen Kapitel ausführlich dargestellt wird (6.3.5). Anschließend werden Möglichkeiten der technischen Umsetzung diskutiert (6.4). Insbesondere die Integration der beteiligten Werkzeuge wird in diesem Zusammenhang detailliert betrachtet.

6.1 Die Adaptions-Workbench als Component Ware

Integrierte Vorgehensmodelle bzw. Adaptionswerkzeuge weisen eine hohe Komplexität auf, da sie die verschiedenen Phasen eines Implementierungsprojekts miteinander verknüpfen. Um den spezifischen Anforderungen der Unternehmen gerecht werden zu können, bedarf es einer flexiblen und adaptierbaren Systemarchitektur, welche die Anpassung der Werkzeuge an die unternehmensspezifischen Anforderungen ermöglicht. Die Flexibilität schließt zwei Dimensionen ein:

1. Flexibilität bezüglich der Anpassbarkeit an die Softwarelandschaft des betrachteten Unternehmens und
2. Flexibilität bezüglich der Änderbarkeit der abgelegten Informationen. Dieser Aspekt umfasst die Struktur- und Verhaltensanpassung des Werkzeugs. Unter Strukturanpassung wird in diesem Zusammenhang die Veränderung der Systemstruktur des betrachteten Unternehmens verstanden, etwa durch Ausweitung der Geschäftsfelder und der damit verbundenen zusätzlichen Softwarelösungen. Verhaltensanpassung zielt auf die stetige Verbesserung des Informations- und Organisationsniveaus innerhalb des Unternehmens, z. B. durch Auswahl weiterer Softwarefunktionalität, im Sinne des Continuous System Engineering.

Eine Möglichkeit zur Realisierung dieser Flexibilität ist die Modularisierung des Werkzeugs bzw. der Inhalte, wie sie in der THESEUS-Methode beschrieben wurde (Kapitel 4.1). In der Literatur wird für derartige Werkzeuge bzw. Lösungen, die aus verschiedenen Bausteinen zusammengesetzt sind, der Begriff Component Ware verwendet (z. B. [KELL98, S. 62]). Die Idee hinter diesen Anwendungssystemen ist mit der Grundidee von THESEUS zumindest auf inhaltlicher Ebene vergleichbar, so dass die Aufgaben und Anforderungen dieses Ansatzes auch bei THESEUS Anwendung finden können. Generell sind mit der Entwicklung des hier vorgestellten Werkzeugs folgende Grundaufgaben verbunden:

- **Auffinden von Wissenspaketen**
Zur Wiederverwendung der Wissenspakete ist es erforderlich, dass komfortable Suchmechanismen zur Verfügung stehen, die eine Identifizierung der einzelnen Pakete anhand von Attributen ermöglicht.
- **Komposition der Projektstruktur**
Komposition ist die Montage von Wissenspaketen zu einer fertigen Projektstruktur. Dabei sind die isoliert bereitgestellten Pakete in einer sinnvollen Form zu kombinieren, integrieren, testen und dokumentieren.
- **Granularität der Wissenspakete**
Prinzipiell ist die Bildung von Wissenspaketen in unterschiedlichen Granularitäten möglich. Zum Beispiel kann ein Paket ein kompletter Geschäftsprozess oder eine einzelne Funktion sein. Die Wahl des Abstraktionsgrades ist entscheidend für die Einsetzbarkeit dieses Kompositionsansatzes. Der Granularität wurde bei der Konzeption von THESEUS durch die Festlegung der Eigenschaften und des Charakters eines Pakets Rechnung getragen (vgl. Kapitel 4.3).

Die Bildung der Wissenspakete muss verschiedenen Grundanforderungen genügen, damit die Zusammensetzung zu integrierten Projektszenarien sowie die Anbindung an die Softwarelösung möglich ist. Neben den spezifischen Anforderungen für die Gestaltung der Wissenspakete (vgl. Kapitel 4.3) werden im folgenden die Aspekte Wiederverwendbarkeit, Erweiterbarkeit und Anpassbarkeit betrachtet. Diese Eigenschaften spielen vor allem bei der Gestaltung des Werkzeugs und der Komposition der projektspezifischen Struktur eine große Rolle, wobei der Schwerpunkt auf der funktionalen Definition der Workbench liegt. Im Kontext der

Kopplung von Fundamentalmodell und Anwendungssystem spielen diese Eigenschaften auch bei der Bildung der Wissenspakete eine zentrale Rolle.

WIEDERVERWENDBARKEIT

Der Begriff Wiederverwendbarkeit beschreibt die Eigenschaft von Wissenspaketen, bei der Erstellung des kundenindividuellen Systems teilweise oder vollständig verwendet werden zu können. Diese Eigenschaft ist grundlegend für die hier beschriebene Konzeption, da die Modelle als wiederverwendbare Paketsammlungen eingesetzt werden sollen. Diese Sammlungen entstehen durch die Komposition der Projektstruktur und die damit verbundene Eingliederung der einzelnen Wissenspakete. Eine wesentliche Voraussetzung für die Wiederverwendung ist die Auffindbarkeit der Pakete. Da für die Kopplung an die Softwarelösung des Kunden keine vollständige Neumodellierung von Wissenspaketen durch den Anwender sinnvoll ist, muss die Ablage der vorgefertigten Pakete so erfolgen, dass der Anwender die benötigten Bausteine auffinden bzw. identifizieren kann. In diesem Zusammenhang spielen die Attribute und Zuordnungen eines Wissenspakets die zentrale Rolle. Über eine variablenbasierte Konfiguration des Fundamentalmodells können die relevanten Wissenspakete erkannt und zur Verfügung gestellt werden. Diese Funktionalität muss durch das Werkzeug in jedem Fall unterstützt werden.

ANPASSBARKEIT

Die Wissenspakete sind elementarer Bestandteil des Fundamentalmodells und werden durch den Werkzeuganbieter bereitgestellt. Die postulierte Offenheit des hier entwickelten Konzepts induziert, dass die Wissenspakete an die Erfordernisse des Anwenders angepasst werden können. Hier werden zwei Formen der Anpassung unterschieden: Die Modifikation bestehender und das Hinzufügen neuer Pakete (evtl. unter Nutzung vorhandener Bausteine). Diese Anpassungen können nicht frei vorgenommen, sondern müssen über bereitgestelltes Beziehungswissen und Sicherungsmechanismen kontrolliert werden.

ERWEITERBARKEIT

Die Erweiterbarkeit stellt eine spezielle Form der Modellanpassung dar, in der den einzelnen Wissenspaketen unter Beibehaltung der betriebswirtschaftlichen Bedeutung zusätzliche Elemente hinzugefügt werden. Es können beliebige Änderungen an den Paketen zugelassen werden, was eine größtmögliche Flexibilität der Wissenspakete gewährleistet, aber gleichzeitig die Konsistenz des Anwendungssystems

gefährdet. Somit gilt es, einen geeigneten Kompromiss zwischen maximaler Flexibilität und maximaler Konsistenz zu finden.

Für die Individualisierung der Wissenspakete ist die Vollständigkeit die zentrale Eigenschaft, da sie eine wichtige Voraussetzung für die Kopplung an die Softwarelösungen des Unternehmens ist. Diese Vollständigkeit kann in drei Dimensionen betrachtet werden:

- *Bezogen auf den betriebswirtschaftlichen Inhalt*
Wissenspakete beschreiben eine abgeschlossene Teilaufgabe eines Anwendungssystems. Alle Funktionen, die fachlich zu dieser Teilaufgabe gehören, müssen in diesem Paket enthalten sein. Des Weiteren muss in jedem Wissenspaket angegeben sein, welche betriebswirtschaftlich sinnvollen Vorgänger- und Nachfolgerpakete vorhanden sind. Da die Anforderung bezüglich des betriebswirtschaftlichen Inhalts nicht formalisierbar sind, kann die Vollständigkeit des Projektmodells nicht analytisch validiert werden.
- *Bezogen auf die Zuordnung der Analysewerkzeuge*
Ein Wissenspaket ist demnach vollständig, wenn alle Informationsträger aus dem Bereich der Softwareadaption, die logisch zu der im Paket enthaltenen Teilaufgabe gehören, zugeordnet und bewertet wurden. Ein in diesem Sinne vollständiges Wissenspaket ist unerlässlich für die lückenlose Umsetzung der Analyseergebnisse und die umfassende Abdeckung der ermittelten Anforderungen des Anwenders. Welche Zuordnungen zu treffen sind, hängt von den eingesetzten Werkzeugen und der betrachteten Softwarelandschaft ab.
- *Bezogen auf die verschiedenen Customizingmöglichkeiten des Zielsystems*
Wissenspakete dienen im Rahmen der Softwareadaption als Grundlage für die Parametrisierung des Anwendungssystems. Somit müssen alle alternativen Customizingaktivitäten zur Realisierung des betrachteten Pakets in seiner individuellen Ausprägung enthalten sein. Nur so kann das Customizing auf Basis der Modifikation der Pakete erfolgen.

Die vorangegangenen Ausführungen machen deutlich, dass das hier entwickelte Werkzeug nicht nur die reine Visualisierung der Analyseergebnisse bzw. die Do-

kumentation der vorgenommenen Systemeinstellungen unterstützen muss. Vielmehr muss das Werkzeug in der Lage sein, über eine Engineering-Komponente umfangreiche Modifikationen am Fundamentalmodell zu gestatten und in den Gesamtkontext der Werkzeugfunktionalitäten einzugliedern. Somit stellt THESEUS eine umfassende Arbeitsumgebung für die Realisierungsphase einer Softwareadaptation bereit, die dem Anwender viele Freiräume bei der Gestaltung der Umsetzungsaktivitäten zusichert und gleichzeitig flexibel einsetzbar ist.

6.2 Komponenten der Workbench

Die Realisierung einer Adaption-Workbench ist aufgrund der vielfältigen Anforderungen an das Werkzeug ein komplexes Vorhaben. Um dieses zu erleichtern, sollte die Entwicklung stufenweise erfolgen. Dieses Vorgehen hat zudem den Vorteil, dass Teile des Werkzeugs bereits eingesetzt werden können, auch wenn noch nicht der gesamte Funktionsumfang verfügbar ist.

THESEUS besteht aus den folgenden Komponenten:

- Monitor,
- Navigator und
- Engineer.

6.2.1.1 MONITOR

Die erste Stufe der Realisierung wird als Monitor bezeichnet. Diese zeichnet sich durch folgende Eigenschaften aus:

- **Fundamentalmodell:**
Mit diesem Modell sollen mögliche Wissenspakete aufgezeigt werden.
- **Anpassende Referenzmodellierung:**
Das Fundamentalmodell kann durch vorhandene Informationen der zugeordneten Werkzeuge projektspezifisch konfiguriert werden.
- **Adaptionsmonitoring:**
Berichte und Auswertungen zu einzelnen Wissenspaketen ermöglichen eine Analyse und Bewertung.

- **Manuelle Anpassung:**
Die Wissenspakete des Fundamentalmodells können manuell aktiviert bzw. deaktiviert werden.

6.2.1.2 NAVIGATOR

Ein Navigator verfügt über einen höheren Grad an Integration zu anderen Werkzeugen und der Standardanwendungssoftware:

- **Flexible Navigationsstruktur:**
Die projektspezifische Komposition der Navigationsstruktur ermöglicht eine flexible Koordination der Realisierungsaktivitäten.
- **Rückwärtsintegration:**
Durch eine enge Anbindung an vorgelagerte Werkzeuge können die Ursachen für Abwahlentscheidungen oder Systemeinstellungen analysiert werden. Die Entscheidungen werden somit dort geändert, wo diese vorgenommen wurden. Dies bedeutet, dass die Konsistenz des Modells über alle Werkzeuge hinweg gewährleistet bleibt.
- **Vorwärtsintegration:**
Mit der Integration zur Softwarelösung können die Ergebnisse an den Wissenspaketen bereitgestellt und vorhandene Informationen in das Modell übernommen werden.

Ein Anforderungsnavigator ist ein Analysewerkzeug in Form einer speziellen Wissensbasis mit einem betriebswirtschaftlich ausgerichteten Leitfaden für eine heuristische Evaluierung. Im Leitfaden werden sowohl die Beschreibung von Betriebsabläufen als auch das Potenzial der Softwarebibliothek berücksichtigt [THOM96, S. 165]. THESEUS ist ein Werkzeug mit einer projektspezifischen Wissensbasis, das relevante Pakete und die Möglichkeiten der Softwarelösungen nutzt. THESEUS verwendet die Paketauswahl, Reduktion und Zuordnungslogik und schlägt eine Sammlung notwendiger Aktivitäten für die Umsetzungsphase vor. Aufgrund dieser Affinitäten kann THESEUS - in Analogie zum betriebswirtschaftlichen Anforderungsnavigator - als Wissenspaketnavigator bezeichnet werden.

6.2.1.3 ENGINEER

Der Engineer ist die funktional mächtigste Realisierungsstufe. Mit diesem ist die aktive Modellierungskomponente verfügbar:

- Individualmodell:
Im Individualmodell können eigene Wissenspakete instanziiert und zugeordnet werden. Zudem ist es möglich, die unternehmensindividuelle Softwarelandschaft durch Erweiterungen in der Zuordnungsebene zu berücksichtigen.
- Gestaltungshilfen:
Konsistenzprüfungen und Restriktionsanalysen unterstützen die Paketgestaltung und -entwicklung.

6.3 Methodische Umsetzung

Der praktische Nutzen von THESEUS beschränkt sich nicht nur auf die Realisierungsphase einer Softwareadaption, wobei der Fokus eindeutig in diesem Bereich liegt. Vielmehr besteht durch die neutrale und flexible Workbench die Möglichkeit, bisher vernachlässigte Teillösungen mit in die Gesamtprojektierung einzubeziehen. Die alleinige Darstellung der Funktionalitäten einer Softwarelösung in Form eines Fundamentalmodells ist dabei nicht genug. Dieses passive Modell kann zwar grundsätzlich zur Umsetzung der Anforderungen eingesetzt werden, eine kundenindividuelle Darstellung bzw. Bearbeitung ist aber nicht möglich.

Der Vorteil von THESEUS liegt dagegen in der Möglichkeit der kundenspezifischen Widmung des Modells. Diese anpassende Modellierung beginnt mit der Kombination des Fundamentalmodells und den Informationen aus den vorgelagerten Werkzeugen der Analysephase. Aus den übergebenen Informationen werden regelbasierte Parameter abgeleitet, die zur Konfiguration des Individualmodells dienen. Die Identifikation der relevanten Wissenspakete erfolgt dabei durch den Zugriff auf verschiedene Individualmodelle oder Data Dictionaries der interagierenden Werkzeuge. Über die Zuordnungsebene werden dann die jeweiligen Pakete aktiviert oder deaktiviert.

6.3.1 Fundamentalmodell

Das THESEUS Fundamentalmodell (vgl. Kapitel 5) kann auch ohne die Anbindung an die vorgelagerten Werkzeuge und die Konfigurationsmöglichkeiten, die sich daraus ergeben, als eigenständiges Realisierungswerkzeug genutzt werden. In diesem Fall steht mit dem unmodifizierten Modell eine Abbild der Standardsoftware zur Verfügung, das nach Wissenspaketen sortiert ist. Einzig die Erklärungs-komponente, die den direkten Zugriff auf die zugeordneten Werkzeuginhalte ermöglicht, kann nicht genutzt werden. Die Komposition der Projektinstanzen und der zu bearbeitenden Wissenspakete kann dennoch ohne Einschränkung durchgeführt werden. Das Modell ist dabei so flexibel gestaltet, dass eine spätere Berücksichtigung eventuell durchgeführter Analyseworkshops ohne Konsistenz- und/oder Informationsverlust gewährleistet ist.

6.3.2 Entwicklung des Individualmodells

Die kundenspezifische Widmung wird durch die Adaptionmethoden der Auswahl und Reduktion erreicht. Die Wissenspakete werden auf Basis gewonnener Informationen einer strukturierten Analyse mit unterstützenden Werkzeugen konfiguriert. Im Rahmen der Workshops zur Anforderungsanalyse bzw. Prozessmodellierung werden kundenindividuelle Anforderungen an das Zielsystem formuliert und diese Informationen in Form eines Individualmodells bzw. eines Anforderungsdictionaries strukturiert abgelegt. Ebenso verhält es sich bei der Durchführung einer Rasterhypothese, die dem SPARTA-Konzept von SCHIPP entstammt [SCH199]. Die Merkmalsausprägungen werden strukturiert gespeichert und können von anderen Werkzeugen aufgenommen werden. Die Wissenspakete sind über die Zuordnungsebene mit verschiedenen Elementen aus den angesprochenen Werkzeugen verknüpft, so dass ein Zugriff auf die hinterlegten Informationen die Aktivierung bzw. Deaktivierung einzelner Wissenspakete zulässt. Die Entscheidung darüber, ob ein Paket letztendlich für die Bearbeitung in einem Projekt relevant ist, wird durch Prüffälle getroffen. Im Rahmen einer solchen aktiven gewidmeten Konfiguration ist eine Kommunikationsgrundlage geschaffen, die Konsistenzprüfungen von Seiten des Anwenders ermöglicht und Missverständnisse weitestgehend vermeidet. Das Adaption-Monitoring hilft auch, Anforderungen an die Adaption zu verifizieren und gegebenenfalls neu zu formulieren. Somit steigt die Qualität des

Baseline Systems. Damit ist THESEUS eine Erweiterung der integrierten Vorgehensweise zur kundenspezifischen Adaption von betriebswirtschaftlichen Softwarebibliotheken (ITHAKA).

6.3.3 Anpassende Modellierung

Theseus liefert mit den ausgeprägten Monitoren ein Fundamentalmodell, das eine übersichtliche konzeptionelle Darstellung der adaptionsrelevanten Informationen in Bezug auf ein umzusetzendes Wissenspaket ermöglicht. Die enthaltenen Informationen sind, ähnlich dem Leistungsumfang einer komplexen Softwarebibliothek, weitaus mehr, als das Unternehmen sinnvoll nutzen kann. Daher gilt es, mit den kundenspezifischen Informationen aus bereits durchgeführten Analysetätigkeiten die Inhalte des THESEUS-Fundamentalmodells soweit zu reduzieren, dass nur noch die Wissenspakete übrig bleiben, die für das betrachtete Unternehmen von Belang sind. Diese Methode der anpassenden Modellierung ermöglicht die einfache Reduktion der Komplexität und die Identifikation des Relevanten. Das so entstehende Individualmodell ist dann fertiggestellt, wenn kein Wissenspaket mehr weggelassen werden kann. In dieser Stufe wird der volle Nutzen von THESEUS ersichtlich, der darin besteht, dass auf einem sehr hohen Detaillierungsgrad nur noch die unternehmensindividuellen sowie realisierungsrelevanten Wissenspakete dargestellt und bearbeitet werden.

Nach der Individualisierung des allgemeinen Modells gilt es im nächsten Schritt die Projektstruktur festzulegen, um einzelne Arbeitsaufgaben für das Projektteam bei der Umsetzung zu definieren. THESEUS als flexible Navigationsumgebung gibt keine feste Strukturierung der einzelnen Realisierungsaktivitäten vor, d. h. der Freiheitsgrad ist in diesem Fall sehr hoch. Lediglich die Vorgänger-/Nachfolgereigenschaft der Wissenspakete determiniert einige Aufgabenfolgen. Durch die Kapselung der Pakete wird es möglich, diese an beliebigen Stellen der Projektstruktur einzubauen und so den Gesamtkontext der Realisierung zu bestimmen. Die eigentlichen Aktivitäten, wie z. B. die Aufwandsermittlung oder die Dokumentation der Systemeinstellungen, wird auf Ebene der Wissenspakete durchgeführt. Alle anderen Stufen der Projekthierarchie dienen lediglich der transparenten Koordination und Kontrolle. Abbildung 6-1 zeigt beispielhaft eine frei definierte Projektstruktur mit den identifizierten Wissenspaketen.

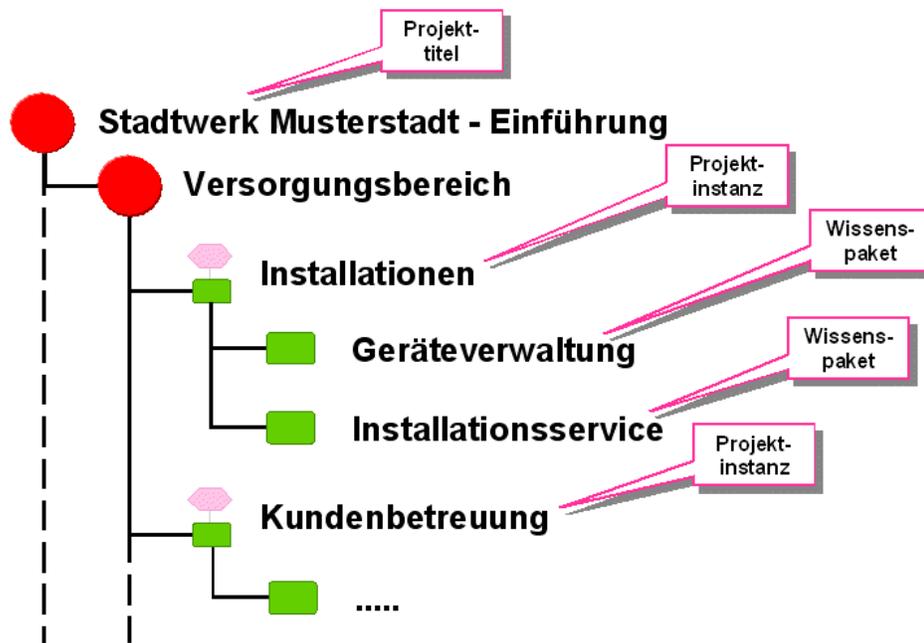


Abbildung 6-1: Projektstruktur in der Adaption-Workbench (Beispiel)

Für das Stadtwerk Musterstadt wurde eine mehrstufige Projektstruktur festgelegt, über die sämtliche Realisierungstätigkeiten im Rahmen der Softwareeinführung abgewickelt werden. Dabei wird der Versorgungsbereich separat betrachtet. Dieser Bereich gliedert sich wiederum in mehrere Projektinstanzen (z. B. Installationen, Kundenbetreuung), an deren unteren Ende jeweils die umzusetzenden Wissenspakete, z. B. die Geräteverwaltung oder der Installationservice, stehen. Dieses Beispiel zeigt die Kompositionsmöglichkeiten von THESEUS. Jede andere Strukturierung der Aktivitäten ist denkbar, lediglich die Auswahl der Pakete ist durch die vorangegangenen Analysephasen vorgegeben. Im weiteren Projektverlauf wird mit den einzelnen Wissenspaketen weitergearbeitet.

6.3.4 Aufwandsermittlung

An die anpassende Modellierung schließt sich die Bestimmung des Realisierungsaufwands, der zur Umsetzung eines identifizierten Wissenspakets gemäß den ermittelten Anforderungen nötig ist, an. Zum besseren Verständnis wird zunächst der Aufbau und die Logik der Aufwandsberechnung erläutert.

Jedes WiPak vereint mehrere unterschiedliche betriebswirtschaftliche (Teil-) Aspekte, die gemeinsam implementiert werden. Für jeden dieser Aspekte sind gewisse

Aufwendungen während der Realisierungsphase nötig, die von der Ausprägung des behandelten Themas gemäß den Unternehmensanforderungen abhängen. Über die verschiedenen Analysewerkzeuge werden die Adaptioneninhalte spezifiziert und stehen als Informationsquelle für die Berechnung des Konfigurationsaufwands zur Verfügung, d. h. durch den Zugriff auf die Auswertungsdateien der eingesetzten Werkzeuge ist es möglich, für die verschiedenen Bestandteile eines Wissenspakets den Aufwand zu quantifizieren und als Grundlage für die Projektkalkulation zu verwenden.

In der Zuordnungsebene werden alle Bausteine der externen Werkzeuge, die Informationen für die Berechnung liefern, einem Wissenspaket zugeordnet. Dabei folgt die Vergabe der Bausteinschlüssel jeweils den Konventionen des Quell-Werkzeugs, um eine eindeutige Zuordnung der Informationen zu gewährleisten und unnötige Konvertierungsschritte zu vermeiden. Aus den Informationsquellen werden dann die benötigten Kennzahlen gelesen und über die Berechnungsalgorithmen für den Anwender als Aufwandswert aufbereitet.

Die Berechnung setzt sich folgendermaßen zusammen: Jedem Wissenspaket werden eine oder mehrere Dienstleistungsvariablen (DLV) zugeordnet (z. B. DLPM22 für die Abbildung der Anlagenstruktur aus dem Bereich der Instandhaltung). Diese Variablen bestehen wiederum aus einem oder mehreren Bestandteilen, die über die Schlüsselkennung eindeutig zuordenbar sind. Der errechnete Aufwand ist jeweils die Summe der eingehenden Bestandteile auf jeder Ebene. Für jede DLV setzt sich die Aufwandsberechnung aus einem Basiswert und einem Gewichtungsfaktor zusammen. Der Basiswert ist der Aufwand, der für die Umsetzung eines Wissenspakets in der Standardausprägung veranschlagt wird. Dieser Wert ist in der Datenbank in Abhängigkeit des Wissenspakets abgelegt. Über die Anforderungsanalyse werden Zu- bzw. Abschläge bezüglich dieser DLV ermittelt, die aus der Auswahl bzw. Nichtauswahl optionaler Funktionalitäten der Softwarebibliothek resultieren. Die Ablage der Information erfolgt in Form eines Gewichtungsfaktors pro Bestandteil der DLV. Ein Beispiel soll diesen Zusammenhang verdeutlichen. Für die Anlagenstruktur kann wahlweise mit den Standardkennzeichen gearbeitet werden oder es können kundenindividuelle Kennzeichen eingeführt werden. Bei der Wahl kundenindividueller Kennzeichen erhöht sich der Implementierungsaufwand für diese DLV entsprechend. Die Beantwortung der zugehörigen Reduktionsfrage induziert, dass die betrachtete DLV für die Anlagenstrukturierung um einen be-

stimmten Faktor (z. B. 0,25) erhöht werden muss. Alle der Variablen zugeordneten Gewichtungsfaktoren werden nach Abschluss der Analysephase summiert und mit dem hinterlegten Basiswert multipliziert. Somit ergibt sich ein errechneter Aufwandswert bezüglich dieser DLV und dem betrachteten Wissenspaket. Die Summe der einzelnen Dienstleistungsvariablen wiederum beziffern den Gesamtaufwand zur Einführung des Pakets in der gewünschten Ausprägung.

Abbildung 6-2 zeigt exemplarisch die Berechnung der DLV für die Anlagenstrukturierung aus den ermittelten Anforderungen. Als Basiswert wurden in diesem Beispiel 12 Stunden veranschlagt, der Gesamtaufwand errechnet sich durch die Auswahl zusätzlicher Referenzplätze (+0,25) sowie den Verzicht auf eigene Strukturkennzeichen und Objektnetze mit insgesamt 21 Stunden für die Systemkonfiguration.

$$DLPM22 = (0,5+1+0,25) * 12 = 21$$

DLPM22 = (0,5+1+0,25) * 12 = 21					
DLPM22 Anlagenstruktur		MINIMUM 0,5 Nur Technische Plätze	+0,25 STRUKTURKZ + Neue Strukturkennzeichen	EQUIPMENT 1 Objektorientierte Anlagenstruktur	+0,25 REFERENZPLATZ + Referenzplätze +0,5 OBJEKTNETZ + Darstellung der Objektverbindungen

Abbildung 6-2: Aufwandsberechnung THESEUS

Aus den hinterlegten Werten des Fundamentalmodells werden somit durch den Zugriff auf die Auswertungsdateien der zugeordneten Werkzeuge dynamisch die Aufwandszahlen berechnet und dem Anwender zur weiteren Bearbeitung angezeigt. Ähnlich der Reduktion der Prozessbelege beim Übergang vom Fundamental- zum Individualmodell von PENELOPE sind auch für die einzelnen Dienstleistungsvariablen die enthaltenen Bausteine mit den ermittelten Kennzahlen ersicht- lich und bei einer vorausgegangen Abwahl als nicht relevant gekennzeichnet.

Im Anschluss an diese automatische Berechnung des Aufwands werden folgende Schritte durch THESEUS unterstützt:

- **Änderung der Paketbestandteile:**
Es ist denkbar, dass im Rahmen der Aufwandsermittlung bestimmte Elemente versehentlich reduziert bzw. nicht reduziert wurden. In diesen Fällen können einzelne Elemente manuell zu- bzw. abgewählt werden.
- **Änderung der Gewichtungsfaktoren:**
Die automatisch errechneten Gewichtungsfaktoren können bei Bedarf geändert werden, um beispielsweise außerordentliche Projektumstände in die Kalkulation mit einfließen zu lassen.
- **Erweiterung der Dienstleistungsvariablen:**
Nicht berücksichtigte Aufwandsgrößen können mit Bezug zu einem Wissenspaket ergänzt und mit den entsprechenden Werten belegt werden. Somit kann auch Aufwand berücksichtigt werden, der nicht unmittelbar Gegenstand der Anforderungsanalyse ist. Ein Beispiel hierfür ist die Erfassung des Aufwands für verschiedene Testläufe des Baselinesystems.
- **Erfassung des Ist-Aufwand:**
Die automatisch ermittelten Aufwandswerte stellen die Planwerte der Realisierungsphase dar. Zur genauen Projektkontrolle ist es zwingend erforderlich, dass neben diesen Planzahlen auch die tatsächlich benötigten Aufwandswerte pro Wissenspaket nach Abschluss der Systemkonfiguration hinterlegt und analysiert werden, um beispielsweise die Qualität bzw. Güte der Kalkulationsgrundlage sukzessive durch Projekterfahrungen zu verbessern.

6.3.5 Baselineing

Unter Baselineing wird der Prozess zur Erstellung eines Baseline-Systems verstanden. Das Baseline-System ist eine Eröffnungslösung, die schnell realisierbar und im Unternehmen einzuführen ist. Dieses System enthält genau die Anforderungen, die unbedingt erfüllt sein müssen, und solche, die als eindeutig lösbar identifiziert wurden. Während der Implementierung und Integration dieser Eröffnungslösung werden bereits Anforderungen deutlich, die in eine verbesserte Folgelösung einfließen können. Im ersten Schritt liegt der Fokus auf den kritischen Anforderungen.

Die Möglichkeiten für das Baselining durch THESEUS hängen zum großen Teil von der gewählten Softwareumgebung ab. Nur wenn offene Strukturen und Schnittstellen einen direkten Zugriff auf einzelne Parameter bzw. Tabellen erlauben, kann das Customizing aktiv unterstützt werden. Diese Voraussetzung ist beispielsweise bei der SAP-Software gegeben. Bei anderen Lösungen beschränkt sich der Nutzen auf die flexible Dokumentation der Systemeinstellungen mit Hilfe des Customizing-Monitors, eine direkte Konfiguration ist dann nicht realisierbar.

Im folgenden wird die Methode zur Unterstützung des Baselining am Beispiel der Softwarebibliothek R/3 vorgestellt, da sie die Grundlage für die Entwicklung des THESEUS-Prototypen ist. Die Implementierungsmöglichkeiten haben sich seit Einführung der SAP-Software ständig weiterentwickelt und erreichen mit der Version 4.6C und der darin eingesetzten Technik der Business Configuration Sets (BC-Set) eine neue Dimension der Konfiguration, die es erlaubt mit Hilfe externer Werkzeuge eine automatische Systemkonfiguration vorzunehmen. THESEUS nutzt diese Art der Vorkonfiguration, um die Realisierungsphase zu beschleunigen und die Qualität der Adaption durch die Vermeidung von Fehlern bei der Umsetzung der Ergebnisse zu verbessern. Im folgenden werden zunächst die Möglichkeiten der BC-Sets aufgezeigt, bevor dann am Ende des Kapitels die Verwendung dieser Technik im Zusammenhang mit den Wissenspaketen dargestellt wird.

BC-Sets dienen zur Dokumentation und Analyse der Systemeinstellungen, wodurch das Customizing transparenter wird. Ebenso können BC-Sets für einen Konzern-Rollout genutzt werden, wobei die Customizingeinstellungen von einer Konzernmutter an ihre Tochtergesellschaften gebündelt und strukturiert weitergereicht werden. BC-Sets werden von SAP AG für ausgewählte Branchen ausgeliefert und können darüber hinaus vom Kunden selbst erstellt werden [SAP00, BC-CUS].

Beim Anlegen eines BC-Sets werden Werte und Wertekombinationen aus den ursprünglichen Customizingtabellen in das BC-Set kopiert und können im Baseline-System in die Tabellen, Views und Viewcluster eingespielt werden.

Mit BC-Sets werden Tabellenwerte, die einen bestimmten Bereich des Customizing abbilden, gruppiert. So können z. B. die Customizingeinstellungen zu einzelnen Wissenspaketen in einem BC-Set abgebildet werden. Dabei ist die Zusammenstellung der Tabellenwerte beliebig. Es ist möglich, die Werte einer gesamten Tabelle

oder auch einzelne Felder aus einer oder mehreren Tabellen zu einem BC-Set zusammenfügen. Darüber hinaus können mehrere BC-Sets zu einem übergeordneten Set zusammengefasst werden.

Grundsätzlich gibt es zwei Arten von BC-Sets:

1. Einfache BC-Sets (im folgenden auch nur "BC-Sets" genannt) und
2. Hierarchische BC-Sets.

Ein BC-Set wird über folgende Attribute charakterisiert:

- Name des BC-Sets,
- Typ des BC-Sets (einfaches oder hierarchisches BC-Set),
- Releasestand sowie
- letzter Änderer, Änderungsdatum und -uhrzeit.

EINFACHES BC-SET

Ein BC-Set enthält ausgewählte Daten aus beliebig vielen Customizing-Tabellen. Die Daten werden anhand einer Auswahl von Tabellenzeilen und -spalten zusammengestellt, wobei eine Zeile einem Datensatz und eine Spalte einem Tabellenfeld entspricht. Der Umfang eines BC-Sets ist beliebig. Es kann Daten zu einer einzigen oder zu mehreren IMG-Aktivitäten beinhalten. Jede IMG-Aktivität wird bei der Anzeige gesondert mit den zugehörigen Customizing-Objekten (Tabellen bzw. Views) aufgelistet (Abbildung 6-3).

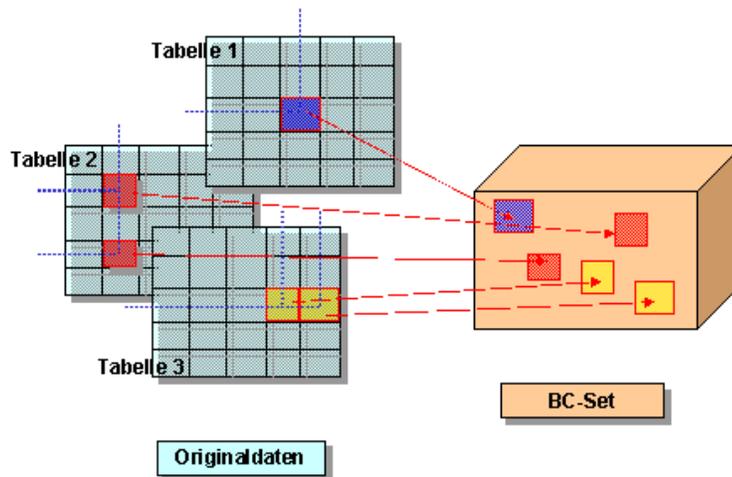


Abbildung 6-3: BC-Set (einfach) [SAP00, BC-CUS]

HIERARCHISCHES BC-SET

Ein hierarchisches BC-Set setzt sich aus mehreren BC-Sets zusammen. Die hierarchische Gliederung kann beliebig fortgesetzt werden. Sie dient bei komplexen Systemeinstellungen zur Strukturierung der Daten. Untergeordnete BC-Sets können jederzeit gelöscht oder neu hinzugefügt werden (Abbildung 6-4).

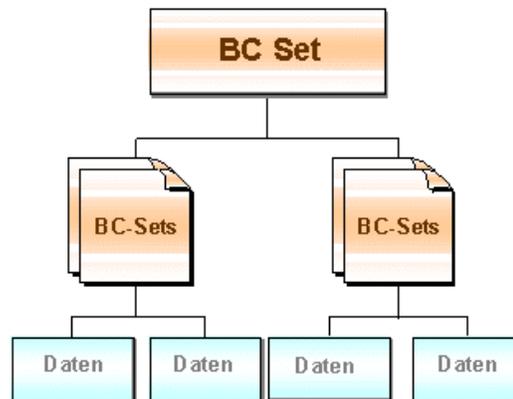


Abbildung 6-4: BC-Set (hierarchisch) [SAP00, BC-CUS]

THESEUS nutzt vorwiegend einfache BC-Sets zur Abbildung der benötigten Systemeinstellungen, eine Öffnung des Ansatzes gegenüber hierarchischen Sets oder den von der SAP AG ausgelieferten Sets für die Branchenkomponenten ist in der dreistufigen Architektur über die unabhängige Zuordnungsebene bereits

dreistufigen Architektur über die unabhängige Zuordnungsebene bereits berücksichtigt.

Der Anforderungsnavigator LIVE KIT Structure beinhaltet neben den Reduktions- bzw. Strategief Fragen bereits vorgefertigte Lösungen, die den Funktionsumfang und insbesondere die verschiedenen Alternativlösungen aus betriebswirtschaftlicher Sicht widerspiegeln. Zu diesen sogenannten betriebswirtschaftlichen Profilen wird eine Dokumentation erstellt, welche die gewählten Profile pro Fachbereich zusammenstellt und Hinweise für die vorzunehmenden Tabelleneinträge enthält. An dieser Stelle sei nochmals auf die Nachteile dieser Lösung hingewiesen, die sich unter folgenden Punkten zusammenfassen lassen:

- Die Dokumentation ist nicht fachbereichsübergreifend, d. h. wichtige Integrationsaspekte gehen verloren.
- Die Dokumentation ist statisch in Form von einzelnen MS Word Dokumenten, eine dynamische Anpassung ist nur bedingt möglich.
- Die Inhalte der Dokumente sind nur mit erheblichem Aufwand anpassbar. Vor allem bei Änderungen der betroffenen Customizingtransaktionen entsteht ein hoher Anpassungsaufwand.
- Es gibt keine Konsistenzprüfung bezüglich der vorgenommenen Einstellungen, da die Ergebnisse lediglich als Einzeldokumente vorliegen.

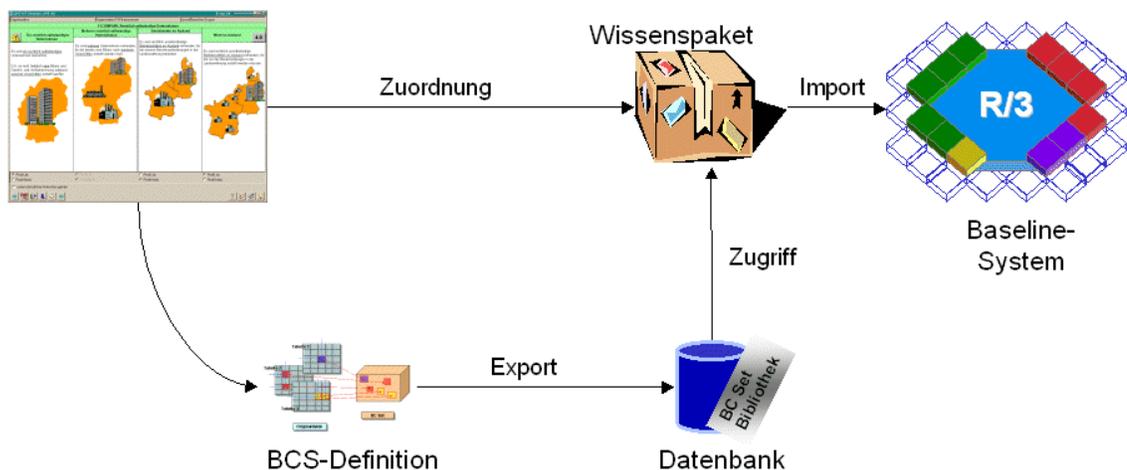


Abbildung 6-5: Automatische Konfiguration mit THESEUS

Die datenbankbasierte Realisierungskomponente THESEUS ermöglicht in Verbindung mit der BC-Set-Technik eine automatische Systemkonfiguration gemäß

den ausgewählten betriebswirtschaftlichen Profilen. Abbildung 6-5 zeigt die Vorgehensweise bei der Konfiguration des Baseline-Systems.

Im Fundamentalmodell werden für alle technischen Profile des Anforderungsnavi-gators eigenständige BC-Sets mit den zur Umsetzung notwendigen Tabelleneinträ-gen definiert. Zudem besteht die Möglichkeit, eigene Sets ohne Bezug zu einem Profil zu definieren und in der Datenbank abzulegen. Somit wird eine BC-Set-Bibliothek aufgebaut, die als Grundlage für die spätere Systemkonfiguration dient. THESEUS erlaubt zwei Arten des Zugriffs auf diese BC-Sets.

- Direkter Zugriff:
In der Zuordnungsdatenbank von THESEUS werden einzelne BC-Sets di-
rekt einem Wissenspaket zugeordnet und können bei Relevanz des Pakets
genutzt werden.
- Indirekter Zugriff:
In der Zuordnungsdatenbank sind ein oder mehrere betriebswirtschaftliche
Profile zu einem Wissenspaket zugeordnet. Bei der Auswahl dieser Profile
im Rahmen der Anforderungsanalyse können durch die Verbindung von
BC-Set und Profil diejenigen Sets evaluiert werden, die für die Umsetzung
relevant sind.

In der Realisierungsphase werden die aus der Bibliothek benötigten Sets identifi-
ziert und über eine Importschnittstelle an das Baseline-System übergeben. Dort
stehen dann vorgefertigte Tabellensätze zur Verfügung, welche die Anforderungen
des Unternehmens beinhalten. Den letzten Schritt bildet die Aktivierung der Sets.

Durch das Aktivieren werden die im BC-Set gespeicherten Werte für die Imple-
mentierung genutzt. Hierbei werden die veränderten oder neu geschriebenen Da-
tensätze in einen Änderungsauftrag geschrieben, der die Daten gegebenenfalls in
das entsprechende Produktivsystem transportiert. Vor der Aktivierung müssen ei-
nige Prüfschritte durchgeführt werden, dazu stehen u. a. folgende Funktionen zur
Verfügung [SAP00, BC-CUS]:

- Vergleich mit Customizing-Tabellen (mit und ohne Schlüsselabfrage),
- Prüfen auf Schlüsselkonflikte (Konsistenzprüfung) sowie
- Kombinierbarkeit.

Nach der Aktivierung stehen die Einstellungen im System zur Verfügung und können genutzt bzw. getestet werden.

6.3.6 Continuous System Engineering

Die kontinuierliche Anpassung von betrieblicher Informationsverarbeitung und Organisation verlangt eine zielgerichtete Ablage und Nutzung des gesammelten Wissens aus dem Bereich der Softwareadaption. THESEUS unterstützt diese Zielsetzung durch eine strenge Kapselung der implementierten Softwareumgebung auf Basis der Wissenspakete.

Die vorangegangenen Ausführungen haben gezeigt, dass es mit der hier entwickelten Workbench möglich ist, ein Produktivsystem mit allen relevanten Informationen zu beschreiben. Dabei werden neben den Ergebnissen der Anforderungsanalyse bzw. der Prozessmodellierung auch die vorgenommenen Systemeinstellungen ausführlich dokumentiert. Das erstellte Baseline-System stellt aus Sicht der Wissenspakete eine in sich abgeschlossene Einheit mit einer bestimmten Ausprägung dar, die als Grundlage für weitere Untersuchungen und Verbesserungsmaßnahmen dient. Aus informationstechnischer Sicht wird nach der Abnahme des Systems ein Versionsstand der Lösung festgehalten bzw. eingefroren. Weiterentwicklungen werden unter einem neuen Versionskennzeichen durchgeführt, wobei auf dem Ursprungszustand eines Wissenspakets aufgesetzt wird (Abbildung 6-6).

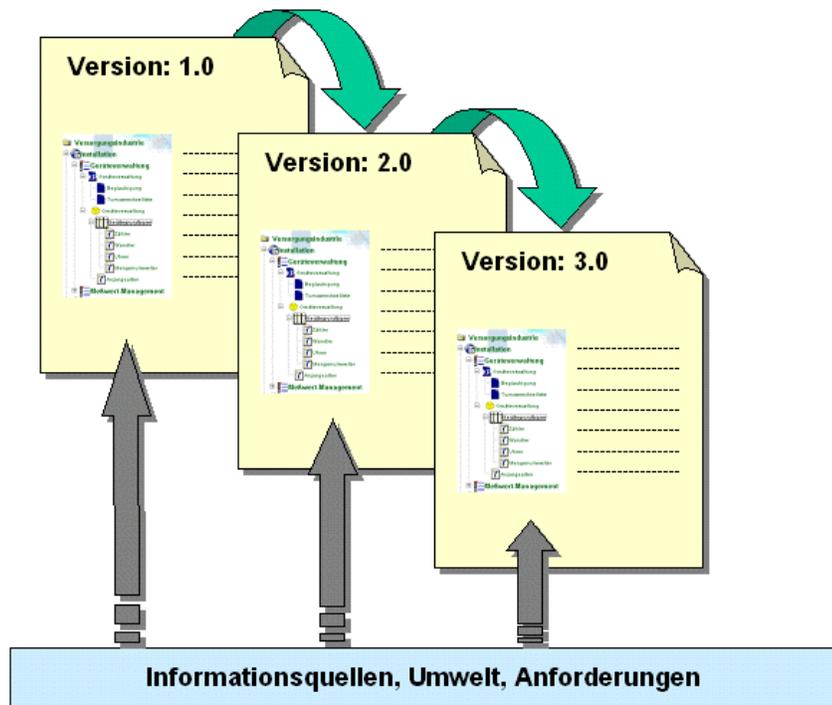


Abbildung 6-6: Versionierung der Wissenspakete

Somit wird sukzessive der Informationsgehalt eines Pakets erhöht und die Wissensbasis erweitert. Dies hat mehrere Vorteile:

1. Bei einem Releasewechsel können die geänderten Customizingtransaktionen zu einem Wissenspaket schnell und einfach identifiziert bzw. bearbeitet werden.
2. Nach der Anpassung des System aufgrund von sich ändernden Umwelteinflüssen kann die neue Systemausprägung mit den vorangegangenen Versionen verglichen werden, da eine gemeinsame und stabile Grundstruktur in Form der Wissenspakete vorliegt. Releaseabhängige Strukturänderungen in den Referenz- bzw. Fundamentalmodellen werden in der Modifikation der Zuordnungsebene berücksichtigt.
3. Neue BC-Sets zu einem Wissenspaket, die im Fundamentalmodell hinzugekommen sind, können identifiziert und gegebenenfalls genutzt werden, um die Prozessabwicklung zu verbessern.
4. Änderungen in den Unternehmensanforderungen können schnell und sicher durch die Auswahl neuer Wissenspakete identifiziert und umgesetzt werden.

6.4 Technische Umsetzung

Nachdem die methodischen Gestaltungsgesichtspunkte in den vorangegangenen Kapiteln ausführlich erläutert wurden, erfolgt nun die Beschreibung der technischen Umsetzung, d. h. die eigentliche Realisierung eines Prototypen. Die Ausführungen teilen sich in die Auswahl eines geeigneten Werkzeugs (6.4.1) und die Erläuterung des Datenmodells für die einzelnen Monitore (6.4.2).

6.4.1 Präsentationswerkzeug

Die Umsetzung des Modells ist abhängig von den technischen Fähigkeiten eines Werkzeugs. Diese Möglichkeiten determinieren und restringieren den Nutzen der hier vorgestellten Methode. Daher ist es von entscheidender Bedeutung, dass die Voraussetzungen, die von einer derartigen Adaption-Workbench erfüllt werden müssen, aus der methodischen Umsetzung und den Zielen von THESEUS abgeleitet werden. STRELLER hat bei der Umsetzung des PANDORA-Konzepts einen Anforderungskatalog definiert, der die Grundfunktionalitäten eines derartigen Werkzeugs aufzeigt. Das hier entwickelte Werkzeug muss diese Aufgaben in gleichem Maß erfüllen. Im einzelnen werden folgende Forderungen gestellt [STRE99, S. 167-168]:

- **Mehrbenutzerfähigkeit:**
Die Pflege des Inhalts und die Wartung des Programms müssen durch mehrere Benutzer möglich sein. Auch im Hinblick auf eine dezentrale Systemdokumentation innerhalb des Projektteams ist diese Voraussetzung essentiell.
- **Releasefähigkeit:**
Das Programm muss die gestellten Aufgaben über unterschiedliche Versionsstände hinweg erfüllen können. In der Regel wird dies durch die angesteuerte Datenbank gewährleistet.
- **Lizenzfähigkeit:**
Der Funktionsumfang des Werkzeugs sollte an eine Lizenzierung gebunden sein, um unterschiedliche Anwendungsmöglichkeiten und/oder Preisvorstellungen seitens des Kunden abzudecken.

- **Sprachenverwaltung:**
Das Programm muss mehrsprachig gestaltet sein, um auch internationale Projekte betreuen zu können. Ein Sprachwechsel muss ohne Informationsverluste jederzeit möglich sein.
- **Auswertungsfunktion:**
Das Werkzeug muss über eine adaptierbare Auswertungskomponente verfügen, die es dem Anwender ermöglicht, beliebige Berichte und Arbeitspapiere (z. B. Aufgabenlisten, Abnahmedokumente) zu erstellen und diese an seine Anforderungen anzupassen.
- **Programmierung:**
Eigene Masken und Formulare sollten erstellt werden können.
- **Projektverwaltung:**
Das Werkzeug muss in der Lage sein, mehrere Projekte zu verwalten und diese gegebenenfalls miteinander vergleichen zu können.
- **Dokumentation:**
Über die Funktionalitäten und Eigenschaften des Programms sollte eine Online-Dokumentation vorliegen, die eine kontextsensitive Hilfestellung für den Anwender bietet.
- **Handhabung:**
Die Grundlagen des Werkzeugs müssen intuitiv erlernbar sein. Diese Anforderung kann durch die Einhaltung vorhandener Standards sichergestellt werden.
- **Internetfähigkeit:**
Die Pflege der Inhalte sowie die Navigation durch die Workbench sollte inter- bzw. intranetfähig sein.
- **Datenbankanbindung:**
Das Werkzeug muss in der Lage sein, aus verschiedenen Datenbanken Informationen zu beziehen und diese für eine automatische Generierung der Monitorinhalte zu nutzen.

Das Werkzeug für die Umsetzung der THESEUS-Methode wurde mit Hilfe des Microsoft Visual Studio entwickelt und basiert auf der Programmiersprache Visual Basic [MICR00].

6.4.2 Datenmodell

Die nachfolgende Beschreibung des Datenmodells ist auf die technische Implementierung des Fundamentalmodells auf der einen und der strukturierten Ablage der projektspezifischen Informationen auf der anderen Seite ausgerichtet. Gegenstand der Betrachtungen sind die notwendigen Tabellen mit deren Bedeutung, Aufgaben und Inhalt.

Das gesamte Datenmodell gliedert sich in drei Bereiche, die nachfolgend beschrieben werden:

- Tabellen zur Ablage der Wissenspakete und aller zugehörigen Informationen (z. B. Attribute, Zuordnungen),
- Tabellen für die Projektverwaltung sowie
- Tabellen für die Benutzer- und Berechtigungsverwaltung.

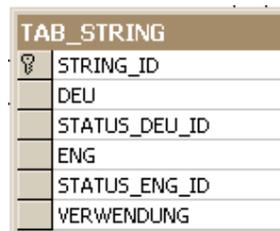
Die technische Umsetzung erfolgt auf Basis des SQL Server der Firma Microsoft in der Version 7. SQL Server ist ein relationales Datenbankmanagementsystem (RDBMS), das ähnlich der mehr desktoporientierten Datenbanksoftware Microsoft Access gleichzeitig mehreren Benutzern den Zugriff auf einen zentral verwalteten Datenbestand ermöglicht [ROBI99, S. 1-2]. Im Gegensatz zu MS Access ist Microsoft SQL Server besser für die Unterstützung von Unternehmenslösungen und großen Datenbanken geeignet, da es u. a. verteilte Transaktionen, aber auch Trigger und gespeicherte Prozeduren ermöglicht. Weitere Vorteile des SQL Server sind u. a. [MICR99, S. V-VI]:

- Anbindung von Anwendungen, die mit Entwicklungssystemen, wie z. B. Microsoft Visual C++ oder Microsoft Visual Basic, erstellt wurden und die Datenbank-APIs (Application Programming Interfaces) einsetzen,
- Unterstützung von Webseiten, die Informationen aus Datenbanken extrahieren und
- Replikation von Daten aus SQL Server in verschiedene Datenbanken, d. h. die Unterstützung von verteilten Datenbanksystemen.

Insbesondere die komfortable Unterstützung von webbasierten Anwendungen mit den vielfältigen Zugriffs- und Manipulationsmöglichkeiten unterstreicht die Eignung von Microsoft SQL Server für die technische Umsetzung von THESEUS.

Für die Identifikation der einzelnen Objekte und Beziehungen, die in der Datenbank gespeichert werden, wird auf die Verwendung von global eindeutigen Bezeichnern (Globally Unique Identifier, GUID) Wert gelegt, um die Datenbankmigration zu erleichtern. Klassische Datenbanken vergeben für die Primärschlüssel von Relationen einen Autowert, der für die installierte Datenbank eindeutig ist. Bei einem Einsatz von mehreren Datenbanken kann es bei der Migration zu Überschneidungen durch mehrfach verwendete Primärschlüssel kommen, die über Konvertierungs- bzw. Migrationsschritte eliminiert werden müssen. GUIDs sind 128-Bit breite Zahlen, deren Generierung auf einer Konvention der "Open Software Foundation (OSF)" beruht. Die dafür verwendeten Algorithmen garantieren, dass diese Zahlen weltweit eindeutig sind. Durch den Verzicht auf die Vergabe eines Autowerts und die Generierung der GUIDs zur Laufzeit durch den SQL Server wird eine überschneidungsfreie Datenmigration ermöglicht, die auf Konvertierungstransaktionen verzichten kann, ohne die Konsistenz zu gefährden.

Ein weiteres Merkmal des Datenbankentwurfs ist der Einsatz zentraler Tabellen für die Speicherung der sprachabhängigen Informationen zu den jeweiligen Objekten. Alle Texte, die innerhalb des Werkzeugs dargestellt werden, sind in der Tabelle TAB_STRING (vgl. Abbildung 6-7) gespeichert, um beispielsweise eine mehrsprachige Objektbeschreibung zu gewährleisten.



TAB_STRING	
PK	STRING_ID
	DEU
	STATUS_DEU_ID
	ENG
	STATUS_ENG_ID
	VERWENDUNG

Abbildung 6-7: Tabelle zur Steuerung der sprachabhängigen Textvergabe

Jedes Objekt wird über die STRING_ID mit dem jeweiligen Beschreibungstext verbunden und durch die sprachabhängigen Spalten (z. B. deu, eng) benannt. Mit dieser normalisierten Vergabe des Objektnamens kann neben der Mehrsprachigkeit auch eine unternehmensindividuelle Namensvergabe realisiert werden.

ABLAGE DER WISSENSPAKETE

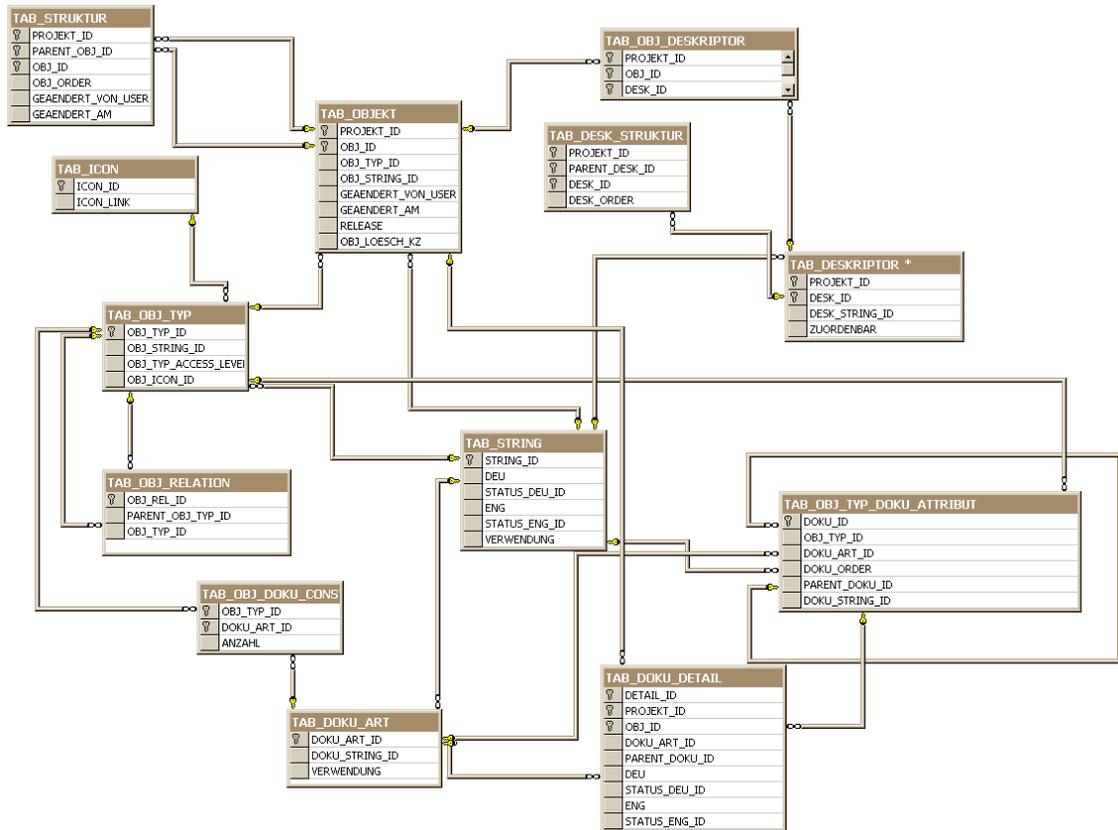


Abbildung 6-8: Tabellenstruktur zur Ablage der Wissenspakete

Abbildung 6-8 zeigt einen Ausschnitt aus dem Datenmodell. Die dargestellten Tabellen sind zur korrekten Ablage der Wissenspakete mit ihren Beziehungen und Integrationsaspekten nötig.

In Tabelle 6-1 sind die Tabellen aufgelistet und beschrieben.

Tabelle 6-1: Tabellen zur Definition und Ablage der Wissenspakete

Tabellenname	Beschreibung und Inhalt
TAB_OBJEKT	<p>Die Tabelle TAB_OBJEKT beinhaltet alle verwendeten Objekte, die im Rahmen von THESEUS eingesetzt bzw. verwaltet werden. Beispiele für derartige Objekte sind</p> <ul style="list-style-type: none"> • Wissenspakete, • IMG-Bausteine, • LIVE KIT Structure Fragen, • LIVE KIT Power Belege oder • Kalkulationsvariablen zur Aufwandsberechnung. <p>Über zusätzliche Tabellen, die nachfolgend beschrieben sind, werden die Ausprägungen, Verknüpfungen und Eigenschaften der Elemente definiert. Die Klassifizierung erfolgt über die Definition von Objekttypen (vgl. Tabelle TAB_OBJ_TYP).</p> <p>Ein Objekt wird mit Bezug zu einem konkreten Projekt in der Tabelle abgelegt. Das Fundamentalmodell wird in diesem Zusammenhang ebenfalls als eigenständiges Projekt angesehen und entsprechend in der Datenbank abgebildet.</p> <p>Unterschiedliche Versionsstände der einzelnen Objekte werden über ein spezielles Kennzeichen unterschieden. Dieses Kennzeichen ermöglicht in Verbindung mit der Projekt-ID die Speicherung unterschiedlicher Paketumgebungen, wie sie für die Nutzung im Sinne des CSE benötigt werden.</p>
TAB_STRUKTUR	<p>In dieser zentralen Strukturierungstabelle werden alle Zuordnungen zwischen den in TAB_OBJEKT angelegten Objekten gespeichert. Ein Beispiel soll diese Logik verdeutlichen:</p> <p>In der Tabelle TAB_OBJEKT wird das Wissenspaket für die Geräteableitung definiert. Zur Implementierung des Pakets sind drei IMG-Bausteine nötig, die ebenfalls als eigenständige Elemente in der Objektabelle abgelegt sind. Die Verbindung zwischen den Bausteinen auf der einen und dem angesprochenen Wissenspaket auf der anderen Seite wird über drei Einträge in der hier beschriebenen Strukturtable realisiert. Falls die betroffenen Bausteine auch für die Implementierung eines anderen Pakets relevant sind, werden weitere Einträge in TAB_STRUKTUR vorgenommen. Somit kann eine redundante Speicherung der Bausteine vermieden und dennoch eine flexible Nutzung derselben erreicht werden.</p> <p>Die Tabelle TAB_STRUKTUR bildet den Kern der Zuordnungsebene von THESEUS.</p>
TAB_DESKRIPTOR	<p>Die Deskriptoren dienen der näheren Beschreibung einzelner Objekte (z. B. der Wissenspakete). In der Tabelle TAB_DESKRIPTOR werden alle verwendeten Beschreibungselemente abgelegt. Im Gegensatz zu den Attributen besitzen die hier abgelegten Deskriptoren keine steuernden Eigenschaften, sie unterstützen vorwiegend das Auffinden von Objekten.</p>

Tabellenname	Beschreibung und Inhalt
TAB_OBJ_DESPKIPTOR	In dieser Tabelle wird die Zuordnung eines Objekts und den jeweiligen Deskriptoren (1:n) vorgenommen.
TAB_DESK_STRUKTUR	Die Tabelle ermöglicht die Strukturierung bzw. Hierarchisierung der verwendeten Deskriptoren. So kann beispielsweise eine einheitliche Beschreibungssequenz festgelegt werden, welche über die Zuordnung des obersten Elements die Hierarchie mit einem Objekt verknüpft.
TAB_OBJ_TYP	<p>Gleichartige Objekte sind einem Objekttyp zugeordnet. Über den Typ werden die spezifischen Ausprägungen der Objekte festgelegt. Die unterschiedlichen Arten von Wissenspaketen beispielsweise werden mit Hilfe von Objekttypen differenziert, ebenso werden die verschiedenen Zuordnungselemente aus den Adaptionswerkzeugen durch Objekttypen getrennt.</p> <p>Anhang A enthält eine Auflistung der Objekttypen, die für den entwickelten Prototyp in der Datenbank angelegt wurden.</p>
TAB_ICON	<p>Zu jedem Objekttyp wird ein Icon für die einheitliche visuelle Darstellung der Objekte gleichen Typs definiert. Dabei wird auf vorhandene Elemente aus den zugeordneten Werkzeugen (z. B. Prozessbeleg-Icons aus dem LIVE KIT Power) zurückgegriffen, um die Wiedererkennung auf Seiten des Anwenders zu fördern. Die Grafiken werden zur Laufzeit des Werkzeugs dynamisch aus der Datenbank gelesen und angezeigt.</p> <p>Die Zuordnung des angezeigten Icons erfolgt indirekt über die Verknüpfung von Objekt und Objekttyp.</p>
TAB_OBJ_RELATION	Diese Tabelle regelt die Zulässigkeit der Objekttypbeziehungen, d. h. es wird über eine Positivzuordnung festgelegt, welche Objekttypen zueinander in Beziehung stehen dürfen. Über diese Kombinationen kann somit zum Beispiel festgelegt werden, aus welchen Hierarchiestufen eine Projektstruktur bestehen darf und an welcher Stelle ein Wissenspaket in dieser Struktur platziert werden kann.
TAB_OBJ_TYP_DOKU_ATTRIBUT	<p>Die einzelnen Attribute eines Objekttyps werden in der Tabelle TAB_OBJ_TYP_DOKU_ATTRIBUT definiert. Unterschiedliche Objekttypen können demnach völlig verschiedene Eigenschaften bzw. Zusatzinformationen besitzen. Als Beispiel kann an dieser Stelle die Möglichkeit zur Anlage der Vorgänger-/Nachfolger-Elemente zu dem Objekttyp Wissenspaket genannt werden.</p> <p>Diese Tabelle steuert somit die Ausprägungen eines Objekttyps.</p>
TAB_DOKU_DETAIL	Auf Basis der Objekttypdefinition werden in dieser Tabelle die spezifischen Ausprägungen der einzelnen Objekte (z. B. Wissenspakete) abgelegt. Die Zuweisung des konkreten Vorgängerpakets ist ein Beispiel für eine derartige Detailinformation.

Tabellenname	Beschreibung und Inhalt
TAB_DOKU_ART	Die Tabelle TAB_DOKU_ART dient vorwiegend der programmtechnischen Steuerung der Bildschirmdarstellung bzw. -inhalte. Über die zugewiesene Art der Attribute können spezielle Subroutinen aus der Werkzeugprogrammierung angesprochen und auf diese Weise die attributabhängige Dialogsteuerung ausgeprägt werden.

PROJEKTVERWALTUNG

Die Projektverwaltung wird über die in Abbildung 6-9 dargestellten Tabellen TAB_PROJEKT und TAB_PROJEKT_USER umgesetzt.

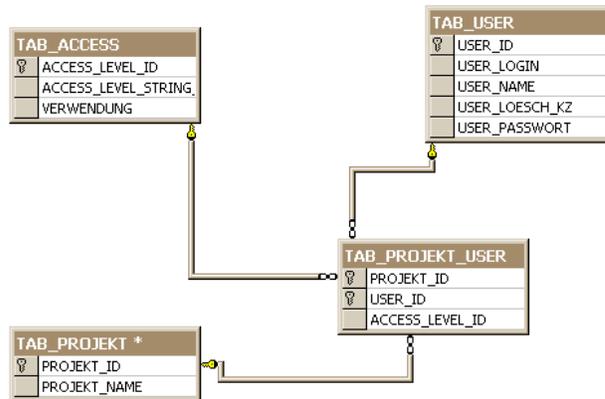


Abbildung 6-9: Tabellenstruktur für die Projekt-, Benutzer- und Berechtigungsverwaltung

Grundsätzlich werden alle Informationen in der Datenbank mit Bezug zu einem Projekt gespeichert, d. h. die Projekt-ID ist als Schlüsselement in jeder Tabelle enthalten. Das Fundamentalmodell wird über die Projekt-ID „0“ verwaltet und in der Datenbank abgelegt. Tabelle 6-2 enthält eine genaue Beschreibung der Tabellen.

Tabelle 6-2: Tabellen für die Projektverwaltung

Tabellenname	Beschreibung und Inhalt
TAB_PROJEKT	Die Projekt-ID identifiziert eindeutig ein Projekt, das im Rahmen der Softwareimplementierung durchgeführt und verwaltet wird. Neben den projektspezifischen Informationen (z. B. Projektname, Projektstart) werden auch Lizenzangaben zur Nutzungsberechtigung hinterlegt.
TAB_PROJEKT_USER	Zu einem Projekt werden die in der Tabelle TAB_USER definierten Benutzer zugeordnet und zusätzlich über die Angabe der Zugriffsberechtigung (ACCESS_LEVEL_ID) die Nutzungsmöglichkeiten bestimmt.

BENUTZER- UND BERECHTIGUNGSVERWALTUNG

Für die Verwaltung der Benutzer und die Regelung der Zugriffsmöglichkeiten auf die Inhalte bzw. Funktionalitäten der Adaption-Workbench werden zwei Tabellen benötigt, die ebenfalls in Abbildung 6-9 dargestellt sind.

Die Berechtigungsverwaltung ist über eine einstufige Zugriffsregelung abgebildet, die sich in einer Positivzuordnung von Benutzer und Access-Level manifestiert.

Tabelle 6-3 beschreibt die Tabellen und Zusammenhänge näher.

Tabelle 6-3: Tabellen für die Benutzer- und Berechtigungsverwaltung

Tabellenname	Beschreibung und Inhalt
TAB_USER	Jeder Benutzer wird in dieser Tabelle mit einem eindeutigen LOGIN-Namen angelegt und über eine Passwortzuweisung für die Benutzung autorisiert.
TAB_ACCESS	Die Tabelle TAB_ACCESS beinhaltet alle Ausprägungen der möglichen Zugriffe auf den Datenbestand von THESEUS. Die Berechtigungsverwaltung gliedert sich in zwei Stufen: <ul style="list-style-type: none"> • Dem Benutzer wird über die Tabelle TAB_PROJEKT_USER in Abhängigkeit des betrachteten Projekts ein bestimmter Access-Level zugewiesen. • Die ACCESS_LEVEL_ID ist zudem einem Objekttyp (TAB_OBJ_TYP) zugeordnet und regelt somit die Zugriffsmöglichkeiten eines Users bei der Bearbeitung bestimmter Objekte.

7 Einsatzszenarien

Mit THESEUS ist es möglich, eine transparente Umsetzung der Ergebnisse einer Analysephase durchzuführen und dabei auf unterschiedliche Informationsquellen zurückzugreifen. Im folgenden werden nun typische Einsatzszenarien von THESEUS vorgestellt. Der involvierten Werkzeuge aus den vorgeschalteten Projektphasen werden nicht mehr in der Tiefe des Details betrachtet, da diese in den konzeptionellen Ausarbeitungen - u. a. PENELOPE [VOGE97, S. 215-234], ODYSSEUS [THOM96, S. 125-152], MENTOR [WED97, S. 179-199], SPARTA [SCHI99, S. 110-182] - zu den jeweiligen Themen nachgelesen werden können. Vielmehr sollen der zusätzliche Aspekt, den THESEUS in die Vorgehensweise der Softwareadaption hineinbringt, herausgestellt und die Nutzenpotenziale der Adaption-Workbench verdeutlicht werden.

Dazu wird zunächst diskutiert, wie THESEUS im Projektverlauf genutzt werden kann (7.1), bevor anschließend näher auf die verschiedenen Möglichkeiten des Einsatzes eingegangen wird (7.2). Dabei werden vorrangig die unterschiedlichen Einstiegspunkte betrachtet, die den Ausgangspunkt für die Arbeit mit der Workbench bieten. Schließlich wird die besondere Vorgehensweise des Konzerneinsatzes von THESEUS beschrieben (7.3).

7.1 Nutzung im Projektverlauf

In diesem Abschnitt wird herausgearbeitet, in welchen Phasen der Implementierung einer Softwarebibliothek THESEUS eingesetzt werden kann. Die Einflüsse der CSE-Methode liefern dabei Ansätze, die klassischen Projektzyklen zu verkürzen [THHU96, S. 90-95]. Die Phasen der Projektierung richten sich nach den von STRELLER identifizierten Schritten eines Adaptionprojekts [STRE99, S. 189-198], wie sie bereits bei der Definition des Lebenszyklus von Wissenspaketen dargestellt wurden (Kapitel 4.3.7). Diese sind in Abbildung 7-1 nochmals einzeln aufgeführt.

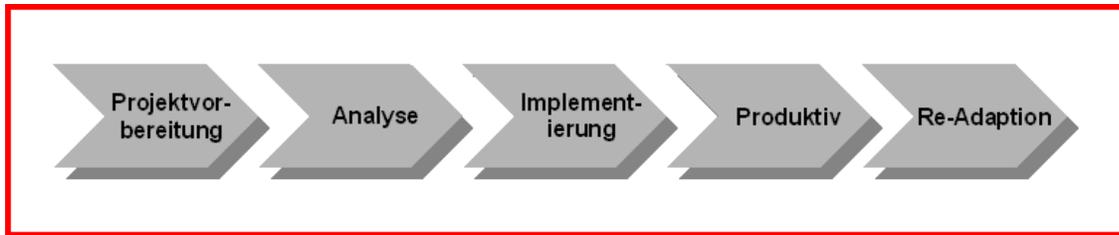


Abbildung 7-1: Projektphasen

Der Einsatz von THESEUS ist analog zu den anderen Konzepten des ITHAKA-Ansatzes (z. B. ODYSSEUS, PENELOPE, MENTOR) in verschiedenen Phasen vorstellbar. Während in der Presales-Phase die Darstellung der Softwarebibliothek und die Analyse des Abdeckungsgrades im Vordergrund stehen, rücken mit zunehmender Implementierungsnähe die anpassende Modellierung und das Baselineing in den Mittelpunkt.

PROJEKTVORBEREITUNG

Bereits in der Projektvorbereitungsphase ist ein Einsatz von THESEUS sinnvoll, da auch ohne Kenntnisse der Softwarelösung ein Überblick über die im Fundamentalmodell enthaltenen Wissenspakete und den damit verbundenen Möglichkeiten des betrachteten Systems geliefert wird.

Als Grundlage für einen Einsatz können sowohl das erwähnte Fundamentalmodell als auch branchentypische Vorlageprojekte aus den zugeordneten Werkzeugen (z. B. LIVE KIT Structure) dienen. Diese zeigen gezielt den empfohlenen Funktionsumfang der Software bzw. typische Lösungen auf und determinieren dabei die Auswahl der Wissenspakete sowie deren Inhalt. Somit wird eine Presselektion der später gewünschten Funktionalität ermöglicht, ohne spezielle Workshops durchgeführt zu haben.

Eine weitere Möglichkeit ist die frühzeitige Einbeziehung bzw. Verknüpfung unterschiedlicher Softwarelösungen und Werkzeuge. THESEUS ist konzeptionell ein offen gestalteter und flexibler Ansatz zur Unterstützung der Softwareadaption, der auch beim Einsatz heterogener Systemlandschaften genutzt werden kann. Ist vor dem eigentlichen Projektstart bereits bekannt, dass neben der neu einzuführenden Softwarebibliothek auch bestehende System weiter betrieben werden, so muss das Fundamentalmodell um diese Aspekte erweitert bzw. angepasst werden, damit alle Integrationsaspekte abgebildet sind. Diese Modifikationen können neue Wissens-

pakete oder geänderte Zuordnungsregeln in den bestehenden Strukturen sein. Über diese Anpassungen entfaltet THESEUS als softwareunabhängige Adaption-Workbench seinen vollen Nutzen.

Die Möglichkeit der Aufwandsermittlung auf Basis von Vorlageprojekten kann in Verbindung mit der Beschreibung bereits feststehender Schnittstellen zu produktiven Altsystemen zur Absicherung der Softwareauswahl bzw. der Kaufentscheidung genutzt werden.

ANALYSE

Im Analyseprozess werden verschiedene Workshops zur Ermittlung der Unternehmensanforderungen durchgeführt. Dabei werden die verschiedenen zielgerichteten Werkzeuge eingesetzt, um wesentliche Lösungsvorschläge für die Implementierungsphase zu erarbeiten. Die mit Hilfe eines LIVE KIT Structure Workshops gesammelten Informationen werden mit den bekannten Werkzeugen ausgewertet und die einzelnen Monitore generiert.

THESEUS kann in dieser Phase dazu genutzt werden, einen transparenten und umfassenden Überblick über die Inhalte der zur Verfügung stehenden Wissenspakete zu erlangen und gegebenenfalls notwendige Modifikationen bzw. Lücken zu erkennen. Die Monitore unterstützen das Projektteam dabei, eine gezielte Vollständigkeitskontrolle bezüglich der durchgeführten Analyseworkshops vorzunehmen und unter Umständen weitere Werkzeuge in die Anforderungsermittlung aufzunehmen. Zudem können schon in dieser Phase erste Hinweise auf die vorzunehmenden Systemeinstellungen gewonnen und in die weiteren Planung aufgenommen werden.

Die vorausgegangenen Workshops erlauben eine Lückenanalyse, die zum einen die Integration der Informationsflüsse zum Inhalt hat, zum anderen auch eine Identifikation funktional nicht abdeckbarer Bereiche zulässt [VOGE97, S. 217-218]. Können diese Lücken nicht beseitigt werden, entsteht ein Konflikt zwischen den Anforderungen des Unternehmens und den Möglichkeiten des Systems. Als Lösungsalternative verweist VOGELANG in diesem Fall auf die kritische Prozessanalyse, die als Vorlage für notwendige Ergänzungsentwicklungen dienen kann [VOGE97, S. 218-219] bzw. auf den Einsatz zusätzlicher Softwarelösungen, welche die gewünschten Funktionalitäten abdecken können. Die Entscheidung für zusätzliche Lösungen bzw. die Aufrechterhaltung bestehender Systeme parallel zur

Softwarebibliothek macht eine Anpassung des Fundamentalmodells durch Modifikationen in der Zuordnungsebene und/oder Ergänzungen bei den Wissenspaketen notwendig, um den vollen Nutzen aus den Möglichkeiten von THESEUS zu ziehen.

Durch den Zugriff auf die einzelnen projektspezifischen Informationsquellen kann auch bereits vor Abschluss der Analysephase für Teilbereiche eine Aufwandsberechnung vorgenommen werden

IMPLEMENTIERUNG

Die Implementierungsphase bildet den Schwerpunkt der Einsatzfelder von THESEUS. Im Rahmen der Anforderungsanalyse, Prozessmodellierung, Berichts-adaption oder Organisationsgestaltung werden wichtige Informationen über das Unternehmen sowie die Ausprägungen der einzusetzenden Softwarebibliothek gewonnen und in den jeweiligen Wissensbasen der einzelnen Werkzeuge gespeichert. Dieses verteilt vorhandene Wissens muss nun für eine effiziente Unterstützung und Koordination der Systemkonfiguration verknüpft und aufbereitet werden.

Zu diesem Zweck werden zunächst die vorhandenen Informationsquellen identifiziert, analysiert und schließlich mit den Paketen verbunden. Auf Basis der eingehenden Daten wird ermittelt, welche Wissenspakete für die Umsetzung der Adaptionsergebnisse relevant sind. Für die Koordination der Customizingaufgaben wird dann die individuelle Projektstruktur durch eine frei definierbare Komposition der Navigationshierarchie unter Verwendung der ausgewählten Wissenspakete erstellt, die dem Projektteam als logische und inhaltliche Vorgabe dient. In diesem Zusammenhang sei nochmals erwähnt, dass es über die Engineering-Komponente möglich ist, zusätzliche Wissenspakete zu definieren, vorhandene Pakete zu erweitern oder neue Zuordnungen vorzunehmen.

Im nächsten Schritt werden die Wissenspakete im System umgesetzt. Dabei wird auf die vorhandenen Ausprägungen des Individualmodells in Form von Tabellen- oder Schnittstellenbeschreibungen zurückgegriffen, die aus dem Fundamentalmodell abgeleitet bzw. in der Wissensbasis der zugeordneten Werkzeuge hinterlegt wurden. Anschließend wird die Systemkonfiguration durchgeführt und mit Hilfe des Customizing-Monitors dokumentiert. Zur Unterstützung kann in dieser Phase der Implementierung mit den BC-Sets gearbeitet werden, die feste Parametersätze und Tabelleneinträge beinhalten und somit die Konfiguration des Baseline-Systems

beschleunigen. In diesem Fall wird die vorhandene Dokumentation, die im Fundamentalmodell zu jedem BC-Set hinterlegt ist, als Grundlage für die Beschreibung des Customizing des bearbeiteten Wissenspakets verwendet.

Nach Abschluss der Systemkonfiguration und der Durchführung intensiver Test-szenarien werden auf Basis der umgesetzten Wissenspakete Abnahmeberichte erstellt und die fertiggestellte Umgebung als eigenständige Version in der Datenbank gespeichert.

PRODUKTIV

Während des Produktivbetriebs kann THESEUS in Verbindung mit dem von WENZEL entwickelten Reverse Engineering Assistant zur Kontrolle des laufenden Systems eingesetzt werden [WENZ00, S. 184-186]. Auf Basis der erstellten Extrakte, die Informationen über vorgenommene Systemeinstellungen, verwendete Stammdaten und durchgeführte Transaktionen enthalten, werden RBE-Sets analysiert und bewertet. Diese Sets sind über die Zuordnungsebene mit den Wissenspaketen verknüpft und können somit Aussagen über die Ausprägungen eines Pakets liefern. Diese Informationen wiederum können mit den ursprünglich ermittelten Anforderungen bezüglich eines Pakets verglichen und bewertet werden. Abweichungen werden identifiziert und zur genaueren Prüfung mit Hilfe des Paketstatus gekennzeichnet. Die weiteren Schritte bilden dann umfangreiche Analysen der Abweichungen bis hin zu Korrekturmaßnahmen aufgrund der festgestellten Ergebnisse.

Die gesamte Navigation erfolgt über THESEUS und die darin enthaltenen Wissenspakete. Somit ist auch in diesem Fall eine hohe Transparenz und Anwenderfreundlichkeit gewährleistet, da im Gegensatz zu den bisherigen Ansätzen auf unnötige Werkzeugwechsel verzichtet wird.

Eine weitere Möglichkeit des Einsatzes in dieser Phase der Softwarenutzung ist die Erweiterung der vorhandenen Dokumentation, die aufgrund von neuen Erkenntnissen aus dem Produktivbetrieb angestoßen wird. Zudem liefert die Analyse der produktiven Software Informationen für weitere Projektphasen im Sinne des Continuous System Engineering.

RE-ADAPTION

Während des produktiven Einsatzes der Softwarebibliothek schließt sich die Phase der dynamischen Adaption (Re-Adaption) an. Die auslösenden Faktoren lassen sich dabei vor allem in zwei Gruppen einteilen [HUF94, S. 242]:

Dynamik des Unternehmens:

- Adaptions- und Organisationsberatung,
- Auswahl von funktionalen Erweiterungen,
- Durchführung von Ergänzungen und
- Abänderung vorgenommener Adaptionen.

Dynamik der Softwarebibliothek:

- Information über Neuerungen oder Änderungen und
- Durchführung von Releasewechseln.

Die Folgen der neuen Anforderungen müssen zunächst überdacht werden, wobei THESEUS als Unterstützungswerkzeug Entscheidungshilfen liefern kann, wie ein Problem zu lösen ist bzw. ob Anpassungen überhaupt und wenn ja, in welcher Art notwendig werden. Grundsätzlich gibt es für die Bewältigung neuer Anforderungen die Möglichkeiten der Erweiterung durch Auswahl neuer Elemente vorhandener Wissenspakete, Ergänzung durch Auswahl neuer Wissenspakete oder Abänderung vorgenommener Einstellungen. Diese Re-Adaption stellt im Produktivbetrieb eine große Herausforderung dar, weil die Konsistenz und Stabilität des laufenden Systems unbedingt sichergestellt werden muss. Umso wichtiger ist die Unterstützung dieser Phase durch geeignete Werkzeuge, die es erlauben, ohne Systemeingriffe mögliche Konsequenzen von Anpassungen zu analysieren und die notwendigen Systemänderungen gezielt unter Beachtung der Integrationsaspekte vorzubereiten.

THESEUS liefert für diese Vorhaben durch die umfassende Speicherung des gesammelten Wissens und den transparenten Zugriff auf die hinterlegten Informationen aus des vorangegangenen Adaptionenprojekten wichtige Hilfestellungen. Ausgehend von den bereits eingesetzten Wissenspaketen können neue Anforderungen in den einzelnen Werkzeugen (z. B. LIVE KIT Structure, LIVE KIT Power) abgebildet und ausformuliert werden. Bezüglich der aktuellen Systemeinstellungen und den sich daraus ergebenden Konsequenzen für die Re-Adaption werden im

Customizing-Monitor neue Konfigurationsaktivitäten angezeigt und darüber hinaus Vorschläge für die Änderungen aus dem Fundamentalmodell abgeleitet. Eine wichtige Rolle spielen in diesem Zusammenhang wiederum die BC-Sets für die automatische Systemausprägung. Neu identifizierte oder modifizierte BC-Sets werden unter Konsistenzprüfungen in das System eingespielt und nach intensiven Tests aktiviert. Somit können alle Integrationsaspekte beachtet und Schnittstellenprobleme vermieden werden. Über die Kalkulationskomponente ist es möglich, den Aufwand für die beschlossenen Änderungen bzw. Erweiterungen zu ermitteln.

Den Einstieg in die Phase der Re-Adaption bildet meist eine ausführliche Analyse des produktiven Systems mit Hilfe des Reverse Business Engineering, das von WENZEL ausführlich beschrieben wurde [WENZ00, S. 61-132]. Dabei wird das laufende System über Extrakte analysiert und in Verbindung mit THESEUS mit den ursprünglichen Anforderungen verglichen. Somit können Defizite oder falsche Systemabläufe anhand einzelner Wissenspakete erkannt und im Rahmen der Re-Adaption behoben bzw. angepasst werden.

7.2 Einstiegsszenarien

Die flexible Zuordnungsebene von THESEUS erlaubt es, auf unterschiedliche Informationsquellen für die Identifikation der relevanten Wissenspakete zurückzugreifen und somit verschiedene Arten von Workshops als Ausgangspunkt für die Realisierungsphase zu nutzen. Im folgenden werden die aus dem ITHAKA-Umfeld bekannten Werkzeuge als Grundlage für den Einstieg in die Arbeit mit THESEUS näher beleuchtet, es sei aber nochmals darauf hingewiesen, dass die hier beschriebene Vorgehensweise auch für andere Datenquellen verwendet werden kann. Voraussetzung hierfür ist die Zuordnung der jeweiligen Elemente zu den Wissenspaketen im Rahmen des Fundamentalmodells und die Definition von Prüffällen, wie sie in Kapitel 4.3.6 beschrieben sind.

ITHAKA beinhaltet eine Vielzahl an Werkzeugen, die als Einstieg für THESEUS verwendet werden können. Namentlich werden hier betrachtet:

- MEDEA [MEHL98] und SPARTA [SCHI99] in Form der Rasterhypothese,
- ODYSSEUS [HUF94] in Form des LIVE KIT Structure ,

- PENELOPE [VOGE97] und MENTOR [WED97] in Form des LIVE KIT Power und
- KIRKE [WENZ00] in Form des Reverse Modelling Assistant.

7.2.1 THESEUS auf Basis der Rasterhypothese

MEDEA wurde entwickelt, um die ODYSSEUS-Methode durch ein geeignetes morphologisches Konzept methodisch zu erweitern und zu verfeinern. Ein merkmalsorientiertes Instrumentarium unterstützt den Anwender bei der Identifikation seiner Anforderungen und stellt ihm zielgerichtet die Informationen zur Verfügung. Diese ermöglichen ihm eine fundierte Auswahl aus dem Lösungspotenzial der Softwarebibliothek. Im Rahmen der Umsetzung wurden betriebstypologische Merkmale und deren Ausprägungen in morphologischen Kästen entwickelt [MEHL98, S. 167-209].

SCHIPP hat im Rahmen des SPARTA- Konzepts eine Methodik entwickelt, wie ein Unternehmen durch die Auswahl der zutreffenden Merkmalsausprägungen aus den betriebstypologischen Rastern über die sogenannte Rasterhypothese das relevante Anwendersegment identifizieren kann [SCHI99, S. 129-132]. Über diese charakteristische Einordnung des Unternehmens werden in einem weiteren Schritt das betreffende vorkonfigurierte System und die entsprechenden Vorlageprojekte in den ITHAKA-Werkzeugen ausgewählt und eingesetzt.

SPARTA liefert die Vorauswahl der Anwendungskomponenten und eine Vorbeantwortung von Fragen und Profilen aus dem LIVE KIT Structure. Zudem beinhaltet es die anwendersegmentspezifische Dokumentation, z. B. in Form der Customizingeinstellungen des vorkonfigurierten Systems. Diese Informationen können als Ausgangspunkt für den Einsatz von THESEUS genutzt werden.

Abbildung 7-2 zeigt die Zusammenhänge zwischen den Konzepten und der Identifikation der Wissenspakete für den Einsatz von THESEUS. Das hier beschriebene Modell geht dabei von einem idealtypischen Fall der vollständigen Abdeckung durch ein vorkonfiguriertes System aus, bei dem auf detaillierte Workshops zur Ermittlung abweichender Anforderungen verzichtet werden kann. In einem solchen Fall dient die Rasterhypothese als einzige Informationsquelle für den THESEUS-Einsatz.

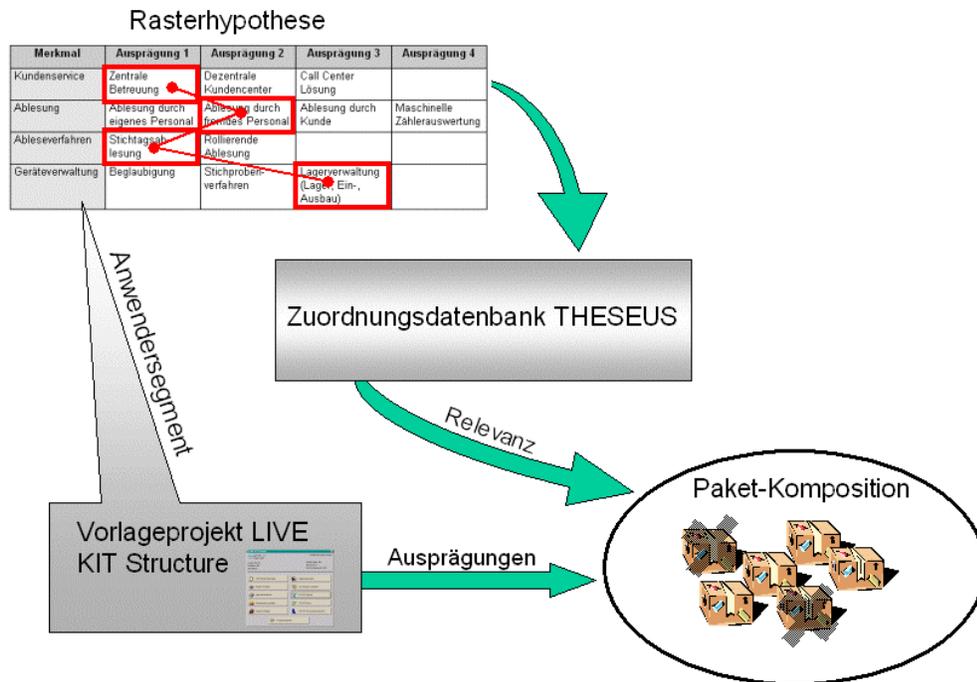


Abbildung 7-2: Rasterhypothese als Informationsquelle für THESEUS

Das Unternehmen erstellt die individuelle Rasterhypothese. Aufgrund der Merkmalsausprägungen wird ein Anwendersegment identifiziert, das als Grundlage für die weiteren Betrachtungen herangezogen wird. In der Zuordnungsdatenbank werden durch Abarbeiten der Prüffälle alle relevanten Wissenspakete für das Anwendersegment ermittelt und im Rahmen der Projektkomposition zur Verfügung gestellt. Über die Integration zu LIVE KIT Structure und dem darin abgelegten Vorlageprojekt können die Detailausprägungen der einzelnen Wissenspakete abgerufen werden. Dies bedeutet, dass die Informationen der Vorauswahl und -beantwortung direkt die Realisierungsaktivitäten bestimmen.

Die Rasterhypothese wird in der Ergebnistabelle der Projektdatenbank gespeichert, die in Abbildung 7-3 dargestellt ist.

ID	ElementID	SegmentID	BenutzerID	Antwort	Zeitstempel
2832	469	2	2	on	20000706029321
2833	490	2	2	Plan	20000706029321
2834	490	2	2	PI	20000706029321
2835	399	2	2	--	20000706029321
2933	407	6	6	100	20000707032817
2934	460	6	6	on	20000707032817
2935	479	6	6	on	20000707032817
2936	480	6	6	on	20000707032817
2937	419	6	6	on	20000707032817
2938	399	6	6	on	20000707032817
2939	399	6	6	on	20000707032817
2940	399	6	6	--	20000707032817
2941	389	6	6	--	20000707032817
2942	467	6	6	on	20000707032817
2943	392	6	6	ist	20000707032817
2944	404	6	6	76	20000707032817
2945	434	6	6	on	20000707032817

Abbildung 7-3: Ergebnistabelle zur Speicherung der Rasterhypothese

Abweichend von dieser idealisierten Darstellung besteht die Möglichkeit, bei der Paketkomposition bzw. bei der Betrachtung der Paketausprägungen auf das modifizierte Kundenmodell aus dem LIVE KIT Structure, das im Rahmen mehrerer Detailworkshops auf Basis des Vorlageprojekts erstellt wurde, zurückzugreifen und somit die Rasterhypothese nur indirekt als Einstiegspunkt für THESEUS zu verwenden. Diese Vorgehensweise ähnelt dann dem klassischen Fall der Paketauswahl auf Basis des LIVE KIT Structure, wie er im folgenden Kapitel dargestellt ist.

7.2.2 THESEUS auf Basis des LIVE KIT Structure

Das LIVE KIT Structure als technische Umsetzungskomponente des ODYSSEUS-Konzepts ermöglicht eine regelbasierte Identifikation der benötigten Potenziale einer Softwarebibliothek. Das Werkzeug wurde für die SAP-Software R/3 konzipiert und sukzessive für neue Softwarelösungen in diesem Umfeld erweitert (z. B. New Dimension-Lösungen der SAP AG). Die drei Analysestufen Auswahl, Reduktion und Zuordnung werden iterativ durchlaufen, wobei das integrierte Regelwerk die Konsistenz der ermittelten Anforderungen in Bezug auf die Umsetzungsfähigkeit der gewählten Softwarebibliothek sicherstellt.

In einem Anforderungs-Dictionary wird das evaluierte Adaptionsergebnis gespeichert und steht als Informationsgrundlage für den Einsatz von THESEUS im

Rahmen der Realisierungsphase zur Verfügung. Abbildung 7-4 verdeutlicht die Nutzung des Dictionaries beim Einstieg in die Arbeit mit der Adaptions-Workbench.

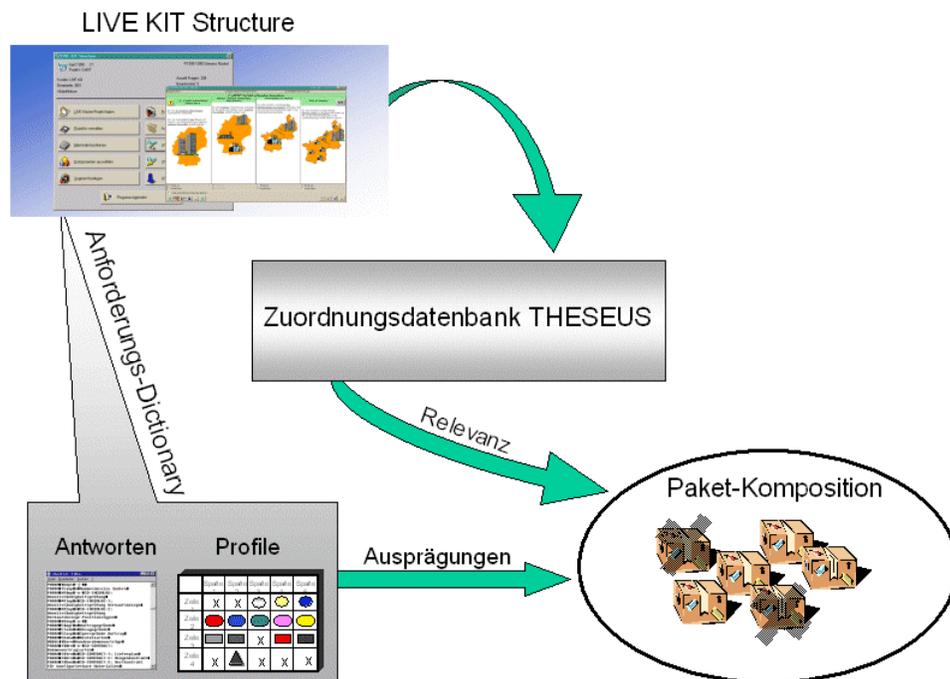


Abbildung 7-4: LIVE KIT Structure als Informationsquelle für THESEUS

Die Anforderungsanalyse beginnt mit der Durchführung eines oder mehrerer LIVE KIT Structure Workshops. Dabei werden der Umfang und die Ausprägungen der einzelnen Prozesse bzw. Funktionen festgelegt und mit Hilfe des Dictionaries gespeichert. Jedem Parameter im Regelwerk (sog. Online Data Dictionary) wird entsprechend der Beantwortung von Reduktionsfragen bzw. der Auswahl betriebswirtschaftlicher Profile ein Wert (j oder n) zugewiesen. In der Zuordnungsdatenbank wird beim Einsatz von THESEUS auf diese Werte zurückgegriffen und die Prüfung für jedes einzelne Wissenspaket durchgeführt. Alle einem Paket zugeordneten Parameter werden evaluiert und der Status wird gemäß den Konventionen der Prüffälle (vgl. Kapitel 4.3.6) entsprechend vergeben. Im nächsten Schritt erfolgt die Komposition der Projektstruktur aufgrund der als relevant gekennzeichneten Wissenspakete. Abweichende Status (z. B. wahrscheinlich relevant, wahrscheinlich nicht relevant, indifferent) stehen dem Anwender für die Projektstruktur ebenfalls zur Verfügung, in diesen Fällen ist eine manuelle Entscheidung über die Verwendung des Pakets notwendig. Es schließt sich die Nutzung der ein-

zelen Monitore bei der Konfiguration des Systems bzw. bei der Berechnung des Realisierungsaufwands an.

Der Einsatz von THESEUS hängt dabei nicht zwingend von einer vollständigen Anforderungsanalyse ab. Durch die Vollständigkeit der Inhalte und Zuordnungen können einzelne Wissenspakete asynchron umgesetzt werden, somit beschleunigt sich bei entsprechender Projektorganisation die Durchführungsdauer für die Implementierungsaktivitäten. Abbildung 7-5 zeigt beispielhaft die Evaluationsdatei (check.txt) aus einem Kundenprojekt.

```

check.txt - Editor
Datei Bearbeiten Suchen ?
PARAM523ge■n■UT-GEW-3: Benutzerdefinierte Gewichtung■
PARAM526ge■-n■■
MODUL03ver■Vertragsabrechnung■
PARAM010ve■-n■UT-CONTRACT: Vertragskategorien■
PARAM011ve■n■UT-CONTRACT-1: Lieferverträge■
PARAM012ve■n■UT-CONTRACT-2: Bezugsverträge■
PARAM013ve■n■UT-CONTRACT-3: Eigenverbrauchsverträge■
PARAM014ve■n■UT-CONTRACT-4: Betriebsverbrauchsverträge■
PARAM015ve■-n■■
PARAM016ve■-n■■
PARAM017ve■-n■■
PARAM018ve■-n■■
PARAM019ve■-n■■
PARAM020ve■-n■■
PARAM021ve■-n■■
PARAM022ve■-n■■
PARAM023ve■-n■■
PARAM024ve■-n■■
PARAM025ve■-n■■
PARAM026ve■-n■■
PARAM027ve■-n■■
PARAM028ve■-n■■
PARAM029ve■-n■■
PARAM030ve■-n■■
PARAM031ve■-n■■
PARAM032ve■-n■■
PARAM033ve■-n■■
PARAM034ve■-n■■
PARAM035ve■-n■■
PARAM036ve■-n■■
PARAM037ve■-n■■
PARAM038ve■-n■■
PARAM039ve■-n■■
PARAM040ve■-n■■
PARAM041ve■-n■■
PARAM042ve■-n■■
PARAM043ve■-n■■
PARAM044ve■-n■■
PARAM045ve■-n■■
PARAM046ve■-n■■
PARAM047ve■-n■■
PARAM048ve■-n■■
PARAM049ve■-n■■
PARAM050ve■-n■■
PARAM051ve■-n■■
PARAM052ve■-n■■
PARAM053ve■-n■■
PARAM054ve■-n■■
PARAM055ve■-n■■
PARAM056ve■-n■■
PARAM057ve■-n■■
PARAM058ve■-n■■
PARAM059ve■-n■■
PARAM060ve■-n■■
PARAM061ve■-n■■
PARAM062ve■-n■■
PARAM063ve■-n■■
PARAM064ve■-n■■
PARAM065ve■-n■■
PARAM066ve■-n■■
PARAM067ve■-n■■
PARAM068ve■-n■■
PARAM069ve■-n■■
PARAM070ve■-n■■
PARAM071ve■-n■■
PARAM072ve■-n■■
PARAM073ve■-n■■
PARAM074ve■-n■■
PARAM075ve■-n■■
PARAM076ve■-n■■
PARAM077ve■-n■■
PARAM078ve■-n■■
PARAM079ve■-n■■
PARAM080ve■-n■■
PARAM081ve■-n■■
PARAM082ve■-n■■
PARAM083ve■-n■■
PARAM084ve■-n■■
PARAM085ve■-n■■
PARAM086ve■-n■■
PARAM087ve■-n■■
PARAM088ve■-n■■
PARAM089ve■-n■■
PARAM090ve■-n■■
PARAM091ve■-n■■
PARAM092ve■-n■■
PARAM093ve■-n■■
PARAM094ve■-n■■
PARAM095ve■-n■■
PARAM096ve■-n■■
PARAM097ve■-n■■
PARAM098ve■-n■■
PARAM099ve■-n■■
PARAM100ve■-n■■
PARAM101ve■-n■■
PARAM102ve■-n■■
PARAM103ve■-n■■
PARAM104ve■-n■■
PARAM105ve■-n■■
PARAM106ve■-n■■
PARAM107ve■-n■■
PARAM108ve■-n■■
PARAM109ve■-n■■
PARAM110ve■-n■■
PARAM111ve■-n■■
PARAM112ve■-n■■
PARAM113ve■-n■■
PARAM114ve■-n■■
PARAM115ve■-n■■
PARAM116ve■-n■■
PARAM117ve■-n■■
PARAM118ve■-n■■
PARAM119ve■-n■■
PARAM120ve■-n■■
PARAM121ve■-n■■
PARAM122ve■-n■■
PARAM123ve■-n■■
PARAM124ve■-n■■
PARAM125ve■-n■■
PARAM126ve■-n■■
PARAM127ve■-n■■
PARAM128ve■-n■■
PARAM129ve■-n■■
PARAM130ve■-n■■
PARAM131ve■-n■■
PARAM132ve■-n■■
PARAM133ve■-n■■
PARAM134ve■-n■■
PARAM135ve■-n■■
PARAM136ve■-n■■
PARAM137ve■-n■■
PARAM138ve■-n■■
PARAM139ve■-n■■
PARAM140ve■-n■■
PARAM141ve■-n■■
PARAM142ve■-n■■
PARAM143ve■-n■■
PARAM144ve■-n■■
PARAM145ve■-n■■
PARAM146ve■-n■■
PARAM147ve■-n■■
PARAM148ve■-n■■
PARAM149ve■-n■■
PARAM150ve■-n■■
PARAM151ve■-n■■
PARAM152ve■-n■■
PARAM153ve■-n■■
PARAM154ve■-n■■
PARAM155ve■-n■■
PARAM156ve■-n■■
PARAM157ve■-n■■
PARAM158ve■-n■■
PARAM159ve■-n■■
PARAM160ve■-n■■
PARAM161ve■-n■■
PARAM162ve■-n■■
PARAM163ve■-n■■
PARAM164ve■-n■■
PARAM165ve■-n■■
PARAM166ve■-n■■
PARAM167ve■-n■■
PARAM168ve■-n■■
PARAM169ve■-n■■
PARAM170ve■-n■■
PARAM171ve■-n■■
PARAM172ve■-n■■
PARAM173ve■-n■■
PARAM174ve■-n■■
PARAM175ve■-n■■
PARAM176ve■-n■■
PARAM177ve■-n■■
PARAM178ve■-n■■
PARAM179ve■-n■■
PARAM180ve■-n■■
PARAM181ve■-n■■
PARAM182ve■-n■■
PARAM183ve■-n■■
PARAM184ve■-n■■
PARAM185ve■-n■■
PARAM186ve■-n■■
PARAM187ve■-n■■
PARAM188ve■-n■■
PARAM189ve■-n■■
PARAM190ve■-n■■
PARAM191ve■-n■■
PARAM192ve■-n■■
PARAM193ve■-n■■
PARAM194ve■-n■■
PARAM195ve■-n■■
PARAM196ve■-n■■
PARAM197ve■-n■■
PARAM198ve■-n■■
PARAM199ve■-n■■
PARAM200ve■-n■■
PARAM201ve■-n■■
PARAM202ve■-n■■
PARAM203ve■-n■■
PARAM204ve■-n■■
PARAM205ve■-n■■
PARAM206ve■-n■■
PARAM207ve■-n■■
PARAM208ve■-n■■
PARAM209ve■-n■■
PARAM210ve■-n■■
PARAM211ve■-n■■
PARAM212ve■-n■■
PARAM213ve■-n■■
PARAM214ve■-n■■
PARAM215ve■-n■■
PARAM216ve■-n■■
PARAM217ve■-n■■
PARAM218ve■-n■■
PARAM219ve■-n■■
PARAM220ve■-n■■
PARAM221ve■-n■■
PARAM222ve■-n■■
PARAM223ve■-n■■
PARAM224ve■-n■■
PARAM225ve■-n■■
PARAM226ve■-n■■
PARAM227ve■-n■■
PARAM228ve■-n■■
PARAM229ve■-n■■
PARAM230ve■-n■■
PARAM231ve■-n■■
PARAM232ve■-n■■
PARAM233ve■-n■■
PARAM234ve■-n■■
PARAM235ve■-n■■
PARAM236ve■-n■■
PARAM237ve■-n■■
PARAM238ve■-n■■
PARAM239ve■-n■■
PARAM240ve■-n■■
PARAM241ve■-n■■
PARAM242ve■-n■■
PARAM243ve■-n■■
PARAM244ve■-n■■
PARAM245ve■-n■■
PARAM246ve■-n■■
PARAM247ve■-n■■
PARAM248ve■-n■■
PARAM249ve■-n■■
PARAM250ve■-n■■
PARAM251ve■-n■■
PARAM252ve■-n■■
PARAM253ve■-n■■
PARAM254ve■-n■■
PARAM255ve■-n■■
PARAM256ve■-n■■
PARAM257ve■-n■■
PARAM258ve■-n■■
PARAM259ve■-n■■
PARAM260ve■-n■■
PARAM261ve■-n■■
PARAM262ve■-n■■
PARAM263ve■-n■■
PARAM264ve■-n■■
PARAM265ve■-n■■
PARAM266ve■-n■■
PARAM267ve■-n■■
PARAM268ve■-n■■
PARAM269ve■-n■■
PARAM270ve■-n■■
PARAM271ve■-n■■
PARAM272ve■-n■■
PARAM273ve■-n■■
PARAM274ve■-n■■
PARAM275ve■-n■■
PARAM276ve■-n■■
PARAM277ve■-n■■
PARAM278ve■-n■■
PARAM279ve■-n■■
PARAM280ve■-n■■
PARAM281ve■-n■■
PARAM282ve■-n■■
PARAM283ve■-n■■
PARAM284ve■-n■■
PARAM285ve■-n■■
PARAM286ve■-n■■
PARAM287ve■-n■■
PARAM288ve■-n■■
PARAM289ve■-n■■
PARAM290ve■-n■■
PARAM291ve■-n■■
PARAM292ve■-n■■
PARAM293ve■-n■■
PARAM294ve■-n■■
PARAM295ve■-n■■
PARAM296ve■-n■■
PARAM297ve■-n■■
PARAM298ve■-n■■
PARAM299ve■-n■■
PARAM300ve■-n■■
PARAM301ve■-n■■
PARAM302ve■-n■■
PARAM303ve■-n■■
PARAM304ve■-n■■
PARAM305ve■-n■■
PARAM306ve■-n■■
PARAM307ve■-n■■
PARAM308ve■-n■■
PARAM309ve■-n■■
PARAM310ve■-n■■
PARAM311ve■-n■■
PARAM312ve■-n■■
PARAM313ve■-n■■
PARAM314ve■-n■■
PARAM315ve■-n■■
PARAM316ve■-n■■
PARAM317ve■-n■■
PARAM318ve■-n■■
PARAM319ve■-n■■
PARAM320ve■-n■■
PARAM321ve■-n■■
PARAM322ve■-n■■
PARAM323ve■-n■■
PARAM324ve■-n■■
PARAM325ve■-n■■
PARAM326ve■-n■■
PARAM327ve■-n■■
PARAM328ve■-n■■
PARAM329ve■-n■■
PARAM330ve■-n■■
PARAM331ve■-n■■
PARAM332ve■-n■■
PARAM333ve■-n■■
PARAM334ve■-n■■
PARAM335ve■-n■■
PARAM336ve■-n■■
PARAM337ve■-n■■
PARAM338ve■-n■■
PARAM339ve■-n■■
PARAM340ve■-n■■
PARAM341ve■-n■■
PARAM342ve■-n■■
PARAM343ve■-n■■
PARAM344ve■-n■■
PARAM345ve■-n■■
PARAM346ve■-n■■
PARAM347ve■-n■■
PARAM348ve■-n■■
PARAM349ve■-n■■
PARAM350ve■-n■■
PARAM351ve■-n■■
PARAM352ve■-n■■
PARAM353ve■-n■■
PARAM354ve■-n■■
PARAM355ve■-n■■
PARAM356ve■-n■■
PARAM357ve■-n■■
PARAM358ve■-n■■
PARAM359ve■-n■■
PARAM360ve■-n■■
PARAM361ve■-n■■
PARAM362ve■-n■■
PARAM363ve■-n■■
PARAM364ve■-n■■
PARAM365ve■-n■■
PARAM366ve■-n■■
PARAM367ve■-n■■
PARAM368ve■-n■■
PARAM369ve■-n■■
PARAM370ve■-n■■
PARAM371ve■-n■■
PARAM372ve■-n■■
PARAM373ve■-n■■
PARAM374ve■-n■■
PARAM375ve■-n■■
PARAM376ve■-n■■
PARAM377ve■-n■■
PARAM378ve■-n■■
PARAM379ve■-n■■
PARAM380ve■-n■■
PARAM381ve■-n■■
PARAM382ve■-n■■
PARAM383ve■-n■■
PARAM384ve■-n■■
PARAM385ve■-n■■
PARAM386ve■-n■■
PARAM387ve■-n■■
PARAM388ve■-n■■
PARAM389ve■-n■■
PARAM390ve■-n■■
PARAM391ve■-n■■
PARAM392ve■-n■■
PARAM393ve■-n■■
PARAM394ve■-n■■
PARAM395ve■-n■■
PARAM396ve■-n■■
PARAM397ve■-n■■
PARAM398ve■-n■■
PARAM399ve■-n■■
PARAM400ve■-n■■
PARAM401ga■-n■UT-GAS: Gasabrechnung■
PARAM402ga■-n■UT-GAS-1: Volumetrische Gasabrechnung■

```

Abbildung 7-5: Evaluationsdatei des LIVE KIT Structure

In dieser Datei sind alle Parameter des Anforderungsnavigators in strukturierter Form abgelegt. Hinter jedem Schlüssel (z. B. 01kla) ist die Bewertung in Form der gegebenen Antwort (z. B. ■n■) gespeichert, die als Grundlage für die Relevanzprüfung herangezogen wird.

7.2.3 THESEUS auf Basis des LIVE KIT Power

Die Prozessmodellierung ist die logische Fortführung der Anforderungsanalyse, die mit Hilfe des LIVE KIT Structure durchgeführt wurde. VOGELANG hat bei der Entwicklung der PENELOPE-Konzeption die Interaktion der beiden Werkzeuge LIVE KIT Structure und LIVE KIT Power ausführlich untersucht und dargestellt [VOGE97, S. 195-213].

Mit dem Anforderungsnavigator wird zunächst durch die optionale Komponentenauswahl der zu betrachtende Umfang der Softwarebibliothek eingeschränkt. Daraufhin wird ein regelbasierter Fragenkatalog bearbeitet, der funktionale Informationen direkt ermittelt. Die Prozessinformationen können implizit, z. B. durch Auswahl des Fachbereichs Instandhaltung, oder explizit durch direkte Fragen, z. B. nach der Fahrzeugverwaltung, ermittelt werden. Durch Regelmechanismen werden Variablen mit einem Wert versehen, der die Konfiguration des Referenzmodells veranlasst. Es schließen sich weitere Modellierungszyklen und Lückenanalysen an. Das Fundamentalmodell von PENELOPE wird so in ein kundenspezifisches Individualmodell überführt, das die benötigten Geschäftsprozesse in der gewünschten Ausprägung enthält.

In Kapitel 7.2.2 wurde gezeigt, wie der Einstieg in die Realisierungsphase auf Basis eines LIVE KIT Structure Projekts möglich ist. Dieser direkte Weg vom Anforderungs-Dictionary zur Auswahl der relevanten Wissenspakete wird durch den in diesem Abschnitt aufgezeigten indirekten Weg ergänzt. Wie bereits erwähnt, besteht die Möglichkeit das Individualmodell des LIVE KIT Power mit den Informationen aus der Anforderungsanalyse zu generieren. Wird nun auf Basis der Ausprägungen dieses Modells eine Paketauswahl in THESEUS vorgenommen, fließen indirekt die Ergebnisse des funktional orientierten Adaptionsschritts in die Auswahl der Wissenspakete mit ein. Diese indirekte Einflussnahme ist durch die Themenorientierung, d. h. die Abbildung der gekapselten Pakete mit allen Aspekten der Adaption, initiiert und beabsichtigt. Eine weitere Möglichkeit ist der Verzicht auf eine Anforderungsanalyse, d. h. der direkte Beginn eines Adaptionsprojekts mit der Prozessmodellierung. In diesem Fall wird auf die automatische Gestaltung des Individualmodells durch importierte Variablen verzichtet und stattdessen eine direkte anpassende Modellierung des Fundamentalmodells vorgenommen. Das so entstandene kundenindividuelle Modell dient dann wiederum als Grundlage der Paketauswahl beim Übergang zur Realisierungsphase.

Abbildung 7-6 veranschaulicht die Zusammenhänge zwischen dem Individualmodell des LIVE KIT Power und dem Einstieg in die Komposition der Projektstruktur in THESEUS. Auf die Darstellung der Interaktion zwischen LIVE KIT Power und LIVE KIT Structure wurde bewusst verzichtet, um die direkte Kopplung von PENELOPE und THESEUS herauszustellen.

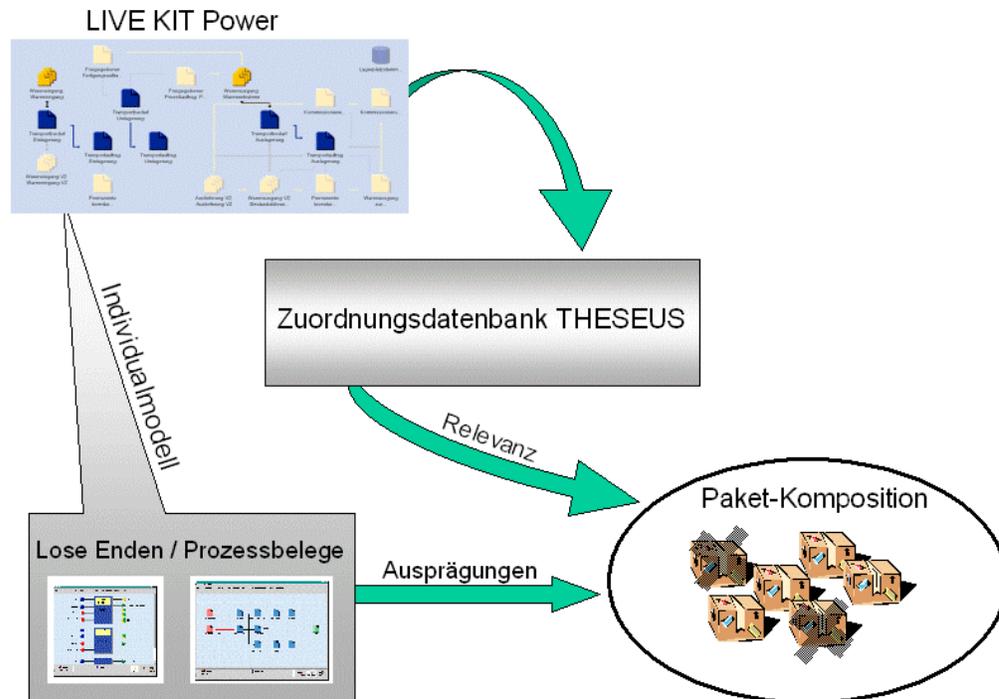


Abbildung 7-6: LIVE KIT Power als Informationsquelle für THESEUS

Die Prozessmodellierung beginnt mit dem Import der Informationen aus einem LIVE KIT Structure Workshop und der automatischen Gestaltung des Individualmodells. Nach der anpassenden Modellierung sind alle Elemente in der PENELOPE-Datenbank mit einem Wert (entweder aktiv oder inaktiv) versehen. Beim Einsatz von THESEUS wird über die Zuordnungsdatenbank auf diese Werte zurückgegriffen und die Prüfung für jedes einzelne Wissenspaket analog zum Vorgehen bei der Nutzung der Informationen aus dem Anforderungsnavigator durchgeführt. Auch in diesem Fall schließt sich die Evaluation der Parameter an, die einem Wissenspaket zugeordnet sind, so dass letztendlich die Statusvergabe erfolgen kann. Das weitere Vorgehen entspricht der Komposition der Projektstruktur, wie sie bereits im vorangegangenen Kapitel beschrieben wurde.

In Abbildung 7-7 ist beispielhaft ein Ausschnitt aus der Projektdatenbank dargestellt.

PB_ID	SEGMENT_ID	LKS_AUSGEBLENDET	ITHAKA_AUSGEBLENDET	NOTIZ	STATUS
16	105	0	1		0
16	106	0	1		0
16	108	0	0		0
16	110	0	1		0
16	111	0	0		0
16	112	0	1		0
16	122	0	1		0
16	123	0	1		0
16	124	1	1		0
16	125	1	1		0
16	126	1	1		0
16	127	1	1		0
16	128	1	1		0
17	104	0	0		0
17	105	1	1		0
17	106	1	1		0

Abbildung 7-7: Projektdatenbank des LIVE KIT Power

Zu jedem Element des Fundamentalmodells, z. B. dem Prozessbeleg mit dem internen Schlüssel 16, ist in Abhängigkeit des betrachteten Projektsegments (z. B. Segment 111) die Statusinformation (aktiv oder inaktiv) gespeichert. Diese Kennzahlen werden bei der Relevanzprüfung analysiert.

7.2.4 THESEUS auf Basis einer RBE-Analyse

Das Reverse Business Engineering ermöglicht eine werkzeuggestützte Istanalyse produktiver SAP-Systeme. Die Analyse ist dabei nicht nur auf technische Einstellungen beschränkt, sondern berücksichtigt insbesondere die betriebswirtschaftliche Anwendung der vom System bereitgestellten Funktionen und Prozesse. Die Situationen, in denen der Reverse Modelling Assistant zum Einsatz kommt, lassen sich in drei Punkten zusammenfassen [WENZ00, S. 237]:

- Überführen einer produktiven Softwarelösung in die ITHAKA-Werkzeuge: Bislang wurde in einem Unternehmen keine Werkzeugunterstützung für die Einführung bzw. Anpassung der eingesetzten Softwarebibliothek bereitgestellt. Im Zuge des Continuous System Engineering sollen zukünftige Modifikationen werkzeugunterstützt durchgeführt werden. Dazu ist es notwendig, die bestehenden Systemdaten mit Hilfe des Reverse Business Engineering zu analysieren und in den verschiedenen Werkzeugen abzubilden.
- Nachdokumentation der Systemeinstellungen:
In einem Unternehmen wurde eine modellorientierte und werkzeugunterstützte Einführung der SAP-Software vorgenommen. Es wurden nach dem

Produktivstart Änderungen an den Systemeinstellungen vorgenommen, ohne diese ausreichend zu dokumentieren bzw. die zugrunde liegenden Modelle, z. B. das Individualmodell aus dem LIVE KIT Power, anzugleichen. In diesem Fall stimmt die Kundenausprägung in den einzelnen Werkzeugen nicht mit dem tatsächlich eingesetzten System überein und erschwert somit die kontinuierlichen Verbesserungen im Sinne des CSE. Mit Hilfe des RBE-Ansatzes können die Systemeinstellungen mit den vorhandenen Informationen aus den zugrunde liegenden Modellen abgeglichen und so die Aktualität der Dokumentation sichergestellt werden.

- Projektverfolgung ohne Modellabgleich:
Bereits in der Implementierungsphase kann der Reverse Modelling Assistant zur Ermittlung des aktuellen Projektstand eingesetzt werden. Dazu wird in kürzeren Abständen eine RBE-Analyse der bestehenden Systemeinstellungen durchgeführt und mit den Unternehmensanforderungen verglichen. Somit kann festgestellt werden inwieweit die zuvor definierten Anforderungen schon in konkrete Systemkonfigurationen umgesetzt wurden.

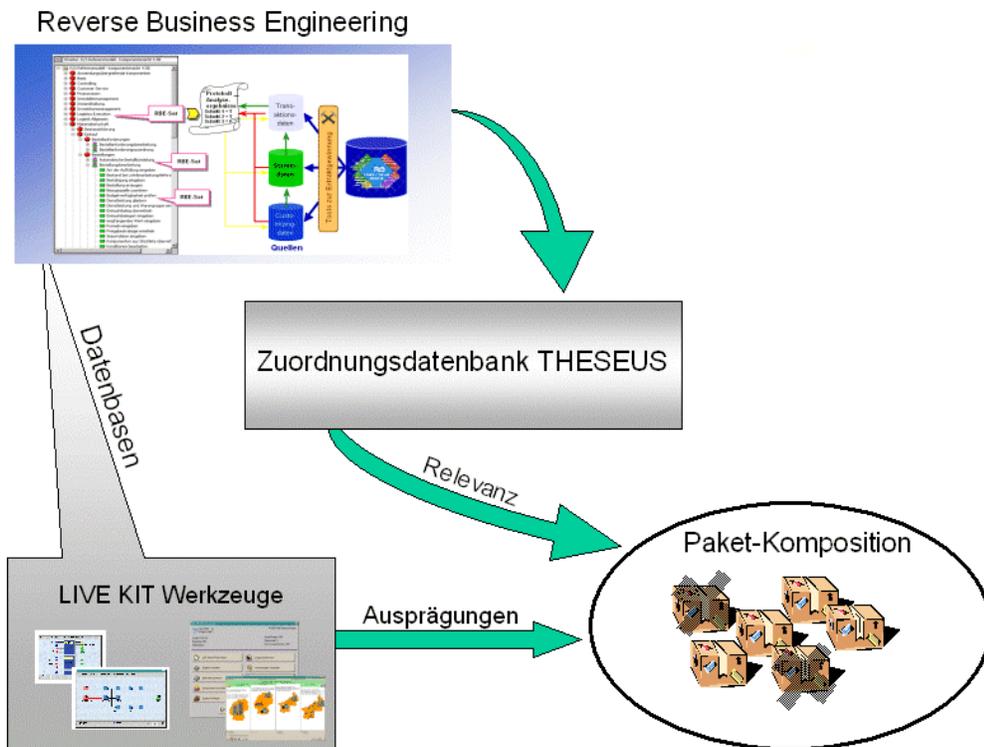


Abbildung 7-8: Reverse Business Engineering als Informationsquelle für THESEUS

Diese Einsatzszenarien zeigen, dass bei vollständiger Integration in das ITHAKA-Konzept eine Projektierung mit Hilfe von THESEUS auf Basis der Ergebnisse einer RBE-Analyse möglich ist. Insbesondere für die beiden ersten Anwendungsfälle bietet sich die Nutzung der RBE-Informationen an.

Abbildung 7-8 veranschaulicht die Zusammenhänge zwischen den beiden Konzepten. An dieser Stelle wird darauf hingewiesen, dass sich die folgenden Ausführungen nur auf die direkte Verbindung der beiden Konzepte bezieht. Ähnlich der Beschreibung des LIVE KIT Power als Informationsquelle, besteht auch in diesem Fall die Möglichkeit, indirekt über die Integration der verschiedenen Werkzeuge mit dem RBE-Ansatz eine Auswahl der relevanten Wissenspakete vorzunehmen.

Das Architekturmodell für das Reverse Business Engineering gliedert sich in folgende Bereiche [WENZ00, S. 165-167]:

1. Datenextraktion:

Die Daten, die sich zur Ableitung von betriebswirtschaftlichen Aussagen eignen, werden aus dem zu untersuchenden System extrahiert. Dabei han-

delt es sich sowohl um Customizing- und Stammdateneinstellungen als auch um Bewegungsdaten (z. B. in Form von Belegen oder Bestelländerungen).

2. Datenanalyse:

Die abstrakte Darstellung der wesentlichen aktiven Elemente einer produktiven Softwarebibliothek ist das Ziel der Analyse. Dadurch wird die Transparenz erhöht und eine einheitliche Kommunikations- und Diskussionsbasis für Betriebswirtschaft und Informationstechnik geschaffen. Zum anderen wird eine Grundlage für die kontinuierlichen Anpassungen geschaffen.

3. Datendarstellung:

Die Erklärungskomponente ermöglicht eine interaktive Kontrolle des Werkzeugs und erlaubt eine Bewertung der erzielten Ergebnisse in einem RBE-Set. Die Modelldarstellung lässt aktive bzw. inaktive Elemente erkennen und hebt Abweichungen zwischen dem bestehenden Kundenmodell und der produktiv untersuchten Softwarelösungen.

Für das Zusammenspiel zwischen dem RBE und THESEUS ist vor allem die Datenextraktion von großer Bedeutung, da durch sie die einzelnen RBE-Sets analysiert und bewertet werden können. Der Aufbau der Sets mit den verschiedenen Prüffällen und -schritten wird bei WENZEL ausführlich beschrieben [WENZ00, S. 133-138].

Die Zuordnungsdatenbank verknüpft einzelne RBE-Sets mit den Wissenspaketen, so dass bei einem Import der Extraktdateien geprüft werden kann, welche Wissenspakete grundsätzlich für die weiteren Projektschritte relevant sind. Die Ausprägungen der Pakete definiert sich durch die Integration des RBE in die anderen Werkzeuge des ITHAKA-Ansatzes. Über die RBE-Analyse wird ein Kundenmodell in LIVE KIT Power und/oder LIVE KIT Structure erstellt bzw. modifiziert, das die näheren Inhalte eines Wissenspakets beschreibt. In Verbindung mit der Analyse der Customizingtransaktionen bzw. -einstellungen ist eine nachträgliche Dokumentation des produktiven Systems über THESEUS möglich. Ebenso verhält es sich mit der Modifikation einer bestehenden Systemdokumentation. In diesem Fall kann durch die Informationen aus der RBE-Analyse ein Abgleich der vorhandenen Systemeinstellungen und der vorliegenden Dokumentation durchgeführt werden, der Aufschluss über notwendige Änderungen gibt.

THESEUS ermöglicht zudem in Verbindung mit den Ergebnissen einer RBE-Analyse die Identifikation fehlender oder fehlerhafter Konfigurationsschritte, die bislang nur mit erheblichem Aufwand vorgenommen werden konnte. Dabei werden auf Basis der Extraktdateien alle relevanten Pakete dargestellt und mit den tatsächlichen Inhalten den geforderten Wissenspaketen gegenübergestellt. Anhand einzelner Customizing-Bausteine und deren Einstellungen kann nun der Abgleich durchgeführt werden.

7.3 Konzerneinsatz

ITHAKA und die daraus entstandenen Konzepte ODYSSEUS [HUGF94], PENELOPE [VOGE97], MENTOR [WED97], MEDEA [MEHL98] und SPARTA [SCH199] beruhen zum größten Teil auf der von Hufgard zugrunde gelegten Annahme der Einführung einer Standardanwendungssoftware in mittelständischen Unternehmen [HUGF94, S. 19]. In diesen Einführungsszenarien gibt es viele Vereinfachungspotenziale, da meist ein effektives Projektteam agiert, welches das notwendige Überblickswissen sowie umfangreiche Kenntnisse der Gesamtzusammenhänge und Interaktionen hat. Die Anforderungen und insbesondere die Geschäftsprozesse sind in einem mittelständischen Betrieb im Normalfall nicht sehr komplex und nicht stark ineinander verwoben, wie das in großen Konzernen der Fall ist [THOM96, S. 140].

WALZ hat sich im Rahmen der OKEANOS-Methode intensiv mit der Einführung einer Softwarebibliothek in Konzernunternehmen beschäftigt [WALZ00]. THESEUS wird im folgenden auf die Tauglichkeit bezüglich des Einsatzes in einem Konzern untersucht und bewertet, insbesondere wird das Zusammenspiel mit OKEANOS dargestellt. Dazu wird zunächst auf dieses Konzept näher eingegangen, bevor aufgezeigt wird, inwieweit THESEUS die Einführung einer Softwarebibliothek in Konzernen als Ergänzung zu OKEANOS unterstützen kann.

7.3.1 Anforderungen im Konzernumfeld

Für den Einsatz im Konzernumfeld wurden die vorhandenen Ansätze (z. B. ODYSSEUS, PENELOPE) angepasst und teilweise erweitert (siehe dazu auch Kapitel 7.3.2). Tabelle 7-1 zeigt die Anforderungen, die an einen Einsatz von THESEUS im Konzernszenario gestellt werden.

Tabelle 7-1: Anforderungen im Konzernumfeld [THOM96, S. 141-142]

Anforderungen im Konzernumfeld	THESEUS
Typische Konzernlösungen und Standards definieren.	Umwandlung des Fundamentalmodells in ein Konzernmodell durch Auswahl und Definition von Paketausprägung. Diese Wissenspakete dienen dann als Vorlage und können entsprechend adaptiert werden.
Typische Konzernlösungen und Standards wiederverwenden.	Konzeptioneller Aufbau und Definition eines Group-Masters. Dokumentation und Sammlung von Wissenspaketen mit typischen Konzernlösungen. Bereitstellen von BC-Sets, die das konzernweit gültige Customizing beinhalten.
Anpassungen und konzernspezifische Ergänzungsentwicklungen zentral vorbereiten und dezentral richtig und schnell umsetzen.	Aufnahme zusätzlicher bzw. Anpassung der vorhandenen Wissenspakete. Erweiterung der Zuordnungsebene um konzernspezifische Elemente bzw. Werkzeuge.
Transparenz und einheitliche Strukturierung von Konzernprojekten.	Vorgabe von verpflichtenden Wissenspaketen und Projektstrukturen. Einschränkung der Kompositionsmöglichkeiten.

Für die Situation im Konzern können vor der Durchführung einzelner Workshops in den Tochterunternehmen drei Schritte vorgeschaltet sein. Strategische betriebswirtschaftliche Analysen helfen dabei, die Unternehmensziele zu definieren oder globale organisatorische Vorgaben (z. B. den Kontenrahmen) zu liefern. Die Gestaltung des Group-Master, d. h. der technischen Ausprägung der Konzernbibliothek, legt die Konzernstandards fest und dient als Vorlage für die weiteren Projekte. Im Verlauf der Folgeprojekte werden zusätzliche oder geänderte Anforderungen in den Group-Master aufgenommen und somit dynamisch die Konzernbibliothek ausgebaut bzw. verbessert. Gerade die Einbeziehung der Folgeprojekte verlangt eine transparente Adaption-Workbench mit einer zentralen Wissensbasis, wie sie THESEUS zur Verfügung stellt. Auf diesen Zusammenhang wird nach der Vorstellung der OKEANOS-Methode näher eingegangen (Kapitel 7.3.3)

7.3.2 OKEANOS-Methode

Mit der OKEANOS-Methode hat WALZ ein Vorgehensmodell entwickelt, wie ein Roll Out von Konzernbibliotheken durchgeführt werden soll. Als Roll Out wird in diesem Zusammenhang die Wiederverwendung eines Basissystems innerhalb eines

Konzerns bezeichnet. OKEANOS beschreibt und unterstützt den Weg von der Entscheidung über die Umsetzung bis zur Weiterentwicklung der Konzernbibliothek [WALZ00, S. 140-179].

Das zentrale Element von OKEANOS ist die Konzernbibliothek, welche die Basis für die Einführung der Softwarelösung(en) darstellt. Die Fundamentalbibliothek ist die Summe aller Funktionen und Parametrisierungsmöglichkeiten einer Softwarelösung. Der Anwender wählt daraus die notwendigen Funktionen aus und gestaltet sie entsprechend seinen individuellen Anforderungen. Auf diese Weise entsteht eine Individualbibliothek als Teilmenge der Fundamentalbibliothek. Für die Erstellung einer Konzernbibliothek wählt die Konzernmutter Bestandteile aus der Fundamentalbibliothek aus, die bei allen Konzerntöchtern identisch sein müssen. Dabei werden Teile der Fundamentalbibliothek bereits parametrisiert, um eine konzernweite Vereinheitlichung von Prozessen zu erreichen. Besonderheiten werden durch die Erweiterungen berücksichtigt, alternative Prozesse werden durch die Konzernmutter zur Verfügung gestellt, aus denen die Töchter die gewünschten Prozesse auswählen können. Auf diese Art und Weise entsteht eine Konzernbibliothek. Die Systeme jeder Konzerntochter bilden Teilmengen ab, sie sind gewissermaßen die Individualbibliotheken im Konzern. Die Konzerntöchter greifen analog zum Standardanwender, der auf die Fundamentalbibliothek zurückgreift, auf die Konzernbibliothek zurück und bauen sich ihr System auf dieser Basis auf [WALZ00, S. 114-117].

Über Anpassungen bzw. Erweiterungen der bestehenden Methoden im ITHAKA-Umfeld konnte mit OKEANOS eine Nutzung der Vorteile des bereits im Mittelstand erprobten Konzepts zur Einführung einer Standardanwendungssoftware realisiert werden, die eine Konzerneinführung vereinfacht und beschleunigt. Für OKEANOS wurden u. a. folgende Modifikation an dem klassischen Instrumentarium vorgenommen [WALZ00, S. 114-156]:

- Ablösen der Fundamentalbibliothek durch eine Konzernbibliothek,
- Erstellen eines Klassifikationsschemas für den Fragekatalog des LIVE KIT Structure (z. B. Empfehlungsregeln),
- Bereitstellen von Manipulationsfunktionen, z. B. für textuelle Anpassungen,
- Einbinden konzernspezifischer Dokumente,

- Bereitstellen einer Erfassungskomponente für den flexiblen Ausbau des Regelwerks,
- Flexibilisierung des Projektumfangs und
- Erstellen konzernspezifischer LIVE KIT Power Objekte.

Der Konzernnavigator wurde als Erweiterung zu den bestehenden Werkzeugen entwickelt und ermöglicht die oben genannte aktive Anpassung der LIVE Tools Neben den technologischen Erweiterungen, die eine schnelle und effiziente Abbildung der betriebswirtschaftlichen Anforderungen eines Konzerns in den bereits vorhandenen Werkzeugen erlauben, wurde ein Monitor entwickelt, der den Ablauf und die hierfür notwendigen Tätigkeiten in einem Roll Out darstellt [WALZ00, S. 140-156].

7.3.3 Die Symbiose zwischen THESEUS und OKEANOS

Die Symbiose ist das nutzbringende Zusammenwirken von Trägern unterschiedlicher Fähigkeiten. Im biologischen Sinne ergänzt sich die unterschiedliche Erbinformation der Symbiosepartner, auf die menschliche Gesellschaft übertragen geht es um unterschiedliche Talente, Kenntnisse und Erfahrungen [PFAF00]. Diese vorwiegend aus der Pflanzen- bzw. Tierwelt bekannte Form der Interaktion kann auch auf den Bereich der Softwareadaption übertragen werden. Die unterschiedlichen Methoden THESEUS und OKEANOS ergänzen sich im Bezug auf die Konzerneinführung einer Softwarebibliothek gegenseitig, was den Nutzen insbesondere für den Anwender erhöht. OKEANOS erlaubt eine standardisierte und somit effektive Vorgehensweise bei der konzernweiten Implementierung einer Softwarelösung, THESEUS stellt demgegenüber ein geeignetes Instrumentarium für die transparente Nutzung von Adaptionwissen zur Verfügung, das die Umsetzungsaktivitäten erheblich beschleunigt. Im folgenden wird das Zusammenwirken der beiden Ansätze beschrieben.

THESEUS verfolgt als übergreifende Adaptions-Workbench das Ziel, über alle Bereiche der Softwareadaption Wissen zu sammeln und aufzubereiten. Gerade in einem heterogenen Konzernumfeld kommt dieser Wissensnutzung eine große Bedeutung zu, um unnötige und hinderliche Redundanzen bzw. Inkonsistenzen zu vermeiden. Mit OKEANOS hat WALZ ein Vorgehensmodell entwickelt, das zur

Erreichung dieser Zielsetzung wertvolle methodische Richtlinien liefert. THESEUS greift diese Ideen auf und stellt Techniken (z. B. Pre-Customizing mit BC-Sets) für die praktische Anwendung zur Verfügung. Die Interdependenzen der beiden Ansätze lassen sich in folgenden Punkten zusammenfassen:

1. Konzernweite, zentrale Datenbank:

Die Sammlung und Verwertung des konzernweit entstandenen Wissens bezüglich der Softwareadaption bedarf einer zentralen Datenbank, die es den verschiedenen Anwendergruppen ermöglicht, zielgerichtet Informationen abzurufen und entsprechend anzuwenden. THESEUS geht ebenfalls von einer zentralen Bereitstellung aller relevanten Daten zu einem speziellen Thema aus und bietet mit der intranetfähigen Adaptions-Workbench das notwendige Instrumentarium, um dieses Wissen transparent zu machen. Die Benutzerverwaltung regelt über die Berechtigungsvergabe den Zugriff auf die abgelegten Informationen. Verteiltes Wissen wird über die zentrale Zuordnungsdatenbank bei Anfragen selektiert und dem Anwender in konzentrierter Form zur Verfügung gestellt. Redundante Ablagen werden bei diesem Vorgehen vermieden.

2. Aufbau einer konzernweiten Wissensbasis:

Die Ausführungen zum Wissensmanagement haben gezeigt, dass die Nutzung des Wissens nur dann möglich ist, wenn eine allgemein zugängliche Wissensbasis aufgebaut wird, die alle zur Verfügung stehenden Informationen beinhaltet. THESEUS regelt in diesem Zusammenhang den Aufbau und die Struktur der Wissensbasis durch die Bereitstellung der Wissenspakete. Dadurch können zu einem Thema konzernweit alle relevanten Daten abgefragt und analysiert bzw. modifiziert werden.

3. Wiederverwendbarkeit von Wissenspaketen

Die Kapselung der Pakete in Verbindung mit der zentralen Eigenschaft der Vollständigkeit ermöglicht die Wiederverwendung einzelner Pakete, um eine einheitliche Systemkonfiguration oder Adaptionsstruktur zu gewährleisten. Der Roll Out im Konzernumfeld, d. h. die Wiederverwendung eines Basisystems, fordert genau diese Eigenschaft. Dies bedeutet, dass THESEUS gerade den Konzern-Roll Out durch die in sich vollständigen Pakete effektiv unterstützt, wenn nicht sogar erst ermöglicht. Die Wiederverwendbarkeit beschränkt sich beim Einsatz von THESEUS nicht nur auf einzelne Wis-

senspakete, vielmehr können ganze Projekte bzw. Projektstrukturen mehrmals eingesetzt und so auf das bereits gespeicherte Wissen des Konzerns zurückgegriffen werden.

4. Nutzung von BC-Sets:

Bereits unter Punkt drei wurde auf die große Bedeutung der Wiederverwendbarkeit von Konzernstandards im Rahmen der OKEANOS-Methode hingewiesen. THESEUS ermöglicht durch die Nutzung der BC-Sets die technische Realisierung dieser Forderung. Der Aufbau des Group Master mit den vereinheitlichten Wissenspaketen beinhaltet auch die Ablage des vordefinierten Customizing zur Abbildung der konzernweiten Prozesse, soweit sie von der Konzernmutter vorgegeben werden. Über BC-Sets werden die Customizingeinstellungen gebündelt und strukturiert an die Tochtergesellschaften weitergereicht. Die integrierten Konsistenzprüfungen bei der Aktivierung der Sets im jeweiligen Anwendungssystem der Tochterunternehmen erlaubt auch eine nachträgliche Uniformierung der Individualbibliotheken. Der Zugriff über die allgemein zur Verfügung stehenden Wissenspakete mit einheitlichen Inhalten beschleunigt den Roll Out und erhöht die Transparenz zusätzlich. In diesem Zusammenhang sei auf die verbesserten Möglichkeiten bei der Verbreitung konzernweiter Anpassungen im Sinne des CSE hingewiesen.

Die vorangegangenen Ausarbeitungen zeigen, dass THESEUS eine Ergänzung der OKEANOS-Methode darstellt und die Softwareeinführung bzw. -betreuung in Konzernunternehmen beschleunigen kann. OKEANOS liefert dabei die notwendigen Anpassungen der Werkzeuge, z. B. in Form der Konzernbibliothek, während THESEUS die Koordination und Vereinheitlichung des Roll Out-Szenarios begleitet und unterstützt. Zudem hilft THESEUS bei der Abwicklung der Folgeprojekte im Sinne des Continuous System Engineering durch die Bereitstellung der Informationen aus der aufgebauten Wissensbasis.

8 Bewertung und Ausblick

Dieses Kapitel zeigt die Möglichkeiten auf, wie THESEUS in künftigen Entwicklungen verwendet werden kann und welche weiteren Einsatzgebiete sich für die Adaptionen-Workbench ergeben.

Wie die Wissenspakete in einem neuen Ansatz zur kontinuierlichen betriebswirtschaftlichen Entwicklung produktiver SAP-Systeme genutzt werden können, zeigt Kapitel 8.1. Die Integration von THESEUS in die von BÄTZ entwickelte Methode zur internetbasierten Abwicklung des Beratungsprozesses [BÄTZ00] beschreibt Kapitel 8.2.

Abschließend wird eine Gesamtbewertung durchgeführt, die zeigt, ob die in Kapitel 1.4 formulierten Ziele durch die entwickelte Methode erreicht werden (8.3).

8.1 Kontinuierliche betriebswirtschaftliche Entwicklung

Bereits im Jahr 1999 wurde unter dem Akronym KRONOS (Kontinuierliche R/3-basierte betriebswirtschaftliche Evolution, Organisationsentwicklung und Einbindung Neuer Releases in Operierende R/3-Systeme) ein Forschungsprojekt ins Leben gerufen, das eine methodische Unterstützung des Continuous Change für produktiv genutzte Softwaresysteme zum Ziel hat [BÄTZ99a].

Ausgangspunkt des Projekts war die Erkenntnis, dass sich die Unterstützung des Continuous Change hauptsächlich auf die technische Seite, z. B. die Verbesserung der Systemperformance oder die Skalierung der Rechnerkapazität, beschränkt. Eine betriebswirtschaftlich ausgerichtete Vorgehensweise, welche die vorhandenen Ansätze berücksichtigt, ist nicht vorhanden [BÄTZ99a]. Die Probleme, die im Rahmen der dynamischen Adaption auftreten, fasst BÄTZ wie folgt zusammen [BÄTZ99a]:

- **Fehlende Durchgängigkeit:**
Die mangelnde Dokumentation der Probleme, Lösungen und Ergebnisse der einzelnen Projekte führt zu Informationsverlust.
- **Keine Frühwarnung oder Unterstützung bei der Diagnose:**
Die fehlende methodische und technische Unterstützung für Frühwarnsysteme, geplante oder ad-hoc Audits führt zu einer Degeneration des Systems.

- Organisatorische Verzögerungen:
Der Handlungsbedarf wird nicht erkannt oder Änderungen werden möglichst lange hinausgezögert, müssen dann aber überstürzt durchgeführt werden.
- Geringe Ausnutzung der Potenziale:
Vorhandene Spielräume, neue Optionen und organisatorische Freiräume der bestehenden Situation sind aufgrund der mangelnden Transparenz nicht erkennbar und können somit nicht gezielt genutzt werden.

Über definierte Szenarien können diese Probleme umgangen werden. Die sogenannten Change Szenarien stellen in dieser Methode die zentralen Elemente dar. Sie sind auf konkrete Problemstellungen ausgerichtet und bestehen aus mehreren lösungsorientierten Profilen.

In Abbildung 8-1 ist der Aufbau eines Change Szenarios dargestellt. Ausgehend von einem Identifikationsteil stehen für die Problemstellung verschiedene Change Profile zur Verfügung. Diese decken verschiedene Alternativen der Problemstellung ab und können unterschiedliche Restriktionen besitzen. Letztere ermöglichen die Bewertung der Lösungen durch Aufzeigen der Konsequenzen einer Umsetzung. Deskriptoren ermöglichen eine schnelle und gezielte Suche aufgrund fachbezogener Schlagworte. Durch den Einsatz von Change Szenarien können Strukturen dargestellt werden, indem Change Profile wiederum eindeutig auf nachgeordnete Change Szenarien verweisen. Dadurch können flexible aber trotzdem stabile Hierarchien entwickelt werden, mit dem Ziel einer (weitgehend) redundanzfreien Beschreibung von Aktivitäten, die bei der Durchführung von Reorganisationsmaßnahmen aus Sicht des Systems durchzuführen sind.

Das Gliederungsprinzip für die Struktur sollte die Ausrichtung des betrachteten Unternehmens widerspiegeln. Diese Ausrichtung wurde bereits bei der Komposition der projektspezifischen Struktur in THESEUS berücksichtigt und kann somit als Ausgangsbasis für die Zusammensetzung der Change Szenarien genutzt werden. Insbesondere der Rückgriff auf die bearbeiteten und dokumentierten Wissenspakete ist eine wichtige Voraussetzung zur Unterstützung betriebswirtschaftlich und systemtechnisch konsistenter Anpassungen im Sinne des CSE. Die Wissenspakete beinhalten alle Auskünfte über die aktuellen Ausprägungen des produk-

tiven Systems. Diese Informationen sind für die Arbeit mit KRONOS von essentieller Bedeutung

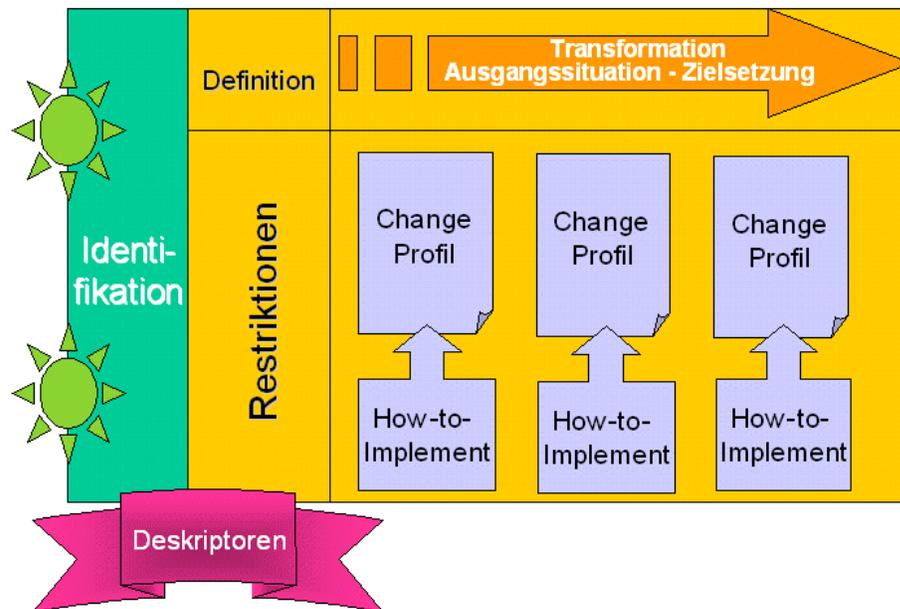


Abbildung 8-1: Aufbau eines Change Szenarios (in Anlehnung an [BÄTZ99a])

In Abbildung 8-2 wird der Zusammenhang zwischen dem Aufbau eines Change Szenarios und der Nutzung der bereits implementierten Pakete verdeutlicht.

Die Abbildung zeigt einen Ausschnitt aus dem entwickelten Prototyp zur Umsetzung von KRONOS. Das im Vorfeld bearbeitete Wissenspaket für die Abbildung der Zählerinstallation dient im vorliegenden Beispiel als Grundlage für die Erweiterung des Zählerwesens innerhalb des Standorts Berlin. Es wird deutlich, dass das Szenario auf die bereits in Kapitel 5.3.2 komponierte Projektstruktur referiert und somit die dort angelegten Informationen direkt nutzen kann.

Für das Wissenspaket stehen die in Kapitel 5 beschriebenen Monitore zur Verfügung. Alle Änderungen, z. B. das Einfügen neuer Systemdokumentationen oder die Änderung der genutzten BC-Sets, werden in der zentralen Datenbank angelegt und bilden eine neue Version des Pakets.

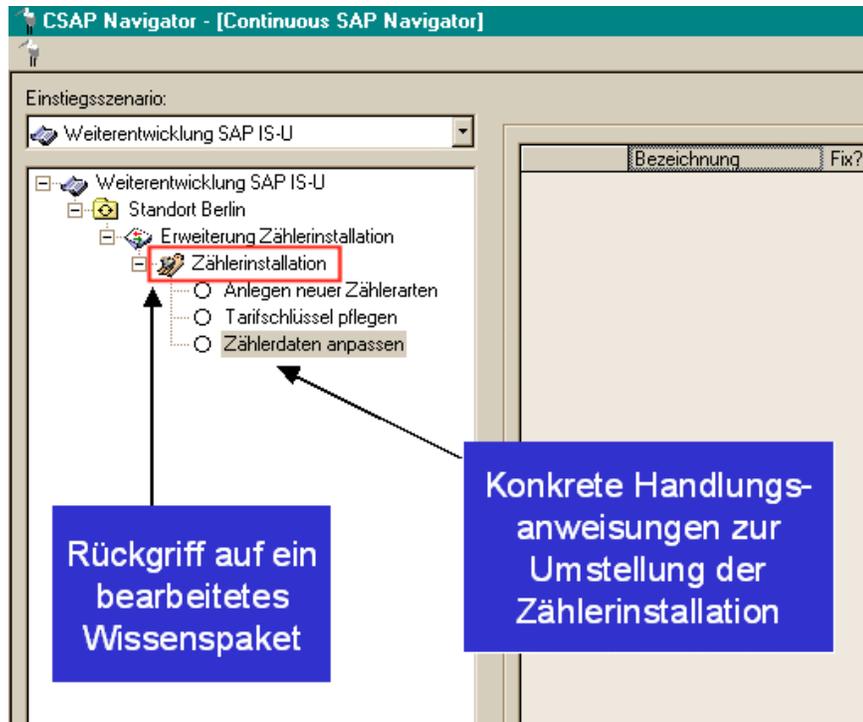


Abbildung 8-2: Nutzung der Wissenspakete bei der Definition eines Change Szenarios

Über die Integration von KRONOS und THESEUS wird somit eine werkzeugunterstützte Vorgehensweise zur konsistenten Weiterentwicklung eines produktiven Systems und zur organisatorischen Koordination der Aktivitäten ermöglicht.

8.2 Internet based Consulting

Berater und Softwarehäuser sind die Zielgruppe für Internet based Consulting (IBC), d. h. für die Unterstützung des Beratungsprozesses durch Methoden und Techniken des Internets. Die grundsätzliche Idee ist die Nutzung des Internets mit Hilfe interaktiver Werkzeuge in der Beratung. IBC basiert auf einer am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg erstellten wissenschaftlichen Arbeit zum Thema „Internetbasierte Abwicklung von Consultingprojekten und -analysen im Umfeld betriebswirtschaftlicher Softwarebibliotheken“ (IANUS) und wurde in Form des Electronic@BusinessCheck realisiert [IBIS00].

Folgende Gründe sprechen für den von BÄTZ entwickelten IBC-Ansatz [BÄTZ00]:

- Die Internettechnologie verspricht höhere Prozesskapazitäten und geringere Abwicklungskosten,
- ein globaler Informationsfluss kann etabliert werden,
- die Datenhaltung kann bei dezentralem Einsatz (mobile, home, inhouse computing) zentralisiert werden,
- die Interaktionskosten zwischen Berater und Kunden können reduziert werden,
- die Markteffizienz kann durch Informationsverteilung erhöht werden,
- geographische Bedingungen und Geschäftsabwicklung können separiert werden,
- die Transaktionskosten können gesenkt werden und
- durch Personalisierung, Kooperationsunterstützung sowie bedarfsgerechte Analysen können Effizienz und Effektivität der Beratung gesteigert werden.

IBC ist geeignet, um die Funktionalitäten eines Produktes auf breiter Basis in Form von Anforderungsanalysen und Checklisten zu vermitteln. Aber auch andere Möglichkeiten, wie die Durchführung strukturierter Interviews oder die Kundenbetreuung, können durch die Potenziale von IBC unterstützt werden.

Grundsätzlich besteht der Electronic@BusinessCheck aus drei Bereichen, die im folgenden kurz vorgestellt werden.

INFORMATION BASE

Die Information Base versorgt den Besucher der Webseite mit Informationen zur Anwendung und zur Vorgehensweise. Ein ausgewiesener Bereich zur Kontaktabwicklung speichert die Kontaktdaten in eine separate Datenbank. Die Dateninhalte werden automatisch aus dem öffentlichen Bereich in die Datenbank der geschützten Analyseinhalte und -ergebnisse übernommen.

ANALYSEBEREICH

Die Gestaltung des Analysebereichs wurde mit Rückgriff auf ein breites Spektrum an Methoden der Implementierung von Standardanwendungssoftware vorgenommen. Mit Hilfe von hierarchischen Strukturen, die über ein komplexes Regelwerk miteinander verknüpft sind, werden die Analyseinhalte dynamisch aufbereitet. Da-

bei werden komplexe Projektorganisationsformen, Rollenkonzepte und die Handhabung von Analysevarianten unterstützt. Das Verständnis für die Inhalte wird durch die Verknüpfung der Elemente mit kontextsensitiven Dokumentationen erhöht. Ziele einer Analyse können

- die Ermittlung von Anforderungen,
- Eignungsbetrachtungen für vorkonfigurierte Lösungen,
- strukturierte Interviews bzw. Checklisten oder
- die Strukturierung projektbezogener synchroner oder asynchroner Kommunikation sein (z. B. E-Mail, Internet-Forum, Chat).

Je nach Umfang können vollständige Analysen oder Vorabanalysen zur Vorbereitung detaillierter Workshops durchgeführt werden.

ERGEBNISSE

Als Ergebnis der Analysen werden unterschiedliche Berichte angeboten. Diese geben verschiedene Perspektiven auf die Beantwortung der Analysefragen oder leisten deduktive Ableitungen auf Basis des Regelsystems. Auf diese Art und Weise können automatisch Schlussfolgerungen für weitere Projektschritte bzw. -inhalte gezogen werden, die direkt aus der Analyse resultieren.

Der Ablauf des Electronic@BusinessCheck gliedert sich in vier Schritte, die sequentiell durchlaufen werden:

1. Die Projektadministration zur Erfassung und Verwaltung der Rahmendaten des Unternehmens:
In diesem Schritt werden zudem die Mitarbeiter gepflegt und die benötigten Berechtigungen vergeben.
2. Die Beantwortung des Online-Fragebogens zur Analyse der Ist-Situation:
Diese Analyse wird durch Dokumentationen und Tutorials unterstützt sowie mit Hilfe eines Regelwerks zielgerichtet gesteuert.
3. Die automatische Generierung der Ergebnisse:
Der Anwender kann wählen zwischen der reinen Ergebnisdarstellung des Online-Fragebogens oder einer strukturierten Auswertung, durch die das Unternehmen über einen Regelmechanismus in die e-Business-Entwicklung eingeordnet wird.

4. Der strukturierte Kommunikationsfluss über das projektspezifische Forum: Dieser Schritt dient vor allem der Vorbereitung von Detailworkshops, die auf den Ergebnissen der Analyse aufbauen. Über diesen Bereich erfolgt gleichermaßen der Datenaustausch in Form des beantworteten Fragebogens.

Die Beschreibungen zeigen, dass sich der Electronic@BusinessCheck vor allem auf die Kontaktaufnahme und die Ermittlung der Unternehmensanforderungen über einen Fragebogen beschränkt. Der volle Nutzen des IBC ist erst dann erreicht, wenn die gesammelten Informationen in strukturierter Form als Grundlage für weitere Analyseworkshops genutzt werden können.

THESEUS kann als integrierte Adaption-Workbench dazu eingesetzt werden, die ermittelten Ergebnisse in einzelne Wissenspakete zu überführen und damit eine Umsetzung der e-Business-Strategie zu unterstützen. Abbildung 8-3 zeigt eine mögliche Integration von THESEUS und IANUS.

Der online beantwortete Fragebogen dient als Einstieg für die erste Stufe der THESEUS-Methode. Über die ausgewählte Beschaffenheit der Geschäftsabwicklung werden relevante Wissenspakete aus dem Fundamentalmodell identifiziert und für die Projektkomposition zur Verfügung gestellt. Nach der Durchführung der einzelnen Workshops zur detaillierten Ermittlung der Anforderungen an das e-Business durch den Einsatz der unterstützenden Werkzeuge (z. B. LIVE KIT Structure) werden weitere Informationsquellen eingelesen und die Ausprägungen der Paketinhalte näher spezifiziert. Die weitere Projektentwicklung bzw. der weitere Einsatz von THESEUS entspricht den bereits skizzierten Einsatzszenarien in Kapitel 7.

Die Ausführungen machen deutlich, dass THESEUS auch in diesem Umfeld zur koordinierten und wissensbasierten Unterstützung der Implementierungsphase genutzt werden kann. Über die flexible Zuordnungsebene werden beliebige Elemente aus dem IBC-Ansatz mit den Wissenspaketen verknüpft und somit das Fundamentalmodell um die gewünschten Objekte erweitert.

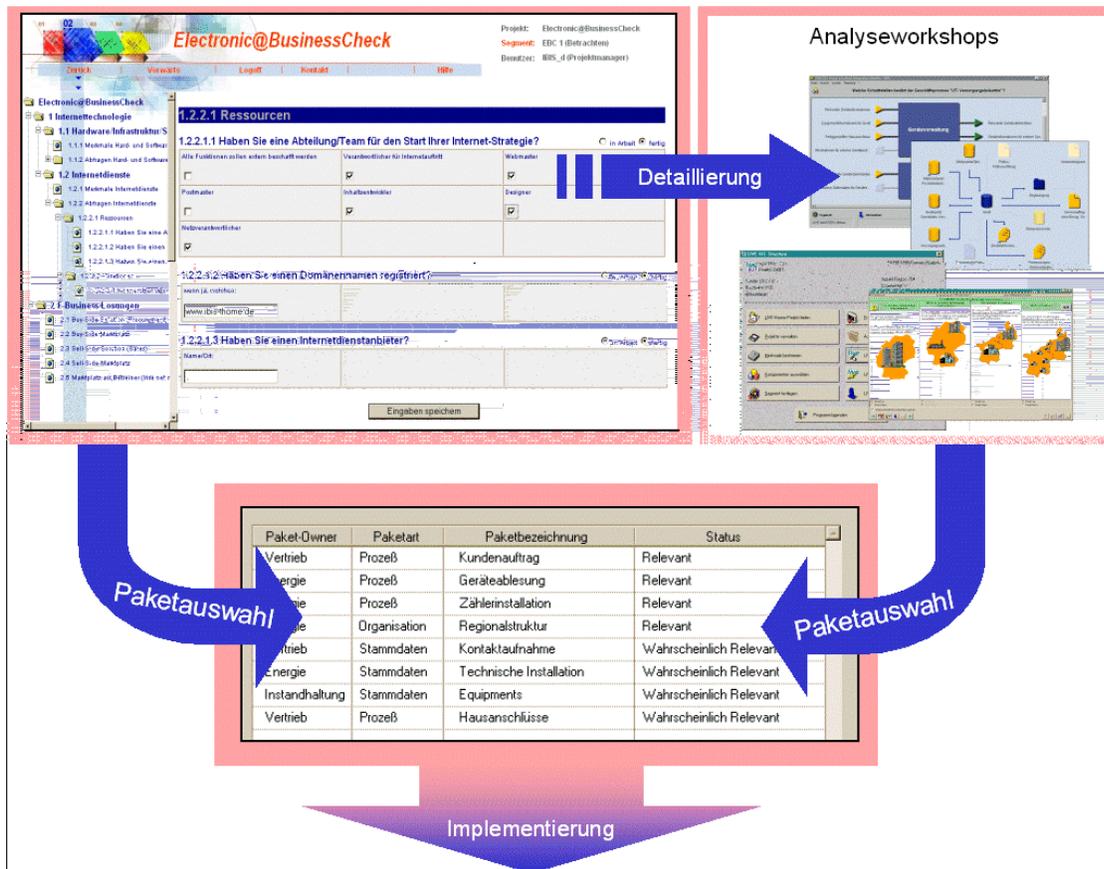


Abbildung 8-3: Integrationsmöglichkeit von THESEUS und IANUS

THESEUS ist somit eine integrierte Erweiterung des bestehenden Instrumentariums zur Implementierung von Standardsoftwarelösungen, mit dessen Hilfe die Koordination der Umsetzungsaktivitäten und die Dokumentation der eingestellten Systemlandschaft transparent wird.

8.3 Gesamtbewertung

Die Leistungsfähigkeit der THESEUS-Methode muss an den im Vorfeld formulierten Zielsetzungen und Anforderungen gemessen werden.

Im folgenden werden die verschiedenen Merkmale und Ziele bewertet:

1. Unterstützung verschiedener Softwarelösungen:

Die dreistufige THESEUS-Architektur mit der Trennung von Zuordnungs- und Projektebene sowie der Entwicklung einer neutralen Adaption-Workbench ermöglicht eine flexible Verknüpfung der betriebswirtschaftlichen Themen mit den jeweils eingesetzten Werkzeugen und Softwarelösungen. Durch

den direkten Zugriff auf die Customizingwerkzeuge, z. B. den Einführungsleitfaden aus dem SAP-System, können die Parametereinstellungen bzw. Tabelleneinträge ausgehend von einem Wissenspaket softwareübergreifend dokumentiert und gespeichert werden (siehe Kapitel 4.1.3 und 6.4.2).

2. Einbeziehung unterschiedlicher Vorgehensmodelle:

Die flexible Gestaltung des Datenmodells durch die normalisierte Ablage der Verknüpfungsinformationen in einer separaten Tabelle ermöglicht neben der Integration unterschiedlicher Softwarelösungen die Nutzung verschiedener Vorgehensmodelle für die Implementierung. Analog zur offenen Zuordnung der Customizingwerkzeuge werden auch die verschiedenen Werkzeuge der einzelnen Vorgehensmodelle den jeweiligen Themen zugewiesen. Somit dienen unterschiedliche Informationsquellen für die Bestimmung der Paketausprägungen als Basis, z. B. die Q&Adb aus ASAP oder die Checklisteneinträge aus RUP. Die Engineering-Komponente der Workbench gewährleistet jederzeit eine Erweiterung des Fundamentalmodells, um neue Analysewerkzeuge zu berücksichtigen (siehe Kapitel 4.1.3 und 6.4.2).

3. Definition betriebswirtschaftlicher Themen:

Die neutralen Wissenspakete sind nicht abhängig von einzelnen Softwarelösungen. Sie dienen als zentrale Wissensträger über den gesamten Lebenszyklus eines Systems. Die zu einem Paket hinterlegten Informationen können jederzeit analysiert werden, wobei historische Entwicklungen bzw. Änderungen nachvollziehbar sind. Der Inhalt eines Wissenspakets orientiert sich dabei nur an betriebswirtschaftlichen Aufgabenstellungen bzw. Prozessen (siehe Kapitel 4.1.1 und 4.3).

4. Entwicklung einer transparenten Navigationsumgebung:

Ein wichtiger Bestandteil der THESEUS-Methode ist es, dem Anwender die Adaptionsergebnisse und die Systemeinstellungen entsprechend präsentieren zu können. Dies trägt wesentlich dazu bei, dass die Akzeptanz und Identifikation mit dem eingestellten System gefördert wird. Zudem wird durch die konsequente Unterstützung der stetigen Systemanpassung im Sinne des CSE die Bereitschaft für Weiterentwicklungen, sowohl aus organisatorischer als auch aus informationstechnischer Sicht, erhöht. Die Ausrichtung und Gestaltung der entwickelten Workbench unterstützen zudem die intuitive Bedienbarkeit des Werkzeugs. Dabei hilft der Rückgriff auf bekannte Techniken zusätzlich die

Arbeit mit THESEUS zu erleichtern, als Beispiel wird auf die Unterteilung in eigenständige Frames oder die Aufbereitung der Inhalte über einen Web-Browser verwiesen. Die strenge Ausrichtung auf die zentralen Wissenspakete und die geforderte Stabilität dieser Elemente schaffen die notwendige Transparenz, da alle Informationen zu genau einem Paket abruf- und gegebenenfalls modifizierbar sind (siehe Kapitel 5.2 bis 5.5).

5. Offenheit gegenüber Veränderungen:

Der Grundgedanke von THESEUS basiert auf der Anpassung eines allgemein gültigen Fundamentalmodells durch die Überführung desselben in ein projektabhängiges Individualmodell mit spezifischen Ausprägungen. Die postulierte Offenheit des Ansatzes bedingt die Möglichkeit, Änderungen am Fundamentalmodell vorzunehmen und somit das Basisprojekt gemäß den unternehmensspezifischen Umgebungsvariablen anzupassen. Nur auf diese Art ist die Berücksichtigung von unterschiedlichen Softwarelösungen auf der einen und verschiedenen Informationsquellen auf der anderen Seite möglich. Das flexible Datenmodell, insbesondere die normalisierte Abbildung der Objekttypen, in Verbindung mit der Engineering-Komponente der Workbench schafft die Voraussetzungen für derartige Anpassungen (siehe Kapitel 6.2.1.3 und 6.4.2).

Diese Bewertung zeigt, dass durch die spezifische betriebswirtschaftliche Definition von neutralen Wissensträgern, zusammen mit einer flexiblen und anwenderfreundlichen Arbeitsumgebung, eine Basis geschaffen wurde, die es den Unternehmen ermöglicht, auch heterogene Softwarelandschaften werkzeugunterstützt zu implementieren bzw. zu betreuen. Somit wird THESEUS den formulierten Ansprüchen gerecht.

Mit der vorliegenden Arbeit werden die bisher bestehenden Ansätze zur Nutzung und Verwaltung der Adaptionsergebnisse, die Modul- bzw. Prozessorientierung, um eine weitere Strategie ergänzt. Die stringente Ausrichtung an betriebswirtschaftlichen Themen schafft die Grundlage für eine applikationsübergreifende Sammlung und Verwertung von Wissen, wie sie mit dem bisherigen Instrumentarium nicht möglich war.

Anhang A Objekttypen (TAB_OBJ_TYP)

Objekttyp	Bezeichnung
StPAK	Stammdatenpaket
PrPAK	Prozesspaket
FuPAK	Funktionspaket
BePAK	Berichtspaket
BaPAK	Basispaket
ScPAK	Schnittstellenpaket
OrPAK	Organisationspaket
LKS_FB	LIVE KIT Structure Fachbereich
LKS_KP	LIVE KIT Structure Komponente
LKS_REDU	LIVE KIT Structure Reduktionsfrage
LKS_PROF	LIVE KIT Structure Profil
LKS_KALK	LIVE KIT Structure Aufwandsvariable
LKP_GP	LIVE KIT Power Geschäftsprozess
LKP_KP	LIVE KIT Power Kernprozess
LKP_LOE	LIVE KIT Power Schnittstelle
LKP_PB	LIVE KIT Power Prozessbeleg
LKP_CB	LIVE KIT Power Containerbeleg
LKP_FKT	LIVE KIT Power Funktion
LKP_AP	LIVE KIT Power Arbeitsplatz
LKC_BH	LIVE KIT Control Berichtshierarchie
LKC_BE	LIVE KIT Control Bericht
RAS_ME	RASTER Merkmal
RAS_MA	RASTER Merkmalsausprägung
RBE	RBE-Set
IMG_SAP	IMG-Baustein des SAP-Systems
HLP_SAP	SAP Online-Dokumentation
ZO	Zuordnungsdokumentation
BCS	Business Configuration Set
HTML_IN	Internes HTML-Dokument zur Beschreibung der Systemeinstellungen

Anhang B Wissenspakete der Versorgungsindustrie

Die folgende Tabelle zeigt die im Rahmen des Prototypen entwickelten Wissenspakete für das Anwendersegment der Versorgungsindustrie. Zu jedem Paket sind die zugeordneten Werkzeugelemente aufgelistet, um die Logik der Zuordnungsebene von THESEUS zu verdeutlichen.

Wissenspaket	LKS (Parameter-ID)	LKP (Prozessbeleg-ID)	KALK (DL-Variable)
Organisation	81ut/01sgd/10pos	35019	DLUT01_POST
	81ut/01sgd/11grp	35021	DLUT01_POL
	81ut/01sgd/12ber	35023	DLUT01_INT
	81ut/01sgd/13wdg	35039	DLUT01_DRUCK
	81ut/01sgd/14whg		DLUT01_HAERTE
	81ut/01sgd/27skz		DLUT01_REGIO
	81ut/01sgd/30kok		DLUT01_KONTO
	81ut/01sgd/31inf		DLUT01_KENN
	81ut/01sgd/36fel		DLUT01_SACH
Geräteablesung und Terminsteuerung	81ut/02vab/010ab	35051	DLUT02_BASIS
	81ut/02vab/020tu	35027	DLUT02_DYNAM
	81ut/02vab/030er	35053	DLUT02_PORT
	81ut/02vab/040ma	35055	DLUT02_ABLESE
	81ut/02vab/050au	35063	DLUT06_SPERR
	81ut/02vab/060ho	35061	DLUT06_REIHE
	81ut/02vab/07pla	35069	DLUT06_PLAUS
	81ut/02vab/080pl	35025	DLUT06_AUSSEN
	81ut/02vab/20rei		DLUT06_GEW
	81ut/02vab/520ge		

Wissenspaket	LKS (Parameter-ID)	LKP (Prozessbeleg-ID)	KALK (DL-Variable)
IS-U Stammdaten	81ut/01sgd/52spa	35043	DLUT03_SPANN
	81ut/01sgd/40ver	35037	DLUT03_STELLE
	81ut/01sgd/51ana	35035	DLUT03_ANLAGE
	81ut/01sgd/32ins		DLUT03_INST
	81ut/01sgd/50sic		DLUT03_GRUE
	81ut/01sgd/60mig		DLUT03_SD
	81ut/01sgd/61dat		DLUT03_MIGR DLUT03_AUS
Technische Geräteverwaltung	81ut/02ger/02sort	35045	DLUT04_ANZ
	81ut/02ger/05anz		DLUT04_ZKENN
	81ut/02ger/06ken		DLUT04_ZART
	81ut/02ger/07art		DLUT04_ZBEZ
	81ut/02ger/08bez		
	81ut/02ger/12art		
Geräteinstallation	81ut/02ger/42umb	35041	DLUT05_TEMP
	81ut/02ger/44ate	35045	DLUT05_INST
	81ut/02ger/40gru		DLUT05_GRUPP
	81ut/02vab/51tem		DLUT05_EIN
Geräteprüfung	81ut/02ger/50pru	35049	DLUT07_BEGL
	81ut/02ger/60pru	35047	DLUT07_TURNUS
	81ut/02ger/65tur		DLUT07_UPLAUS
	81ut/02ger/710bg		DLUT07_APLAUS
	81ut/02ger/720gl		DLUT07_PM
	81ut/02ger/80auf		
	81ut/02ger/81ext		
	81ut/02vab/07pla		

Wissenspaket	LKS (Parameter-ID)	LKP (Prozessbeleg-ID)	KALK (DL-Variable)
Vertragsabrechnung	81ut/03ver/010ve	35069	DLUT08_RABATT
	81ut/03ver/01kla	35071	DLUT08_SIM
	81ut/03ver/12typ	35067	DLUT08_AUS
	81ut/03ver/21bka	35073	DLUT08_IND
	81ut/03ver/010ve	35065	DLUT08_KONZ
	81ut/03ver/10bau	11830	DLUT08_ABRKLA
	81ut/03ver/30rab		DLUT08_GAS
	81ut/03ver/400ga		DLUT08_BAU
	81ut/03ver/50kon		DLUT08_TYP
	81ut/03ver/510ko		DLUT08_BRENN
	81ut/03ver/600ma		
	81ut/03ver/60sim		
	81ut/03ver/610si		
	81ut/03ver/65aus		
	81ut/03ver/660au		
81ut/03ver/67aus			
81ut/03ver/70var			
Fakturierung	81ut/04fak/100st	38620	DLUT09_ABSCHLAG
	81ut/04fak/20apl	35135	DLUT09_AUS
	81ut/04fak/300av		DLUT09_SIM
	81ut/04fak/400fz		
	81ut/04fak/50ast		
	81ut/04fak/510au		
	81ut/04fak/60sim		
	81ut/04fak/70paw		
Kundenservice	81ut/05kse/10htm	85	DLUT10_KONTAKT
	81ut/05kse/20kon	87	DLUT10_RICH
	81ut/05kse/25pri	88	DLUT10_WECH
	81ut/05kse/27ric	10191	DLUT10_ABSPERR
	81ut/05kse/28zus	10197	DLUT10_ZUSATZ
	81ut/05kse/29ver	35090	DLUT10_OUT
	81ut/05kse/30for	35092	DLUT10_SCEDUL

Wissenspaket	LKS (Parameter-ID)	LKP (Prozessbeleg-ID)	KALK (DL-Variable)
	81ut/05kse/40var 81ut/05kse/45bef 81ut/05kse/700in 80ins/07ser/010so 80ins/07ser/020sv 80ins/07ser/030re 80ins/07ser/05ins 80ins/07ser/10loe	35094 35096 45332	DLUT10_GIS
Informationssystem	81ut/09inf/010be 81ut/09inf/20fo 81ut/09inf/20for 81ut/07ber/01nrk 81ut/07ber/20suc 81ut/07ber/30not 81ut/07ber/500dr		DLUT50_BESTAND DLUT50_VORGANG DLUT50_VERKAUF

Anhang C THESEUS-Navigator (Projekt-Wizard)

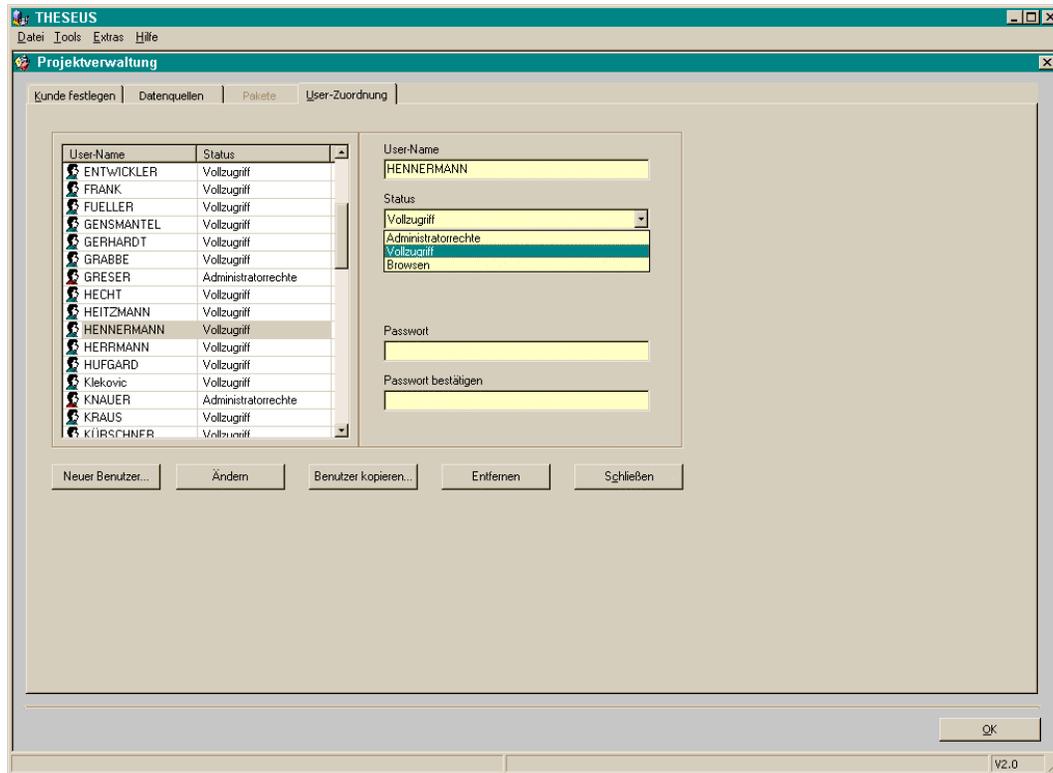
Die folgenden Abbildungen zeigen den im Rahmen dieser Arbeit entwickelten Prototyp. Der Schwerpunkt der Darstellung liegt auf dem Projekt-Wizard, der den Anwender zielgerichtet zur Komposition der Projektstruktur führt. Im einzelnen werden folgende Schritte präsentiert:

1. Festlegen der Projekt- und Kundendaten,
2. Zuordnung der Benutzer und Vergabe der Berechtigungen,
3. Auswahl der zur Verfügung stehenden Informationsquellen und
4. Selektion der relevanten Wissenspakete.

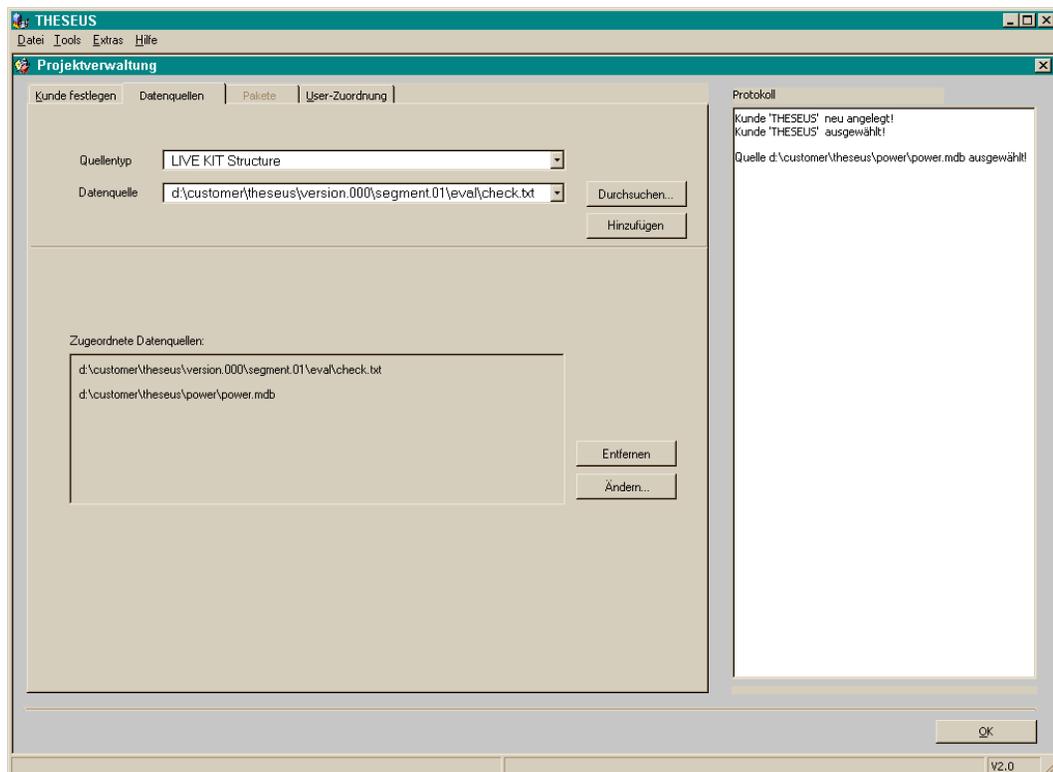
Nach diesen Schritten schließt sich die Komposition der projektspezifischen Navigationsstruktur an, wie sie in Kapitel 4.1 beschrieben ist.

The screenshot shows the 'THESEUS' application window with the 'Projektverwaltung' (Project Management) sub-window active. The 'Kunde festlegen' (Set Customer) tab is selected. The main area contains a text box with instructions: 'Um einen neuen Kunden anzulegen, geben Sie in das Feld "Kunde" den Namen des Kunden ein und klicken Sie anschließend auf "Kunde speichern". Sie können auch einen vorhandenen Kunden aus der Liste wählen, dessen Daten ändern oder mit "Kunde löschen" diesen löschen.' Below this are several input fields: 'Kunde:' (dropdown menu with 'THESEUS' selected), 'Name:', 'Straße:' (Mergentheimer Str. 29), 'PLZ / Ort:' (97082 Würzburg), 'Land:', 'Kunden-Nr.:', 'Telefon:' (0931/12345), 'Fax:', and 'E-Mail:' (THESEUS@bis-thome.de). There are 'Kunde speichern' and 'Kunde löschen' buttons. On the right, a 'Protokoll' (Log) window shows: 'Kunde "THESEUS" neu angelegt!' and 'Kunde "THESEUS" ausgewählt!'. The 'OK' button is at the bottom right, and the version 'V2.0' is in the bottom right corner of the application window.

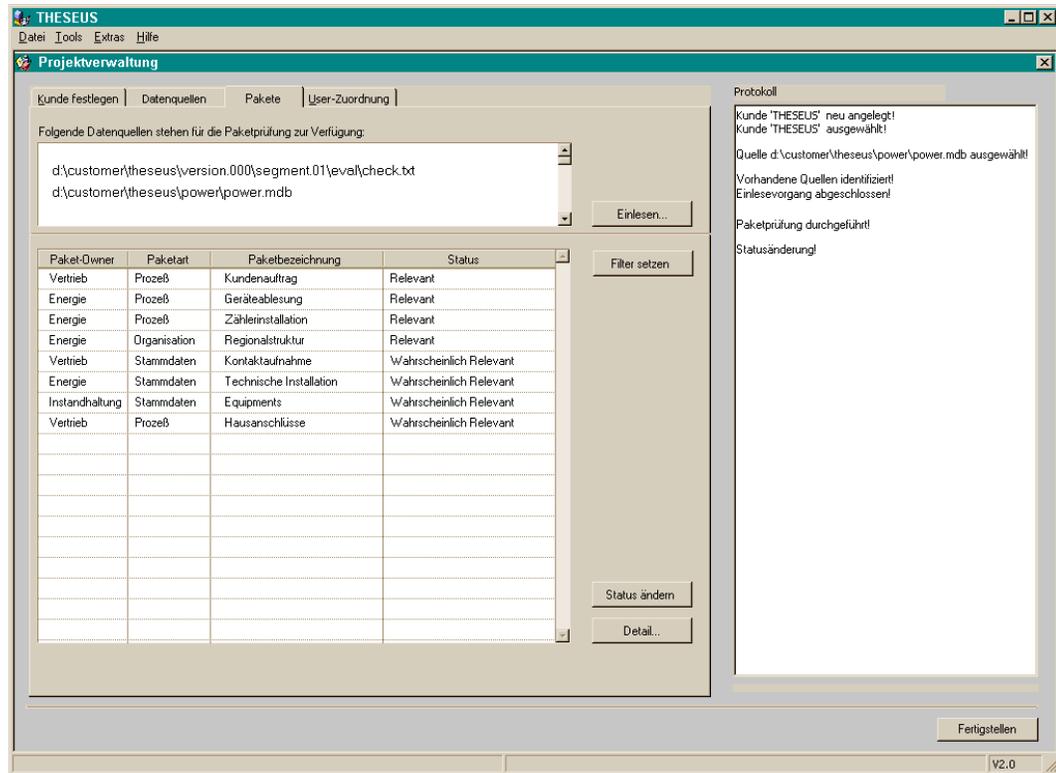
THESEUS-Navigator: Kunde festlegen



THESEUS-Navigator: User-Zuordnung

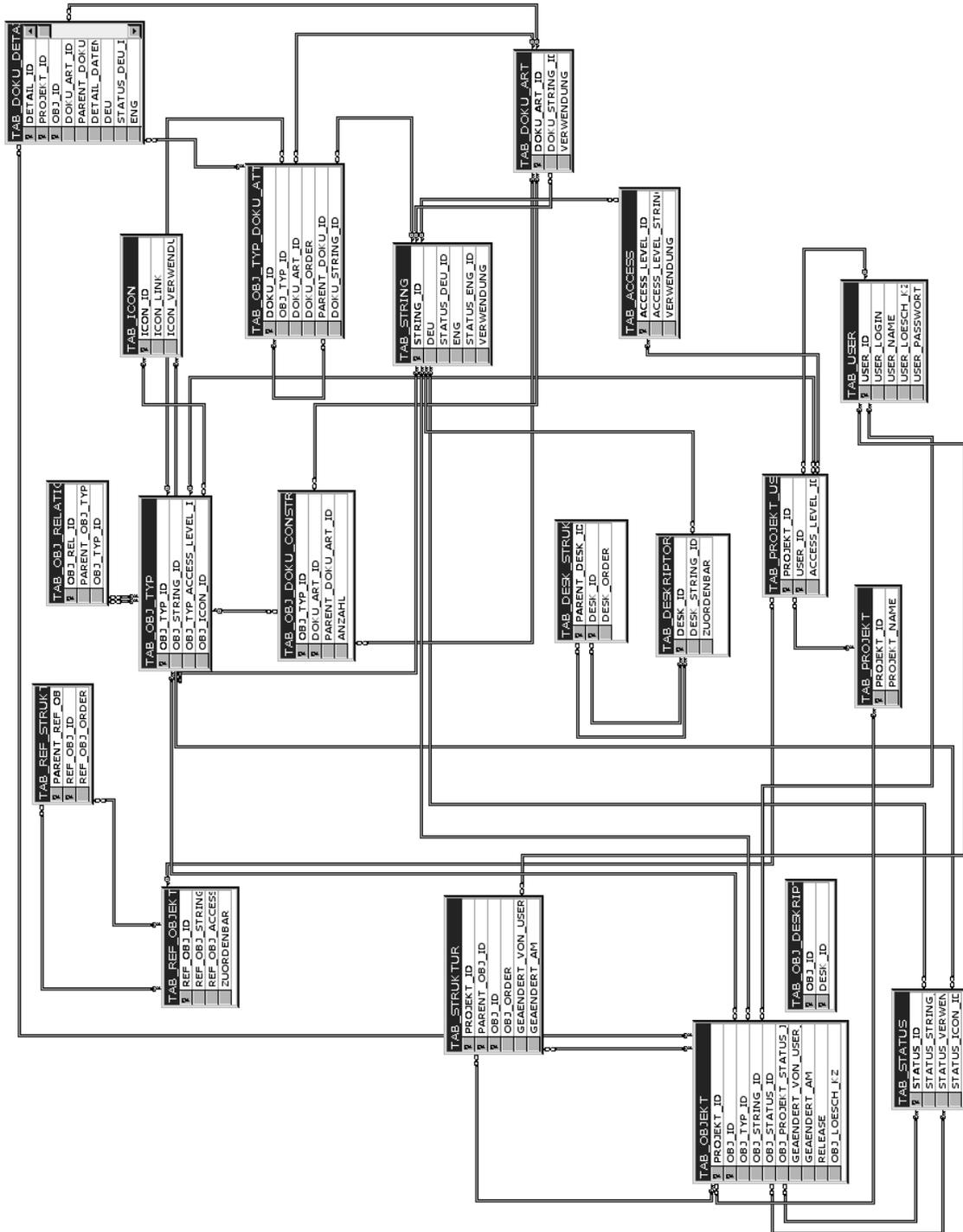


THESEUS-Navigator: Datenquellen



THESEUS-Navigator: Wissenspakete

Anhang D THESEUS-Datenmodell



Literaturverzeichnis

- ALBA93 Albach, H.:
Wirtschaftswissenschaften. In: Gabler Wirtschafts-Lexikon, 13. Auflage. Gabler, Wiesbaden 1993.
- ALBR93 Albrecht, F.:
Strategisches Management der Unternehmensressource Wissen – inhaltliche Ansatzpunkte und Überlegungen zu einem konzeptionellen Gestaltungsrahmen. Frankfurt a. M. 1993.
- APPE00 Appelrath, H.-J.; Ritter, J.:
R/3-Einführung, Methoden und Werkzeuge. Springer-Verlag, Berlin, Heidelberg 2000.
- BÄTZ00 Bätz, V.:
Internetbasierte Abwicklung von Consultingprojekten und –analysen im Umfeld betriebswirtschaftlicher Softwarebibliotheken. Unveröffentlichtes Manuskript am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg 2000.
- BÄTZ94 Bätz, C.:
Adaption und Anwendung der Zeitwirtschaft im Personalwesen einer Standardanwendungssoftware an betriebswirtschaftliche Anforderungen. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1994.
- BÄTZ96 Bätz, V.:
Adaptionsstrategie für ein Standardanwendungsmodul zur prozeßorientierten Fertigung. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1996.

- BÄTZ99 Bätz, C.:
Systematische Gestaltung und kontinuierliche Anpassung von Organisationsstrukturen bei der Anwendung betriebswirtschaftlicher Softwarebibliotheken. Organisationsgestaltung und dynamische Adaption, Veröffentlichte Dissertation am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg 1999.
- BÄTZ99a Bätz, C.:
CSAP – Continuous SAP: Kontinuierliche betriebswirtschaftliche Entwicklung. In: Beschleunigte R/3-Einführung und Continuous Change. Begleitmaterial zur SAP-IBIS Tagung in Würzburg vom 25. bis 26.01.1999, Würzburg 1999.
- BEIN00 Beinhauer, M.; Markus, U.:
From Hypertext to Hyperknowledge – Konzepte zum interaktiven Wissensmanagement. In: Scheer, A.-W. (Hrsg.): E-Business – Wer geht? Wer bleibt? Wer kommt?. 21. Saarbrücker Arbeitstagung 2000 für Industrie, Dienstleistung und Verwaltung. Heidelberg, 2000, S. 307-327.
- BERN97 Berndl, I.:
Adaptionsstrategie für Prozeßebenen eines Standardanwendungsmoduls im Vertrieb. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1997.
- BÖRN95 Börner, C.:
Adaption einer Standardanwendungssoftware für ausgewählte Prozesse des Vertriebs. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1995.

- BOLL99 Boll, M.:
Knowledge Management bei der Steeb Anwendungssysteme GmbH.
In: Teufel, T.: SAP R/3 Prozeßanalyse mit Knowledge Maps. Von
einem beschleunigten Business Engineering zum organisatorischen
Wissensmanagement. Addison Wesley Longmann Verlag, München,
1999, S. 181-187.
- BROK00 o. V.: Brokat AG:
e-Commerce Anwendungen. Informationsabfrage am 3.10.2000.
<http://www.brokat.com/de/products/uebersicht/index.html>.
- BROS94 Brosch, G.:
Adaptionsstrategien für ein Standardanwendungsmodul zur Material-
wirtschaft. Unveröffentlichte Diplomarbeit am Lehrstuhl für
Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1994.
- BUZA97 Buzan, T; Buzan, B.:
Das Mind-map-Buch. Landsberg am Lech 1997.
- CHES99 o. V.: SNI AG (Hrsg.):
Chestra. For Companies On The Move. Productivity Toolkit for
Chestra V3.11C/SNI 1.1. CD-ROM, Release 1.1, München 1999.
- CLAR00 o. V.: Clarify, a Nortel Networks Company:
Industry Solutions. Informationsabfrage am 03.10.2000.
<http://www.clarify.com/solutions>.
- COHE97 Cohen, D. et al.:
Managing Knowledge for Business Success. In: The Conference
Board, Report Number: 1194-97-CH, New York 1997.
- DEBI00 o. V.: debis Systemhaus GmbH:
debis Systemhaus. Informationsabfrage am 03.10.2000
<http://www.debis.de/debis/systemhaus>.

- ESSW92 Esswein, W.:
Das Rollenmodell der Organisation: Berücksichtigung
aufbauorganisatorischer Regelungen in Unternehmensmodellen. In:
Augsburger, W.; Sinz, E. j. (Hrsg): Bamberger Beiträge zur
Wirtschaftsinformatik, Nr. 14. Bamberg 1992
- FRIE93 Fries, K.:
Projektorganisation und Stammdatenanpassung zur Einführung von
Standardsoftware für die Finanzbuchhaltung. Unveröffentlichte
Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und
Wirtschaftsinformatik, Universität Würzburg 1993.
- GENT99 Gentsch, P.:
Wissen managen mit innovativer Informationstechnologie. Gabler,
Wiesbaden 1999.
- HAN98 Hansen, H.-R.:
Wirtschaftsinformatik I, 7. Auflage, UTP 1998.
- HECH96 Hecht, H.:
Adaptionsstrategie eines Standardanwendungsmoduls für ausge-
wählte Berichte des Finanzwesens. Unveröffentlichte Diplomarbeit
am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik,
Würzburg 1996.
- HENN96 Hennermann, F.:
Adaptionsstrategie für ein Standardanwendungsmodul der Instand-
haltung. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebs-
wirtschaftslehre und Wirtschaftsinformatik, Würzburg 1996.
- HESE94 Hesel, W.:
Adaptionsstrategie für ein Standardanwendungsmodul zur Lagerver-
waltung. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebs-
wirtschaftslehre und Wirtschaftsinformatik, Würzburg 1994.

- HILL97 Hillier, D.; Mezick, D.:
Active Server Pages - Programmierung, Microsoft Press Deutschland,
Unterschleißheim 1997.
- HUFG94 Hufgard, A.:
Betriebswirtschaftliche Softwarebibliotheken und Adaption:
Empirischer Befund, Produkte, Methoden, Werkzeuge, Dienst-
leistungen und ein Modell zur Planung und Realisierung im Unter-
nehmen. Vahlen, München 1994.
- HUFG97a Hufgard, A.:
Adaption. In: Mertens, P. et al. (Hrsg.): Lexikon der
Wirtschaftsinformatik. 3. Auflage, Springer, Berlin u. a. 1997, S. 5f.
- IBIS00 o. V.: IBIS Prof. Thome AG:
Electronic@BusinessCheck. Informationsabfrage am 05.10.00.
<http://www.livekit.de>.
- IDS00 o. V.: IDS Prof. Scheer AG (Hrsg.):
ARIS for R/3. Informationsabfrage am 03.10.2000.
<http://www.ids-scheer.de>.
- INTE00 o. V.: Intershop Software Entwicklungs GmbH:
Enfinity – Transactivity Server. Informationsabfrage am 15.10.2000.
[http://www.intershop.de/products/index.htm?callname=
products_enfinity_transactserver](http://www.intershop.de/products/index.htm?callname=products_enfinity_transactserver).
- JDED98 J. D. Edwards Deutschland GmbH (Hrsg.):
Genesis – Composer (Funktionsbeschreibung). Informationsabfrage
am 10.03.01
<http://www.jdedwards.com>
- KELL98 Keller, G.; Teufel, T.:
SAP R/3 prozeßorientiert anwenden. 2. Auflage, Addison Wesley
Longmann Verlag, Bonn 1998.

- KIRC96 Kirchmer, M.:
Geschäftsprozessorientierte Einführung von Standardsoftware.
Wiesbaden 1996.
- KLUK97 Klukas, P.: Noch setzen die Standard-Softwarepakete sich ihre
eigenen Grenzen. In: Computerwoche 22 (1997), S. 48-50.
- KNAU95 Knauer, R.:
Adaptionsstrategie für ein Standardanwendungsmodul zu
ausgewählten Bereichen des Personalwesens. Unveröffentlichte
Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und
Wirtschaftsinformatik, Würzburg 1995.
- KORB95 Korbl, M.:
Adaption einer Standardanwendungssoftware für ausgewählte
Prozesse der Materialwirtschaft. Unveröffentlichte Diplomarbeit am
Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik,
Würzburg 1995.
- LAMB95 Lambrecht, L.:
Adaptionsstrategie für ein Standardanwendungsmodul zu
ausgewählten Bereichen des Vertriebs. Unveröffentlichte
Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und
Wirtschaftsinformatik, Würzburg 1995.
- MEHL98 Mehlich, S.:
Merkmalsorientierte Anforderungsnavigation zur Adaption
betriebswirtschaftlicher Softwarebibliotheken. Veröffentlichte
Dissertation am Lehrstuhl für Betriebswirtschaftslehre und
Wirtschaftsinformatik der Universität Würzburg, Würzburg 1998.
- MICR00 Microsoft Corporation:
Visual Basic. Informationsabfrage am 10.10.2000.
<http://www.microsoft.com/germany/msdn/vbasic>.

- MICR99 Microsoft Corporation:
Microsoft Transact-SQL Referenz, Microsoft Press Deutschland,
Unterschleißheim, 1999.
- MOTZ97 Motz, S.:
Adaptionsstrategie für Prozeßebenen eines Standardanwendungs-
moduls in der Produktionsplanung und -steuerung. Unveröffentlichte
Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und
Wirtschaftsinformatik, Würzburg 1997.
- MÜSS91 Müßig, M.:
Computergestütztes Vorgangssystem für die Projektabwicklung eines
modellbasierten Systementwurfs – COMPOSE. Dissertation,
Universität Würzburg 1991.
- NAVI00 o. V.: Navision Software a/s:
Business Software. Informationsabfrage am 15.10.2000.
http://www.navision.com/de/cat_148.asp.
- NONA97 Nonaka, I.; Takeuchi, H.:
Die Organisation des Wissens. Wie japanische Unternehmen eine
brachliegende Ressource nutzbar machen. Campus Verlag, Frankfurt
a. Main/New York 1997.
- OBER96 Obermann, T.:
Adaption einer Standardanwendungssoftware für ausgewählte
Fertigungsauftragstypen. Unveröffentlichte Diplomarbeit am
Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik,
Würzburg 1996.
- ORAC00 o. V.: Oracle Corporation (Hrsg.):
Industry Applications. Informationsabfrage am 15.10.2000.
<http://www.oracle.com/consulting/services/index.html>.
- PEOP00 o. V.: PeopleSoft GmbH:
PeopleTools und Technologien. 03.10.2000.
<http://www.peoplesoft.de>.

- PFAF00 o. V. Bürgernetz Pfaffenwinkel:
Die Bildung von Symbiosen ist das Fortschrittsprinzip der Evolution.
Informationsabfrage am 11.10.2000
<http://www.pfaffenwinkel.de/marktplatz/natur/sozialer.htm>
- PLAT00 Plattner, H.:
Wir sind laut, aber nicht vorlaut. Interview, Walldorf, 19.06.2000.
<http://www.sapinfo.net/goto/int/2207/DE>.
- PROB97 Probst, G. et al.:
Wissen managen: Wie Unternehmen ihre wertvolle Ressource optimal nutzen. Wiesbaden 1997.
- RATI00 o. V.: Rational Software GmbH:
Rational Unified Process (RUP). Informationsabfrage am 03.10.2000.
<http://www.rational-software.de>.
- ROBI99 Robison, W.:
Microsoft SQL Server in 10 Minutes, Sams Publishing, Indianapolis, August 1999.
- SAP00a o. V.: SAP AG (Hrsg.):
mySAP.com Online Dokumentation, Release 4.6C, CD-Nummer 50037815, Walldorf 2000.
- SAP00b o. V.: SAP AG (Hrsg.):
Einführung in das mySAP.com Knowledge Management, White Paper, Walldorf, 2000.
- SAP00d o. V.: SAP AG (Hrsg.):
ValueSAP Edition 1 for mySAP.com, CD-Nummer 50038774, Walldorf 2000.
- SAP00s o. V.: SAP AG (Hrsg.):
SAP Solution Maps Edition 2000, Informationsabfrage am 03.10.2000.
<http://www.mysap.com/solutionmaps/index.htm>.

- SCHE97 Scheer, A-W.:
Architektur integrierter Informationssysteme. Grundlagen der Unternehmensmodellierung. 2. Auflage, Springer, Berlin u. a. 1997.
- SCHI94 Schipp, O.:
Adaptionsstrategie für ein Standardanwendungsmodul zum Controlling. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1995.
- SCHI99 Schipp, O.:
Betriebswirtschaftliche Vorkonfiguration von Softwarebibliotheken. Veröffentlichte Dissertation am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg, 1999.
- SCHL95 Schlüter, V.:
Adaptionsstrategie für ein Standardanwendungsmodul zu ausgewählten Bereichen der Produktionsplanung und -steuerung. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1995.
- SCHM97 Schmincke, M.:
Ganzheitliche und prozessorientierte Unternehmensgestaltung. In: Klockhaus, E., Scherhuhn, H. (Hrsg.): Modellbasierte Einführung betrieblicher Anwendungssysteme. Gabler, Wiesbaden 1997.
- SCHN94 Schneider, S.:
Adaptionsstrategie für ein Standardanwendungsmodul zum Finanzwesen. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1994.
- SCHO97 Schotter, C.:
Adaptionsstrategie für ein Standardanwendungsmodul im Bereich des Investitionsmanagements. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1997.

- SCHÜ96 Schüppel, J.:
Wissensmanagement – Organisatorisches Lernen im Spannungsfeld von Wissens- und Lernbarrieren, Wiesbaden 1996.
- SEIB97 Seibt, D.:
Vorgehensmodell. In: Mertens, P. et al. (Hrsg.): Lexikon der Wirtschaftsinformatik, 3. Auflage, Springer, Berlin 1997.
- SERV98 Servatius, H.-G.:
Vom Reengineering zum Wissensmanagement. In: Scheer, A.-W. (Hrsg.): Neue Märkte , neue Methoden – Roadmap zur agilen Organisation. 19. Saarbrücker Arbeitstagung 1998 für Industrie, Dienstleistung und Verwaltung. Heidelberg, 1998, S. 323-351.
- SIEB00 o. V.: Siebel Systems, Inc.:
e-Business Applications. Informationsabfrage am 03.10.2000.
<http://www.siebel.com/german/products-solutions/architecture.html>.
- SIED93 Siedler, U.:
Adaption eines Standardsoftwaremoduls zur Personalwirtschaft an ein mittelständisches Unternehmen. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1995.
- SIED99 Siedler, U.:
Beratungs- und Dienstleistungsgeschäft mittels eines elektronischen Marktplatzes. Unveröffentlichte Dissertation am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg, 1999.
- SITZ00 Sitzmann, I.:
Anwendersegmentorientierte Anforderungs- und Geschäftsprozessnavigation einer Standardanwendungssoftware für Banken. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 2000.

- STRE93 Steller, S.:
Eignung von COMET und R/3 in den Bereichen Einkauf und Bestandsführung. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Universität Würzburg 1993.
- STRE99 Steller, S.:
Projektnavigator zur Einführung einer Softwarebibliothek. Marktüberblick, Konzeption und Entwicklung am Beispiel von R/3. Veröffentlichte Dissertation am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg, 1999.
- TEUF99 Teufel T.; Röhricht, J.; Willems, P.:
SAP R/3 Prozeßanalyse mit Knowledge Maps. Von einem beschleunigten Business Engineering zum organisatorischen Wissensmanagement. Addison Wesley Longmann Verlag, München, 1999.
- THHU96 Thome, R.; Hufgard, A.:
Continuous System Engineering, Entdeckung der Standardsoftware als Organisator. Vogel, Würzburg 1996.
- THOM79 Thome, R. et al.:
Vorstudie über Entwicklungsmöglichkeiten von Darstellungsmethoden für Organisations- und Informationsmodelle (DORIN). Unveröffentlichte Studie für den Arbeitskreis für dezentrale Datenverarbeitung. Forschungsbericht der Universität Heidelberg, Heidelberg 1979.
- THOM90 Thome, R.:
Wirtschaftliche Informationsverarbeitung. Vahlen, München 1990.

- THOM98 Thome, R.:
Vom Customizing zur Adaption des Standardsoftwaresystems R/3.
In: Preßmar, D.; Scheer, A.-W. (Hrsg.): SAP R/3 in der Praxis.
Neuere Entwicklungen und Anwendungen. SzU, Band 62, Gabler,
Wiesbaden 1998, S. 45-56.
- TWAR96 Twardy, P.:
Konzeption und Realisierung des Service-Managements mit einer
Standardanwendungssoftware. Unveröffentlichte Diplomarbeit am
Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik,
Würzburg 1996.
- UPDA00 o. V.: Update.com Software AG:
Marketing Manager. Informationsabfrage am 03.10.2000.
<http://www3.update.com/german/index.asp>.
- VOGE97 Vogelsang, E.:
Geschäftsprozeßorientierte Adaptionstrategie für betriebswirtschaft-
liche Softwarebibliotheken. Prozeß-Ebenen-Analyse für Ergänzungs-
entwicklung, Lückenidentifikation und organisatorische Problem-
lösungen. Veröffentlichte Dissertation am Lehrstuhl für Betriebs-
wirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg,
Würzburg 1997.
- WAGN83 Wagner, G.:
Modell betrieblicher Informationsverarbeitung, Frankfurt 1983.
- WALZ00 Walz, W.:
Organisation von Konzerneinführungen durch Adaption von
Softwarebibliotheken. Unveröffentlichte Dissertation am Lehrstuhl
für Betriebswirtschaftslehre und Wirtschaftsinformatik der
Universität Würzburg, Würzburg 2000.

- WALZ94 Walz, W.:
Adaption eines Standardanwendungsmoduls zu speziellen Bereichen des Finanzwesens. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1994.
- WASC96 Waschhauser, T.:
Adaption einer Standardanwendungssoftware für ausgewählte Berichte des Controlling. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik, Würzburg 1996.
- WEDL97 Wedlich, E.:
Entscheidungsorientierte Adaptionstrategie für Berichte einer Softwarebibliothek. Veröffentlichte Dissertation am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg 1997.
- WENZ95 Wenzel, H.:
Adaptionstrategie für ein Standardanwendungsmodul zu ausgewählten Bereichen der Materialwirtschaft. Unveröffentlichte Diplomarbeit am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg 1995.
- WENZ00 Wenzel, H.:
Retrograde heuristische Evaluierung von Systemdaten produktiver Softwarebibliotheken. Methoden und Techniken der Analyse von produktiven betriebswirtschaftlichen Informationssystemen. Unveröffentlichte Dissertation am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik der Universität Würzburg, Würzburg 1999.
- WÖHE00 Wöhe, G:
Einführung in die allgemeine Betriebswirtschaftslehre. 20. Auflage, Verlag Vahlen, München 2000.

Lebenslauf

Angaben zur Person

Name: Frank Hennermann
Geboren: 14.05.1970 in Ochsenfurt
Familienstand: ledig
Staatsangehörigkeit: deutsch

Schulbildung

Sept. 76 - Juli 80 Grundschule Ochsenfurt
Sept. 80 - Juni 90 Gymnasium Marktbreit

Hochschulausbildung

Okt. 91 – Mai. 97 Studium der Betriebswirtschaftslehre an der Bayerischen Julius-Maximilians-Universität in Würzburg
Abschluss als Diplom-Kaufmann (Univ.)
Juni. 97 – Mai. 01 Promotionsstudium der Betriebswirtschaftslehre an der Bayerischen Julius-Maximilians-Universität in Würzburg

Berufliche Tätigkeit

Aug. 97 – Juli 00 Wissenschaftlicher Mitarbeiter am Lehrstuhl für Betriebswirtschaftslehre und Wirtschaftsinformatik von Professor Dr. R. Thome an der Universität Würzburg
Seit Juli 1997 Mitarbeiter der IBIS Professor Thome AG in Würzburg