Aufbau von produktionsnahen Teaminformationsportalen bei kundenindividueller Produktion mittels Entwurfsmustersprachen

Von der Fakultät Konstruktions-, Produktions- und Fahrzeugtechnik
der Universität Stuttgart
zur Erlangung der Würde eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Abhandlung

Vorgelegt von

Dipl.-Inform. Stefan Gerlach

aus Stuttgart

Hauptberichter: Univ.-Prof. Dr.-Ing. Dr.-Ing. E.h. Dieter Spath

Mitberichter: Univ.-Prof. Dr.-Ing. Prof. E.h. Dr.-Ing. E.h. Dr. h.c. mult. Engelbert Westkämper

Tag der mündlichen Prüfung: 11. November 2010

Institut für Arbeitswissenschaft und Technologiemanagement (IAT)
der Universität Stuttgart

2010

IPA-IAO Forschung und Praxis

Berichte aus dem
Fraunhofer-Institut für Produktionstechnik und
Automatisierung (IPA), Stuttgart,
Fraunhofer-Institut für Arbeitswirtschaft und
Organisation (IAO), Stuttgart,
Institut für Industrielle Fertigung und
Fabrikbetrieb (IFF), Universität Stuttgart
und Institut für Arbeitswissenschaft und
Technologiemanagement (IAT), Universität Stuttgart

Herausgeber:
Univ.-Prof. Dr.-Ing. Prof. e.h. Dr.-Ing. e.h. Dr. h.c. mult. Engelbert Westkämper
und
Univ.-Prof. Dr.-Ing. habil. Prof. E.h. mult. Dr. h.c. mult. Hans-Jörg Bullinger
und
Univ.-Prof. Dr.-Ing. Dr.-Ing. E.h. Dieter Spath

Universität Stuttgart
Institut für Arbeitswissenschaft und Technologiemanagement IAT

Fraunhofer IAO

Stefan Gerlach

Aufbau von produktionsnahen Teaminformationsportalen bei kundenindividueller Produktion mittels Entwurfsmustersprachen

Nr. 498

JOST-JETTER VERLAG
Fachverlag · 71296 Heimsheim

Dr.-Ing. Dipl.-Inform. Stefan Gerlach
Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IAO), Stuttgart

Univ.-Prof. Dr.-Ing. Prof. e.h. Dr.-Ing. e.h. Dr. h.c. mult. Engelbert Westkämper
ord. Professor an der Universität Stuttgart
Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA), Stuttgart

Univ.-Prof. Dr.-Ing. habil. Prof. E.h. mult. Dr. h.c. mult. Hans-Jörg Bullinger
ord. Professor an der Universität Stuttgart
Präsident der Fraunhofer-Gesellschaft, München

Univ.-Prof. Dr.-Ing. Dr.-Ing. E.h. Dieter Spath
ord. Professor an der Universität Stuttgart
Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IAO), Stuttgart

D 93

ISBN 978-3-939890-64-5
Jost Jetter Verlag, Heimsheim

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils gültigen Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.
© Jost Jetter Verlag, Heimsheim 2010.
Printed in Germany.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Sollte in diesem Werk direkt oder indirekt auf Gesetze, Vorschriften oder Richtlinien (z. B. DIN, VDI, VDE) Bezug genommen oder aus ihnen zitiert worden sein, so kann der Verlag keine Gewähr für die Richtigkeit, Vollständigkeit oder Aktualität übernehmen. Es empfiehlt sich, gegebenenfalls für die eigenen Arbeiten die vollständigen Vorschriften oder Richtlinien in der jeweils gültigen Fassung hinzuzuziehen.

Druck: printsystem GmbH, Heimsheim

Geleitwort der Herausgeber

Über den Erfolg und das Bestehen von Unternehmen in einer marktwirtschaftlichen Ordnung entscheidet letztendlich der Absatzmarkt. Das bedeutet, möglichst frühzeitig absatzmarktorientierte Anforderungen sowie deren Veränderungen zu erkennen und darauf zu reagieren.

Neue Technologien und Werkstoffe ermöglichen neue Produkte und eröffnen neue Märkte. Die neuen Produktions- und Informationstechnologien verwandeln signifikant und nachhaltig unsere industrielle Arbeitswelt. Politische und gesellschaftliche Veränderungen signalisieren und begleiten dabei einen Wertewandel, der auch in unseren Industriebetrieben deutlichen Niederschlag findet.

Die Aufgaben des Produktionsmanagements sind vielfältiger und anspruchsvoller geworden. Die Integration des europäischen Marktes, die Globalisierung vieler Industrien, die zunehmende Innovationsgeschwindigkeit, die Entwicklung zur Freizeitgesellschaft und die übergreifenden ökologischen und sozialen Probleme, zu deren Lösung die Wirtschaft ihren Beitrag leisten muss, erfordern von den Führungskräften erweiterte Perspektiven und Antworten, die über den Fokus traditionellen Produktionsmanagements deutlich hinausgehen.

Neue Formen der Arbeitsorganisation im indirekten und direkten Bereich sind heute schon feste Bestandteile innovativer Unternehmen. Die Entkopplung der Arbeitszeit von der Betriebszeit, integrierte Planungsansätze sowie der Aufbau dezentraler Strukturen sind nur einige der Konzepte, welche die aktuellen Entwicklungsrichtungen kennzeichnen. Erfreulich ist der Trend, immer mehr den Menschen in den Mittelpunkt der Arbeitsgestaltung zu stellen - die traditionell eher technokratisch akzentuierten Ansätze weichen einer stärkeren Human- und Organisationsorientierung. Qualifizierungsprogramme, Training und andere Formen der Mitarbeiterentwicklung gewinnen als Differenzierungsmerkmal und als Zukunftsinvestition in *Human Resources* an strategischer Bedeutung.

Von wissenschaftlicher Seite muss dieses Bemühen durch die Entwicklung von Methoden und Vorgehensweisen zur systematischen Analyse und Verbesserung des Systems Produktionsbetrieb einschließlich der erforderlichen Dienstleistungsfunktionen unterstützt werden. Die Ingenieure sind hier gefordert, in enger Zusammenarbeit mit anderen Disziplinen, z. B. der Informatik, der Wirtschaftswissenschaften und der Arbeitswissenschaft, Lösungen zu erarbeiten, die den veränderten Randbedingungen Rechnung tragen.

Die von den Herausgebern langjährig geleiteten Institute, das

- Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA),
- Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IAO),
- Institut für Industrielle Fertigung und Fabrikbetrieb (IFF), Universität Stuttgart,
- Institut für Arbeitswissenschaft und Technologiemanagement (IAT), Universität Stuttgart

arbeiten in grundlegender und angewandter Forschung intensiv an den oben aufgezeigten Entwicklungen mit. Die Ausstattung der Labors und die Qualifikation der Mitarbeiter haben bereits in der Vergangenheit zu Forschungsergebnissen geführt, die für die Praxis von großem Wert waren. Zur Umsetzung gewonnener Erkenntnisse wird die Schriftenreihe „IPA-IAO - Forschung und Praxis“ herausgegeben. Der vorliegende Band setzt diese Reihe fort. Eine Übersicht über bisher erschienene Titel wird am Schluss dieses Buches gegeben.

Dem Verfasser sei für die geleistete Arbeit gedankt, dem Jost Jetter Verlag für die Aufnahme dieser Schriftenreihe in seine Angebotspalette und der Druckerei für saubere und zügige Ausführung. Möge das Buch von der Fachwelt gut aufgenommen werden.

Engelbert Westkämper Hans-Jörg Bullinger Dieter Spath

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter im gemeinschaftlichen Betrieb des Instituts für Arbeitswissenschaft und Technologiemanagement (IAT) der Universität Stuttgart sowie des Fraunhofer-Instituts für Arbeitswirtschaft und Organisation (IAO), ebenfalls in Stuttgart.

Herrn Prof. Dr.-Ing. Dr.-Ing. E. h. Dieter Spath, Leiter des Instituts für Arbeitswissenschaft und Technologiemanagement (IAT) der Universität Stuttgart sowie des Fraunhofer-Instituts für Arbeitswirtschaft und Organisation (IAO), gilt meine besondere Dankbarkeit für seine persönliche Betreuung, wohlwollende fachliche Unterstützung und beständige Förderung der Arbeit.

Herrn Prof. Dr.-Ing. Prof. E. h. Dr.-Ing. E. h. Dr. h. c. mult. Engelbert Westkämper, Leiter des Instituts für Industrielle Fertigung und Fabrikbetrieb (IFF) der Universität Stuttgart sowie des Fraunhofer-Instituts für Produktionstechnik und Automatisierung (IPA), danke ich herzlich für die Übernahme des Mitberichts.

Zum Gelingen der Arbeit trugen meine Kollegen am IAT und IAO in vielen Gesprächen bei. Ausdrücklich erwähnen möchte ich die Unterstützung und Inspiration seitens Herrn Dr.-Ing. Wolfgang Schweizer. Den Herren Dipl.-Wirtsch.-Ing. (FH) Manfred Bender und Dipl.-Ing. Peter Rally danke ich für die gemeinsame Arbeit an dem Forschungsprojekt, das der Arbeit das industrielle Umfeld und den geeigneten Rahmen zur Evaluation bot. Mein Kollege und Freund Dipl.-Ing. Oliver Scholtz übernahm die sorgfältige Durchsicht und Fehlerkorrektur des Manuskripts. Dr.-Ing. Jochen Dietrich war mir stets Vorbild und Ansporn für die Fertigstellung der Arbeit. Beiden danke ich dafür herzlich. Ein Dankeschön gebührt auch den Studierenden, die mir in den vergangenen Jahren zuarbeiteten, insbesondere danke ich Herrn cand.-mach. Sebastian Hodapp für seine Geduld und Akribie bei der Anfertigung der Abbildungen und Tabellen.

Vor allem bedanke ich mich ganz besonders bei meiner Frau Hanne Lohse und unseren Töchtern Antonia und Carla. Ohne ihre Liebe, stetige Unterstützung, Motivation und ihr Verständnis, auch in den arbeitsintensiven Phasen der letzten Jahre, wäre diese Arbeit nicht fertig geworden.

Schließlich danke ich von ganzem Herzen meinen Eltern Lore und Hermann, die mir diesen Werdegang erst möglich machten. Ihnen widme ich diese Arbeit.

Stuttgart, im November 2010 Stefan Gerlach

Inhalt

Verzeichnis der Abkürzungen **13**

Verzeichnis der Abbildungen **15**

Verzeichnis der Tabellen **17**

1 Einleitung **18**

1.1 Ausgangssituation und Problemstellung 18

1.2 Zielsetzung, Handlungsbedarf und Lösungsansatz 19

1.3 Gegenstandsbereich und Aufbau der Arbeit 20

2 Produktionsnahe Informationsbereitstellung bei kundenindividueller Produktion **23**

2.1 Formen und Organisation kundenindividueller Produktion 23

2.1.1 Individualisierte Produkte als Unternehmensstrategie 23

2.1.2 Kundenindividuelle Produktion im industriellen Umfeld 25

2.1.3 Produktionsfaktor Information für die kundenindividuelle Produktion 26

2.2 Informationsbedarfe produktionsnaher Teams 28

2.2.1 Informationsraum der kundenindividuellen Produktion 28

2.2.2 Intensität des Informationsbedarfs 30

2.3 Informationsangebot für produktionsnahe Teams 32

2.3.1 Anforderungen an die Informationsbereitstellung 32

2.3.1.1 Arbeitsplatzspezifische Kerninformationen 32

2.3.1.2 Flankierende Komplementärinformationen 33

2.3.1.3 Sporadische Notifikationsfunktionen 33

2.3.1.4 Bidirektionale Kommunikation 34

2.3.1.5 Personalisierte Informationsangebote 34

2.3.1.6 Gewichtung der Anforderungen im Informationsraum 34

2.3.2 Realisierungsformen und Technologien 36

2.3.3 Praxis der produktionsnahen Informationsbereitstellung 37

2.3.4 Schlussfolgerungen aus der Informationsanalyse 39

2.4 Erstellung produktionsnaher Informationsportale 40

2.4.1 Abgrenzung des Untersuchungsbereichs für Portalerstellungsmethoden 40

2.4.1.1 Charakteristik und Merkmale der Domäne Informationsportale 40

2.4.1.2 Elemente von Methoden zur Erstellung von Informationsportalen 41

2.4.2 Werkzeuge zur Realisierung von Informationsportalen 42

2.4.3 Vorgehensweisen zur Portalentwicklung 43

2.4.4 Modelle für das Web-Engineering 44
2.4.5 Konzepte zur Gestaltung von Informationsportalen 44
2.4.5.1 Ordnungsrahmen für Konzepte 44
2.4.5.2 Problemorientierte Konzepte für produktionsnahe Informationsangebote 45
2.4.5.3 Problemorientierte Konzepte für Informationsportale 49
2.4.5.4 Lösungsorientierte Konzepte für Informationsportale 49

2.5 Vergleich und Bewertung der Methoden 50

3 Grundlagen der Entwurfsmuster 54

3.1 Grundlegende Begriffe zu Entwurfsmustern 54
3.1.1 Definition und Wesen von Entwurfsmustern 54
3.1.2 Entwurfsmustersammlungen 56
3.1.2.1 Musterorganisation 56
3.1.2.2 Musterkomposition 57
3.1.2.3 Entwurfsmustersprachen 58
3.1.2.4 Mustersequenzen 59
3.1.2.5 Notation von Entwurfsmustersprachen 59

3.2 Entwurfsmustersprachen für produktionsnahe Informationsportale 60

3.3 Eignung des Lösungsansatzes 63
3.3.1 Alternative Formen konzeptioneller Unterstützung 63
3.3.2 Nutzen von Entwurfsmustern 63
3.3.3 Bewertung und Handlungsbedarf 64

4 Entwurfsmustersprache für produktionsnahe Informationsportale 67

4.1 Übersicht der neuen Entwurfsmustersprache 67
4.1.1 Struktureller Aufbau 67
4.1.2 Fallbeispiel „Asynchrone Benachrichtigung“ 68

4.2 Strukturierter Entwurfsmusterkatalog 71
4.2.1 Entwurfsmustersammlung 71
4.2.2 Ordnungsstruktur des Entwurfsmusterkatalogs 72
4.2.2.1 Primäre Gliederung „Entwicklungsphasen und Sichten“ 72
4.2.2.2 Sekundäre Gliederung „Aufgaben“ 73
4.2.3 Status der Entwurfsmuster 76

4.3 Kooperatives Entwurfsmustersystem 78
4.3.1 Beziehungsarten 78
4.3.2 Notation von Beziehungen 80

4.4 Kohärente und progressive Entwurfsmustersprache 83
4.4.1 Aufgabensequenzen 83
4.4.2 Aufgabensequenz „Portal-Engine erstellen“ 85
4.4.3 Aufgabensequenz „Portalseiten ergänzen“ 86
4.4.4 Fallbeispiel mit Aufgabensequenz 88

4.5 Werkzeug zur Pflege der Entwurfsmustersprache 89

4.5.1 Adaption der Entwurfsmustersprache 89
4.5.2 Musterbank 89

5 Mustersequenzen für kundenindividuelle Auftragsinformationen 92

5.1 Anwendungsbeispiel „Elektrische Schaltschränke“ 93
5.1.1 Kundenindividuelle Produktausführungen 93
5.1.2 Produktionsprozess 94
5.1.3 Bedarf an Auftragsinformationen 95

5.2 Arbeitsplatzspezifische Informationsbereitstellung 96
5.2.1 Analyse des Informationsbedarfs 96
5.2.2 Konzeption des Informationsangebots 99
5.2.3 Implementierung einer Portalseite 102
5.2.4 Lösungsorientierte Mustersequenz 105

5.3 Mitarbeiterspezifische Informationsangebote 108
5.3.1 Grundlagen der Personalisierung von Informationsangeboten 108
5.3.1.1 Benutzermodelle und Profilierung 108
5.3.1.2 Gegenstände der Personalisierung 110
5.3.1.3 Verfahren der Personalisierung 110
5.3.2 Analyse des produktionsnahen Personalisierungsbedarfs 111
5.3.2.1 Personalisierungsmuster 111
5.3.2.2 Benutzerprofil 112
5.3.2.3 Contentprofil 113
5.3.2.4 Personalisierungsschema 114
5.3.3 Konzeption des mitarbeiterspezifischen Informationsangebots 116
5.3.4 Implementierung einer mitarbeiterspezifischen Portalseite 118
5.3.5 Mustersequenz des mitarbeiterspezifischen Informationsangebots 120

5.4 Übertragbarkeit der Mustersequenzen 121

6 Evaluierung in der kundenindividuellen Produktion 122

6.1 Industrielles Umfeld der Evaluierung 122
6.1.1 Charakterisierung der Unternehmen 122
6.1.2 Informationsnutzung in den Unternehmen 123
6.1.3 Einsatzvoraussetzungen für Informationsportale in der Produktion 125

6.2 Fallbeispiel 1: Portalprototyp für kundenindividuelle Schaltschränke 125

6.3 Fallbeispiel 2: Montageinformationsportal eines Anlagenherstellers 129

6.4 Diskussion und Ausblick 132

7 Zusammenfassung 134

8 Summary 136

9 Anhänge 138

9.1 Literaturverzeichnis 138
9.1.1 Quellen der Erhebung des Informationsraums 138
9.1.2 Quellen der Muster des Entwurfsmusterkatalogs 139
9.1.3 Textquellen 141
9.1.4 Bildnachweis 154
9.2 Entwurfsmustersprache für produktionsnahe Informationsportale 155
9.3 Beispiel Entwurfsmuster DECORATOR 174
9.4 Beispiel Mustersequenz 176

Verzeichnis der Abkürzungen

APS	Advanced Planning and Scheduling
ASP	Active Server Pages
B2B	Business to Business
B2C	Business to Consumer
B2E	Business to Employee
BDE	Betriebsdatenerfassung
BO	Business Objects
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CAP	Computer Aided Planning
CAPP	Computer Aided Process Planning
CAQ	Computer Aided Quality
CNC	Computerized Numerical Control
CMS	Content-Management-System
CORBA	Common Object Request Broker Architecture
CRM	Customer Relationship Management
DB	Datenbank
DCI	Decoupled Context Interface
DIN	Deutsche Industrie Norm
DMZ	Demilitarized Zone
E-Business	Electronic Business
E-Com	Electronic Commerce
E-Commerce	Electronic Commerce
E-Mail	Electronic Mail
EAI	Enterprise Application Integration
ECO	Encapsulated Context Object
EDI	Electronic Data Interchange
eMeeting	Electronic Meeting
ER	Entity-Relationship
ERP	Enterprise Resource Planning
FAQ	Frequently Asked Questions
FLS	Fertigungsleitstand
HCI	Human Computer Interface
HTML	Hypertext Markup Language
IT	Informationstechnik
KIP	Kundenindividuelle Produktion
KISS	Keep It Simple and Stupid
KVP	Kontinuierlicher Verbesserungsprozess
MTS	Make to Stock
MTO	Make to Order
MVC	Model-View-Control Muster
NC	Numerical Control
OOHDM	Object-Oriented Hypermedia Design Method

PDCA	Plan-Do-Check-Act
PDF	Portable Document Format
PDM	Produktdatenmanagement
PHP	Hypertext Preprocessor
PKI	Public Key Infrastructure
PPS	Produktionsplanung und –steuerung
QIS	Qualitätsinformationssystem
QM	Qualitätsmanagement
QS	Qualitätssicherung
RPC	Role Partitioned Context
RSCO	Role Specific Context Object
SCM	Supply Chain Management
SOAP	Simple Object Access Protocol
SW	Software
TQM	Total Quality Management
UML	Unified Modelling Language
UWE	UML-based Web Engineering
WAE	Web Application Extension
WE	Web Engineering
WebML	Web Modelling Language
WSDL	Web Service Definition Language
XML	Extended Markup Language

Verzeichnis der Abbildungen

Abbildung 1-1: Aufgabenzentriertes Informationsangebot 19
Abbildung 1-2: Aufbau der Arbeit 22
Abbildung 2-1: Zeitpunkte der Kundenintegration. Nach: [Stotko03, S. 85] 24
Abbildung 2-2: Modularisierungsprinzipien und Baukastensysteme. Nach: [Piller06, S. 229] 24
Abbildung 2-3: Organisation des Produktionsablaufs. Nach: [Wikner07, S. 146] 25
Abbildung 2-4: Produktionsfaktoren mit Informationsanteilen. Nach: [Piller06, S. 21] 26
Abbildung 2-5: Beispiel einer generisch-quantitativen Modularisierung. Bild: [Festo AG] 27
Abbildung 2-6: Beispielhafter Aufwand für Informationsbeschaffung. Quelle: [Kammermeier09, S. 268] 27
Abbildung 2-7: Portfolio der Intensität des Informationsbedarfs. Nach: [Spath09b] 31
Abbildung 2-8: Anforderungen an das Informationsangebot im Informationsraum 35
Abbildung 2-9: Informationsarchitektur der Auftragsabwicklung. Nach: [Mertens04, S. 267] 36
Abbildung 2-10: Anwendungsbereiche von Unternehmensportalen 37
Abbildung 2-11: Distribution und Medien. Quellen: [Spath03b, S. 31ff; Lang07, S. 30f]. 38
Abbildung 2-12: Elemente einer unterstützenden Methode. Nach: [Schütze99, S. 18] 41
Abbildung 2-13: Basisdienste von Portalanwendungen. Nach: [Huang03] 42
Abbildung 2-14: Aufgaben und Sichten der Softwareentwicklung. Nach: [Hahsler01, S. 18] 43
Abbildung 2-15: Domänenprofil für Informationsportale. Nach: [Austaller04] 45
Abbildung 2-16: Beispielhafte Portalarchitektur. Nach: [Galic04, S. 90] 49
Abbildung 2-17: Beispiel für ein lösungsorientiertes Konzept. Nach: [Goodyear00, S. 17/31] 50
Abbildung 2-18: Defizit bei der Realisierung produktionsnaher Informationsportale 53
Abbildung 3-1: Musterbeispiel aus der Architektur. Quelle: [Alexander77, S. 904] 54
Abbildung 3-2: Balance von Zielkonflikten durch Musterlösungen. Quelle: [Keller99] 55
Abbildung 3-3: Ausschnitt Beziehungsdiagramm. Nach: [Gamma96, S. 16] 57
Abbildung 3-4: Vergleich von Musterbeziehungen und Mustersequenzen. Nach: [Henney06, S. D1-25] 59
Abbildung 3-5: Beispiel einer formalen Notation von Mustersequenzen. Nach: [Henney06, S. D1-31] 60
Abbildung 3-6: Analysemuster OBSERVATION. Nach: [Fowler97, S. 42] 61
Abbildung 3-7: Usability-Muster OVERVIEW BY DETAIL. Quelle: [vanWelie07] 61
Abbildung 3-8: Übersicht der bekannten Mustersprachen für Informationsportale 62
Abbildung 3-9: Vergleich von Lösungszeit und Lösungsqualität. Quelle [Prechelt02] 64
Abbildung 3-10: Sprachübergreifende Musterauswahl und Kombination 65
Abbildung 4-1: Vorgehensweise zur Entwicklung der neuen Mustersprache 67
Abbildung 4-2: Entwurfsmuster für „Asynchrone Benachrichtigung“ 69
Abbildung 4-3: Informationsangebot „Asynchrone Benachrichtigung“ 70
Abbildung 4-4: Duale Musterkategorisierung 72
Abbildung 4-5: Kooperationsbeziehungen der Mustersprache 80
Abbildung 4-6: Beispielhafter Auszug einer Aufgabensequenz 84
Abbildung 4-7: Aufgabensequenzen in der Entwurfsmustersprache 85
Abbildung 4-8: Aufgabensequenz des Anwendungsbeispiels „Asynchrone Benachrichtigung“ 88
Abbildung 4-9: Struktur der Musterbank in UML-Notation 89

Abbildung 4-10: Praxisorientierte Umsetzung der Musterbank in einer Tabellenkalkulation 90
Abbildung 5-1: Beispiele von elektrischen Schaltschränken und Zubehören. Bild: [Rittal GmbH & Co. KG] 93
Abbildung 5-2: Rasterprofile für den individuellen Schaltschrankausbau. Bild: [Rittal GmbH & Co. KG] 93
Abbildung 5-3: Schematisierter Produktionsprozess für individualisierte Schaltschränke. 94
Abbildung 5-4: Mustersequenz „Portalseiten ergänzen“ 96
Abbildung 5-5: Analysemuster Produktionsauftrag und Observation. Nach: [Hay96; Fowler97] 97
Abbildung 5-6: Beispielhafte Observation von kundenindividuellen Auftragsoptionen 97
Abbildung 5-7: Modellierung kundenindividueller Produktionsaufträge. 98
Abbildung 5-8: Gliederung des Informationsangebots und Navigationselemente 100
Abbildung 5-9: Darstellung der Auftragsoptionen 101
Abbildung 5-10: Implementierungsbedingter Einsatz von Entwurfsmustern zur Datenmodellierung 103
Abbildung 5-11: Beispielhaftes Objektdiagramm zum implementierten Datenmodell. 104
Abbildung 5-12: Einheitliche Darstellung und Verarbeitung von Baukasten- und Auftragsstücklisten 105
Abbildung 5-13: Wiederverwendung von Mustersequenzen. 107
Abbildung 5-14: Formen der Personalisierung und Benutzermodellierung. Nach: [Mertens04, S. 5] 108
Abbildung 5-15: Bestandteile eines Benutzermodells. Nach: [Kobsa01, S. 115ff] 109
Abbildung 5-16: Analysemuster Personalized Content. Nach: [vanDuyne02, S. 272] 111
Abbildung 5-17: Routine-Stereotype für die Benutzerprofilierung 113
Abbildung 5-18: Analysemuster zur Contentprofilierung 114
Abbildung 5-19: Seitennavigation im personalisierten Informationsangebot 116
Abbildung 5-20: Präsentation der personalisierten Informationsinhalte 117
Abbildung 5-21: Implementierung der Personalisierung mit Entwurfsmustern 119
Abbildung 6-1: Exemplarische Relevanz und Informationsangebot bei KIP. Quelle: [Gerlach09] 124
Abbildung 6-2: Bildschirmabzug des Informationsportals für die Endmontage von Schaltschränken 126
Abbildung 6-3: Technische Architektur des Portalprototyps mit Implementierungsmustern 127
Abbildung 6-4: Wechselgarnituren einer Abfüllanlage. Quelle: [Kammermeier09, S. 251] 129
Abbildung 6-5: Bildschirmabzug des Montageinformationsportals. Quelle [Kammermeier09, S. 270] 130
Abbildung 6-6: Navigationsprobleme in dem Montageinformationsportal 130
Abbildung 8-1: Anwendungsbeispiel für einen Decorator. Quelle: [Gamma96, S. 200] 174
Abbildung 8-2: UML-Diagramm Decorator. Nach: [Gamma 96, S. 202] 174
Abbildung 8-3: Vergleich von Musterbeziehungen und Mustersequenzen. Nach: [Henney06, S. D1-25] 176
Abbildung 8-4: Beispielhafte Mustersequenzen und Implementierungsresultate 177

Verzeichnis der Tabellen

Tabelle 2-1:	Informationsraum für produktionsnahe Informationsbedarfe [Gerlach09, S. 166ff]	30
Tabelle 2-2:	Anforderungen an Konzepte zur Realisierung produktionsnaher Informationsportale	50
Tabelle 2-3:	Vergleich der Konzepte	52
Tabelle 4-1:	Ausschnitt Entwurfsmustersammlung	71
Tabelle 4-2:	Übersicht der Aufgabengliederung für die Analysemuster	73
Tabelle 4-3:	Übersicht der Aufgabengliederung für die Konzeptionsmuster	74
Tabelle 4-4:	Übersicht der Aufgabengliederung für die Implementierungsmuster	75
Tabelle 4-5:	Entwurfsmusterkatalog mit Gliederung, Musterstatus und Regellösungen	77
Tabelle 4-6:	Übersicht und Symbole der Musterbeziehungen	78
Tabelle 4-7:	Mustersammlung mit Kategorisierung und Kooperationsbeziehungen	82
Tabelle 4-8:	Aufgabensequenz „Portal-Engine erstellen“	86
Tabelle 4-9:	Aufgabensequenz „Portalseiten ergänzen“	87
Tabelle 5-1:	Auftragsinformationsbedarf in der kundenindividuellen Schaltschrankproduktion	95
Tabelle 5-2:	Beispielhafte kundenindividuelle Auftragsoptionen eines Schaltschranks.	96
Tabelle 5-3:	Mustersequenz für die Portalseite mit Montageinformationen	106
Tabelle 5-4:	Personalisierungsschema für Kerninformationen	115
Tabelle 5-5:	Mustersequenz zur Personalisierung der Portalseite „Montageoptionen“	120
Tabelle 6-1:	Charakteristik der untersuchten Unternehmen mit kundenindividueller Produktion	122
Tabelle 6-2:	Bewertung der Methode zur Realisierung produktionsnaher Informationsportale	132
Tabelle 8-1:	Entwurfsmustersprache für produktionsnahe Informationsportale	173

1 Einleitung

1.1 Ausgangssituation und Problemstellung

Kundendominierte Käufermärkte sind geprägt durch eine Heterogenisierung der Nachfrage und einem Streben nach Individualität im Hinblick auf begleitende Dienstleistungen, die Logistik und vor allem auf die Produktgestaltung [Spath09c, S. 11; Reichwald06, S. 22]. Diesem Bedarf folgend zielt eine kundenindividuelle Produktion (KIP) auf eine Wettbewerbsdifferenzierung im Hinblick auf schmale Kundensegmente [Strauß02, S.29]. Bei KIP können Kunden im Auftragsabwicklungsprozess Produktspezifikationen modifizieren in den Dimensionen Geometrie, Funktionalität und Design [Reichwald06, S. 201], für die anschließend individualisierte Produktexemplare erstellt werden. Die Erfüllung vielfältiger Kundenwünsche mittels KIP ist eine erfolgreiche Strategie zur nachhaltigen Sicherung der Wettbewerbsfähigkeit für Unternehmen an Hochlohnstandorten [Schirrmeister03]. Zunehmend orientieren sich die „Geschäftsprozesse im Informationszeitalter" am Kundenbedarf und Kundenprozess [Huber00, S. 29]. Zukünftig ist ein weiter steigender Anteil von KIP zu erwarten [Bullinger05a, S. 3576; Witte03, S. 1; Wiendahl04, S. 24].

Eine essenzielle Voraussetzung für KIP ist die informationstechnische Durchdringung und Vernetzung der Geschäftsprozesse in einer geschlossenen Informationskette zwischen allen Akteuren. [Piller06, S. 238]. In der Produktion sind kundenspezifische Arbeitsgänge ohne Informationen über die individualisierten Produktspezifikationen in aller Regel nicht ausführbar. Vor diesem Hintergrund erfährt der Produktionsfaktor Information eine deutliche Aufwertung [Schoder08, S. 11]. KIP misst Information die Relevanz eines elementaren Produktionsfaktors bei, dessen Präsenz unentbehrlich ist [Piller06, S. 21]. Zudem ist die Nutzungsintensität deutlich höher als in einer Serienproduktion, da Informationen für jedes individualisierte Produktexemplar vorzuhalten sind, und nicht nur je Produktionslos.

Auch bei manuellen Produktionstätigkeiten ist Information eine permanent und intensiv genutzte Ressource. An informationsintensiven Arbeitsplätzen sollte der Informationszugang generell aufgabenzentriert gestaltet sein [Bruse05, S. 120]. Demzufolge ist die produktionsnahe Informationsbereitstellung auf jede Aufgabe und jeden Arbeitsplatz abzustimmen. Ihre Gestaltung verlangt die gleiche Sorgfalt, die der Bereitstellung von materiellen Ressourcen zukommt. Dabei zeigen sich signifikante Unterschiede im Informationsbedarf der einzelnen Arbeitsplätze. Ebenfalls uneinheitlich sind die subjektiven Informationsbedarfe der Mitarbeiter. Infolge der ständigen Arbeitsplatzwechsel, die eine für KIP notwendige, flexible und „atmende" Produktion verlangt [Günthner06, S. 65], sind abweichende Vorkenntnisse und Informationsstände zwischen den Mitarbeitern unvermeidlich. Aufgabenzentrierte Informationsangebote sind daher arbeitsplatzspezifisch und darüber hinaus mitarbeiterspezifisch auszulegen und zu gestalten. Die in der Praxis realisierten produktionsnahen Informationsangebote sehen dagegen weiterhin die Verteilung von stets gleichartig aufgebauten, papierbasierten Medien und den Einsatz von Standardsoftware vor [Lang07, S. 30f]. Eine aufgabenzentrierte Informationsbereitstellung ist damit nicht zu realisieren. Alle Arbeitsplätze und alle Mitarbeiter werden mit identischen Informationen versorgt. Damit bleibt es den Mitarbeitern überlassen, aus dem generalisierten Informationsüberangebot die für ihre Aufgaben relevanten Informationen zu extrahieren. Mehraufwände für die Informationsbeschaffung sind regelmäßig die Folge. Der steigende Anteil an KIP verschärft die Problematik der mangelhaften, produktionsnahen Informationsversorgung.

1.2 Zielsetzung, Handlungsbedarf und Lösungsansatz

Ziel der vorliegenden Arbeit ist die effiziente und bedarfsgerechte Informationsversorgung von operativen Mitarbeitern im Realgüterprozess bei kundenindividueller Produktion. Erwartete Effekte sind reduzierte und möglichst geringe Aufwände für die Informationsbeschaffung sowie die Vermeidung von Fehlern bei der Informationsversorgung. Eine Erfolg versprechende Maßnahme besteht im Aufbau von produktionsnahen Informationsportalen, s. Abbildung 1-1.

Portale sind nutzerorientierte Medien, die Teams verbinden, dem Erfahrungsaustausch dienen sowie einen aktiven und zeitpunktgerechten Informationsabruf im „Pull" durch die Mitarbeiter bieten [Dias01, S. 275; Hoffmann01, S. 115]. In produktionsfernen Bereichen, z. B. im Einkauf und Personalwesen, haben sich Informationsportale bewährt. Sie vollziehen den erforderlichen Paradigmenwechsel von funktionszentrierten Anwendungen zu einer benutzer- und aufgabenzentrierten Informationsbereitstellung [Rösch00, S. 254]. In produktionsnahen Bereichen lassen sich mit Portalen die für KIP benötigten Informationen aus den vorhandenen operativen Informationssystemen extrahieren, aufgabenspezifisch filtern und aufbereiten. Derartige Informationsangebote sind für jeden Arbeitsplatz individuell zu implementieren. Dabei rechtfertigt die Relevanz in Verbindung mit der Nutzungsintensität der Information den Aufwand für die Realisierung von arbeitsplatz- und mitarbeiterspezifischen Informationsportalen.

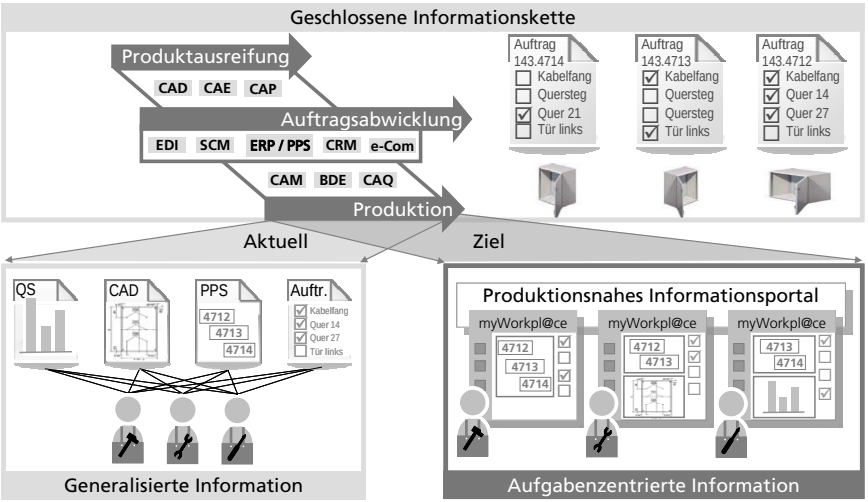


Abbildung 1-1: Aufgabenzentriertes Informationsangebot

Die Realisierung von Web- und Portalanwendungen ist eine stark interdisziplinär geprägte Tätigkeit, die Aufgaben der Programmierung, des Informationsdesigns und der Oberflächengestaltung einschließt [Mendes06, S. 9; Kappel04, S. 18; Pressman09, S. 37f]. Sie erfordert vertiefte Erfahrung und eine umfassende Kenntnis von Lösungskonzepten. Bekannte Verfahren bieten nur punktuelle Lösungskonzepte. Sie adressieren entweder rein technische Fragen der Implementierung von Portalen oder zeigen nur die Gestaltung von Informationsangeboten aus Sicht der Anwender und der Produktion. Handlungsbedarf

besteht hinsichtlich einer Methode, die sowohl die Konzeption einer aufgabenzentrierten Informationsbereitstellung als auch deren Umsetzung in produktionsnahen Informationsportalen unterstützt. Als Lösungsansatz wird in der vorliegenden Arbeit der Einsatz von Entwurfsmustern verfolgt. Entwurfsmuster zeigen bewährte Lösungen für softwaretechnische Entwicklungsaufgaben und sind für vielfältige Entwurfsprobleme bekannt. Davon ausgehend wird eine neue Entwurfsmustersprache für datenintensive Portalanwendungen im produktionsnahen Bereich entwickelt, die konzeptionelle und umsetzungsorientierte Lösungskonzepte verbindet und in einen systematischen Anwendungsbezug setzt.

1.3 Gegenstandsbereich und Aufbau der Arbeit

Gegenstandsbereich
Wie bereits konstatiert, ist der Aufwand für die Realisierung eines aufgabenzentrierten, produktionsnahen Informationsangebots nur bei einer hohen Relevanz und Nutzungsintensität der Informationen gerechtfertigt. Der Gegenstandsbereich der Arbeit ist daher auf die entsprechenden Formen einer KIP einzuschränken, die z. B. in [Spath09d, S. 9ff] ausführlich vorgestellt werden.

Unter *kundenindividueller Produktion* wird im Rahmen der vorliegenden Arbeit die diskrete, auftragssynchrone Serienproduktion von individualisierten Sachgütern in Einzellosen verstanden. Wesentlich dabei ist eine nach industriellen Prinzipien organisierte Serienproduktion. Die Produkte weisen einen nennenswerten Anteil von Standardkomponenten auf. Der Produktionsprozess sieht artteilige und logistisch entkoppelte, repetitive Tätigkeiten vor. Die Zykluszeit für einzelne Tätigkeiten liegt im Bereich mehrerer Minuten bis weniger Stunden. In der Konsequenz entsteht an den Arbeitsplätzen ein permanenter Bedarf an Informationen bezüglich der individualisierten Produktspezifikationen [Spath09a, S. 227, Gerlach09, S. 164]. Die kundenindividuelle Massenproduktion von Produkten für viele Einzelabnehmer fällt in diese Kategorie. Aber auch in kleineren Stückzahlen hergestellte, technisch anspruchsvolle Investitionsgüter liegen im Betrachtungsbereich der Arbeit.

Nicht betrachtet wird dagegen die reine *Einzel- und Unikatfertigung* mit entsprechend großen Arbeitsumfängen und langen Wiederholzyklen. Diese erreicht zwar eine hochgradige oder gar vollständige Produktindividualisierung, ist jedoch eher nach dem Verrichtungsprinzip organisiert und verwendet Projektsteuerungen [Reichwald06, S. 22]. Beispiele sind Unternehmen des Sondermaschinenbaus, die häufig Grundtypen anbieten und zu jeder technisch machbaren Produktvariante umkonstruieren. Der Informationsbedarf an den Arbeitsplätzen ist zwar hoch. Die Einmaligkeit der Tätigkeiten und somit der Informationsnutzung steht dem Aufwand für eine Aufbereitung des Informationsangebots entgegen. „Fertigen nach Zeichnung“ liegt somit nicht im Fokus der Arbeit. Ebenfalls nicht betrachtet wird die *Serienfertigung kundenspezifischer Produkte in großen Losen*, die zum Beispiel im Bereich der Automobilzulieferung verbreitet ist. Dabei werden Produkte, die individuell auf einen Kunden ausgerichtet und nur an diesen verkäuflich sind, in großen Stückzahlen gefertigt. Bezüglich des produktionsnahen Informationsbedarfs unterscheidet sich diese Fertigung jedoch nicht wesentlich von einer Massenfertigung. Entsprechend der großen Lose werden auftragsspezifische Informationen nur in längeren Zeitabständen benötigt. Selten anfallende Beschaffungsaufwände für Informationen sind dabei tolerabel.

Aufbau der Arbeit
Die Zielsetzung einer effizienten Informationsversorgung bei KIP erfordert eine einzelfallbezogene und aufwendige Erstellung produktionsnaher Informationsportale. Entscheidend für die erreichte Effizienz bei der Informationsversorgung ist die Qualität des realisierten Informationsangebots, die wiederum auf unvermeidlichen, konzeptionell anspruchsvollen Tätigkeiten des Informationsdesigns und der Programmierung beruht. Daher wird als Lösungsansatz eine neue Methode entwickelt, zur konzeptionellen Unterstützung der Realisierung von aufgabenzentrierten Informationsangeboten. Der im Folgenden beschriebene Aufbau der Arbeit spiegelt diese mittelbare Wirkungsbeziehung zwischen Zielsetzung und Lösungsansatz wider, s. Abbildung 1-2.

Kapitel 2 analysiert ausgehend von einer Beschreibung der Formen und Organisationsprinzipien einer kundenindividuellen Produktion, die hiervon induzierten Informationsbedarfe. Die Anforderungen an ein adäquat gestaltetes Informationsangebot, die Vorteilhaftigkeit von Informationsportalen für die Informationsbereitstellung und die Notwendigkeit einer konzeptionellen Unterstützung zu ihrer Realisierung werden begründet. Ein Vergleich mit dem Stand der Forschung belegt, dass derzeit keine geeignete unterstützende Methode existiert und ein Handlungsbedarf besteht.

Kapitel 3 stellt den Stand der Technik bezüglich Entwurfsmuster und bekannter Entwurfsmustersprachen für Informationsportale vor. Eine vergleichende Bewertung zeigt die Eignung des Lösungsansatzes und macht deutlich, dass für produktionsnahe Informationsportale noch keine Entwurfsmustersprache formuliert wurde.

In **Kapitel 4** wird eine neue Methode entwickelt, die sowohl alle Aspekte einer produktionskonformen Informationsversorgung als auch deren Umsetzung in Informationsportalen berücksichtigt. Die bekannten Entwurfsmuster werden aufgegriffen, selektiert, zu einer Entwurfsmustersammlung verdichtet und in einer Entwurfsmustersprache strukturell verknüpft. Den praktischen Einsatz der Entwurfsmustersprache fördert ein leicht anwendbares Werkzeug einer Musterbank.

In Kapitel 5 werden zwei, für KIP zentrale Informationsangebote konzipiert, die so gewählt sind, dass sie auf möglichst viele Einsatzfälle übertragbar sind, in dem hier betrachteten Gegenstandsbereich der diskreten Produktion von Sachgütern.

Das erste Informationsangebot betrifft die arbeitsplatzspezifische Bereitstellung von Montagereihenfolgen, auftragsspezifischen Stücklisten und individualisierten Produktspezifikationen am Beispiel einer realen Produktionsaufgabe. Das zweite Informationsangebot zeigt eine mitarbeiterspezifische Informationsbereitstellung anhand der personalisierbar ausgeprägten Darstellung der Montagereihenfolge. Beide Informationsangebote werden unter Einsatz der neuen Entwurfsmustersprache konzipiert und führen zu zwei sogenannten lösungsorientierten Mustersequenzen. Darunter sind die in konkreten Anwendungen eingesetzten Entwurfsmuster zu verstehen. Lösungsorientierte Mustersequenzen sind der Schlüssel zur Wiederverwendung von Lösungskonzepten und tragen zur konsistenten und erwartungskonformen Gestaltung von Informationsportalen bei.

In den abschließenden **Kapiteln 6 und 7** erfolgt die Evaluierung und kritische Würdigung des Ansatzes mit einem Ausblick auf weitergehende Forschungsfragen sowie die Zusammenfassung der Arbeit.

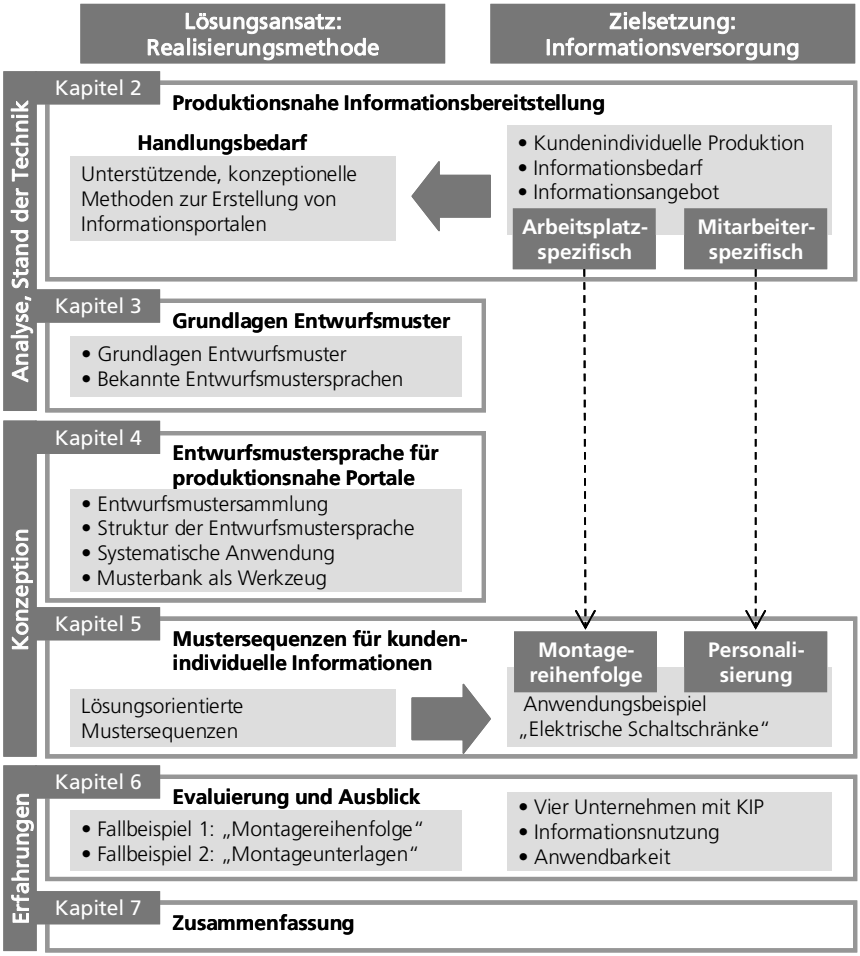


Abbildung 1-2: Aufbau der Arbeit

2 Produktionsnahe Informationsbereitstellung bei kundenindividueller Produktion

2.1 Formen und Organisation kundenindividueller Produktion

2.1.1 Individualisierte Produkte als Unternehmensstrategie

Gemäß der Alternativhypothese von Porter haben Unternehmen zur Wettbewerbsdifferenzierung die Wahl zwischen Kostenführerschaft oder Leistungsdifferenzierung. Die Ausschließlichkeit begründet sich aus den gegenläufigen Kostenfunktionen der Produktivität und der Flexibilität [Porter05, S. 40; Stotko03, S. 189; Piller06, S. 99]. Eine Massenproduktion gleichartiger Güter und die daraus resultierenden Skaleneffekte senken Produktionskosten und sichern ausreichende Erträge auch bei geringen Erlösen. Die alternative Strategie besteht in der Besetzung von Marktnischen mit hochwertigen, schwer zu imitierenden Produkten, für die anspruchsvolle Kundensegmente hohe Marktpreise akzeptieren. Einen Mittelweg verfolgen Hybridstrategien, die größere Marktsegmente mit variantenreichen oder individualisierten Leistungen zu moderaten Preisen bedienen [Bullinger05a, S. 3580]. Nach Pine entspricht Massenproduktion dem Versuch, Produkte herzustellen, die sich nahezu jeder leisten kann, wohingegen Hybridstrategien auf die Maxime setzen, dass nahezu jeder genau das Produkt erhalten kann, das er sich wünscht [Pine94, S. 78]. Die Erfolgspotenziale von Hybridstrategien beruhen auf einer effizienten, auftragssynchronen Fertigung ohne Endproduktlager, der dadurch beträchtlich reduzierten Kapitalbindung und einem geringen Verfallsrisiko [Pine94, S. 159f]. Zudem erhöhen sie die Kundenbindung, das Unternehmensimage, die Marktnähe und intensivieren den Kundenkontakt. Für die Verbraucher können sie gar in Preisvorteilen münden, falls die Endprodukte nur über die tatsächlich gewünschten Funktionen verfügen. [Blecker07, S. 5].

Organisationskonzepte für Hybridstrategien sind z. B. die *variantenreiche Serienfertigung*, *e-Business* und *Mass Customization*. Die Serienfertigung variantenreicher Sachgüter erfordert dezentrale, flexible und modulare Produktionsstrukturen, die eine klassische, funktionale Trennung der Verrichtungen überwinden und die Variantenbildung an das Ende der Prozesskette verlagern [Wiendahl04, S. 14]. Die variantenreiche Serienfertigung zielt somit auf die Flexibilisierung des Realgüterprozesses in der Auftragsabwicklung. E-Business betont dagegen die Kundenintegration und kann sowohl für Sachgüter als auch Dienstleistungen angewandt werden. Der Begriff des e-Business steht für eine Auftragsabwicklung, die eine Online-Kundenschnittstelle für das sogenannte *e-Commerce* paart mit einer durchgängigen, informationstechnischen Unterstützung der inner- und überbetrieblichen Leistungserbringung [Strauss02, S.15]. E-Business und die variantenreiche Serienfertigung finden zusammen unter dem Begriff *Mass Customization* bzw. *kundenindividuelle Massenproduktion* [Pine94, S. 78; Strauss02, S.138]. Kundenindividuelle Massenproduktion vereinigt die Kostenvorteile von Massenfertigung mit der Wettbewerbsdifferenzierung durch Produktindividualität [Bullinger05b], ist auf eher einfache Konsumgüter und große Absatzmärkte ausgerichtet und wie e-Business stark von der Kundenintegration geprägt [Strauss02, S.149; Wiendahl04, S. 6]. In der vorliegenden Arbeit wird eine Definition von KIP verwendet, die stärker ausgerichtet ist auf die Belange der Produktion, weniger auf die Markt- und Wettbewerbssicht. Wesentlich ist die auftragssynchrone Produktion in Einzellosen bei gleichzeitigem Seriencharakter. Wie bereits ausgeführt, erfüllt die kundenindividuelle Massenproduktion diese Definition, aber auch individualisierte, nicht notwendigerweiße für Massenmärkte gefertigte Investitionsgüter.

Die Individualisierung eines Leistungsangebots kann zu unterschiedlichen Zeitpunkten in der Lieferkette erfolgen, gegebenenfalls erst im Handel. Beispiele hierfür sind beim Händler ausgeschäumte Skischuhe [Strauss02, S. 152] oder die individuelle Auswahl und Kombination von standardisierten Einzelprodukten aus großen Produktpaletten im sogenannten *Bundle-to-Order*, s. Abbildung 2-1. Noch später erfolgt eine Selbstindividualisierung von anpassbaren Produkten direkt durch den Kunden [Piller06, S. 219ff]. Früher in der Wertschöpfungskette findet eine materielle Individualisierung von Produkten statt. Unter *Assemble-to-Order* ist die kundenspezifische Kommissionierung und Endmontage eines Produkts aus standardisierten Teilen zu verstehen. Konzepte des *Modify-to-Order* und *Engineer-to-Order* beziehen individuell gefertigte Teile oder gar die Konstruktion von neuen Teilen mit ein [Stotko03, S. 82ff].

Art der Kundenintegration	Integrations-punkt	Intensität der Kundenintegration
Match-to-Order / Bundle-to-Order Auswahl oder Bündelung aus Palette mit vorhandenen und standardisierten Produkten auf Basis der Kundenanforderungen	Point-of-Sale / Handel	Anzahl der Kundenspezifischen Wertschöpfungsaktivitäten Integrationsgrad des Kunden in die Wertschöpfungskette
Assemble-to-Order Herstellung kundenindividueller Produkte aus standardisierten Komponenten	Endmontage	
Modify-to-Order Fertigung kundenindividueller Produkte mit kundenbezogener Herstellung von Produktkomponenten	Fertigung	
Engineer-to-Order Kundenbezogene Entwicklung von Produkten („Customer Co-Design")	Produkt-entwicklung	

Abbildung 2-1: Zeitpunkte der Kundenintegration. Nach: [Stotko03, S. 85]

Hybride Strategien basieren auf Produktkonzepten, die eine möglichst hohe äußere Varianz eines Produkts mit einer möglichst geringen inneren Varianz erreichen. Die äußere Varianz bezieht sich auf die durch den Kunden wahrnehmbare Produktausführung und sollte breit genug sein, um alle Kundenwünsche zu bedienen. Die innere Varianz resultiert aus dem Produktaufbau und dem benötigten Teilespektrum zur Bildung der äußeren Varianz [Wiendahl04, S. 12; Piller06, S. 193]. Eine geringe innere Varianz besagt, dass die Produkte aus wenigen, möglichst standardisierten und kombinierbaren Komponenten aufgebaut sind. Typischerweise liegt der Schlüssel hierfür in der Modularisierung des Produkts und der Prozesse [Schenk02, S. 52; Piller06, S. 196; Pine94, S. 272ff]. Sie sorgen für die notwendige Flexibilisierung und Ökonomisierung der individualisierten Wertschöpfungskette, wie im nächsten Abschnitt beschrieben. Abbildung 2-2 zeigt unterschiedliche Freiheitsgrade der Modularisierung von Produkten.

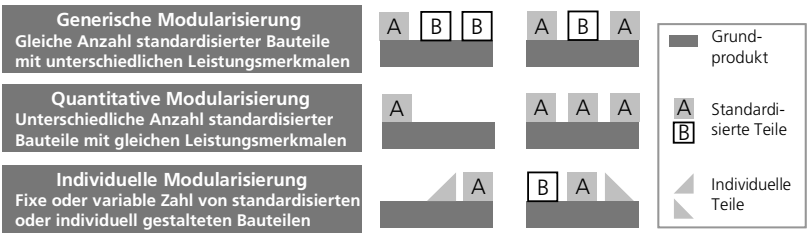


Abbildung 2-2: Modularisierungsprinzipien und Baukastensysteme. Nach: [Piller06, S. 229]

2.1.2 Kundenindividuelle Produktion im industriellen Umfeld

Im Produktionsablauf ermöglicht die Produktmodularisierung und die Verwendung eines Baukastens die Trennung der Wertschöpfungskette am sogenannten Kundenentkopplungspunkt [Syska06, S. 83]. Vor diesem Punkt werden Standardkomponenten auftragsanonym sowie in produktionstechnisch sinnvollen Losen vorgefertigt und zwischengepuffert. Nach dem Kundenentkopplungspunkt folgen die auftragssynchronen Fertigungsschritte zur kundenspezifischen Komplettierung und Individualisierung des Endprodukts [Wikner07, S. 146], s. Abbildung 2-3.

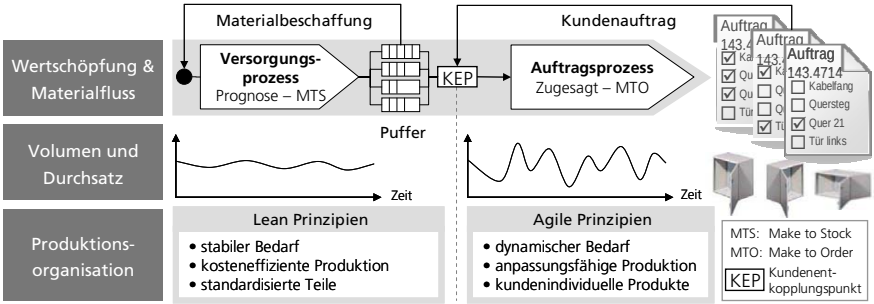


Abbildung 2-3: Organisation des Produktionsablaufs. Nach: [Wikner07, S. 146]

Kundenindividuelle Produktionsprozesse erfordern effiziente Organisationsstrukturen, da wirtschaftliche Abläufe und Materialflüsse bei kurzen Durchlaufzeiten erzielt werden müssen. Die losfreie Endstufe sollte flussorientiert und technologisch sowie kapazitiv flexibel sein. Zur Koordination der Vorproduktion sind Zugsteuerungen und kurze Wiederbeschaffungszyklen vorzuziehen. Geeignete Gestaltungsformen und Beispiele für flexible und modulare Produktionssysteme sind z. B. beschrieben in [Wiendahl04; Witte03; Feldmann04].

Der auftragssynchrone Teil der Wertschöpfungskette unterliegt bei KIP typischerweise schwankenden Kapazitätsanforderungen. Diesen sollte die Arbeitsorganisation begegnen und gestaltet sein nach den Prinzipien der horizontalen und vertikalen Funktionsintegration, der Selbstorganisation und der Flexibilisierung von Arbeitszeit und –inhalt [Piller06, S. 199; Strauss 02, S.199; Günthner06, S. 65]. Als vorteilhaft für die Erfüllung der Forderungen erweist sich die Organisation in selbstregulierenden Arbeitsgruppen, die üblicherweise etwa zehn Mitglieder umfassen. Charakteristisch hierfür sind Maßnahmen zur Arbeitserweiterung, Arbeitsbereicherung und des Arbeitswechsels [Antoni09, S. 216]. Eine Arbeitserweiterung erlaubt, in Abhängigkeit des Kapazitätsbedarfs Arbeitsumfänge dynamisch zwischen Arbeitsstationen aufzuteilen. Zugleich haben die Gruppenmitglieder mehrere Arbeitsstationen im Zuge des Arbeitswechsels zu beherrschen. Zur Arbeitsbereicherung werden dispositive, planende und kollaborative Aufgaben auf die Arbeitsgruppen übertragen. Dabei zeigt sich, dass die Delegation von dispositiven Aufgaben an die Werker die Flexibilität hinsichtlich individueller Kundenwünsche verbessert, die Produktqualität steigert und die Auftragsdurchlaufzeiten um bis zu 30 % verkürzt [Lay05, S. 2ff].

2.1.3 Produktionsfaktor Information für die kundenindividuelle Produktion

Eine KIP stellt entlang der gesamten Wertschöpfungskette hohe Anforderungen an die Auftragsabwicklung, die eine konsequente Unterstützung durch Informations- und Kommunikationssysteme erfordert [Piller06, S. 238; Strauss02, S. 155ff]. Als große Herausforderung erweist sich die notwendige, überbetriebliche Vernetzung aller beteiligten Akteure, vom Kunden über den Produzenten bis zum Lieferanten und zur Distribution. An erster Stelle der Kette stehen eine personalisierte Kundenansprache und präzise Kundenwunscherfassung als Aufgaben eines Customer Relationship Managements [Stotko03, S. 49]. Für eine weitere zuverlässige und rasche Umsetzung der Kundenaufträge sind die Koordination mehrstufiger Wertschöpfungsketten, ein Supply Chain Management der Lieferanten sowie die steuerungstechnische Integration und Flexibilisierung der Fertigungsanlagen unabdingbar [Bach05, S. 85; Piller06, S. 274; Stotko03, S. 197]. Die Bedeutung der Kunden- und Lieferantenintegration in die Auftragsabwicklung bei KIP sowie Lösungsansätze für eine informationstechnische Unterstützung und Integration werden in der Literatur ausführlich behandelt, vgl. hierzu [Reichwald03; Tseng05; Piller07; Dietrich07].

Aus Sicht der Produktion ist die Nutzung des Faktors Information für einen reibungslosen und effizienten Ablauf so wichtig wie die Integration der Auftragsabwicklung für die indirekten Bereiche. Im Produktionsfaktorenschema nach Gutenberg wird Information nicht explizit aufgeführt, aber üblicherweise als dispositiver Faktor angesehen, der dem effizienten Einsatz der Elementarfaktoren dient [Piller06, S. 20]. Bei KIP erfährt der – in allen Produktionszweigen unumstritten wichtige – Produktionsfaktor Information eine erhöhte Aufmerksamkeit [Schoder08, S. 11]. So können die Elementarfaktoren Informationsanteile enthalten, die essenziell für eine kundenindividuelle Produktausgestaltung sind, s. Abbildung 2-4. Beispielsweise ermöglichen intelligente, adaptierbare Bauteile und Werkstoffe ein Softcustomizing der fertigen Produkte. Betriebsmittel übernehmen kundenindividuelle Fertigungsdaten direkt aus den Auftragsinformationen über CAD/CAM-Schnittstellen. Werker an operativen Arbeitsplätzen mit variantenbildenden Tätigkeiten benötigen ebenfalls Individualisierungsinformationen. Ohne diese sind variantenbildende Tätigkeiten häufig nicht ausführbar. Sie sind in diesem Fall elementare Produktionsfaktoren, vergleichbar zu Werkstoffen [Piller06, S. 21].

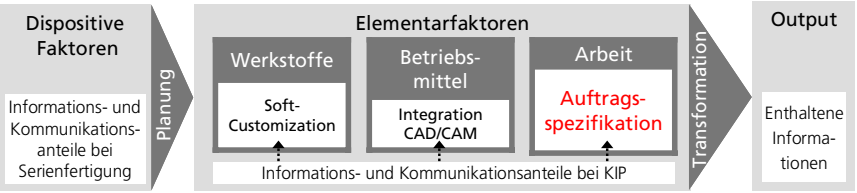


Abbildung 2-4: Produktionsfaktoren mit Informationsanteilen. Nach: [Piller06, S. 21]

Die Unverzichtbarkeit von Informationen für variantenbildende Arbeitsplätze zeigt sich deutlich bei modularen Produktbaukästen. Ein Beispiel hierfür sind pneumatische Versorgungs- und Wartungseinheiten. Sie folgen den Prinzipien einer generischen und quantitativen Modularisierung, und bestehen aus standardisierten Komponenten mit geometrisch identischen Verbindungsflanschen, aber jeweils verschiedenen Funktionen. Die Varianz des Endprodukts ergibt sich aus einer individuellen Kombination und Anordnung der Komponenten in wechselnder Anzahl, wie in Abbildung 2-5 dargestellt. Ohne die Kennt-

nis der vom Kunden individuell festgelegten Komponentenreihenfolge sind die Einheiten nicht korrekt montierbar.

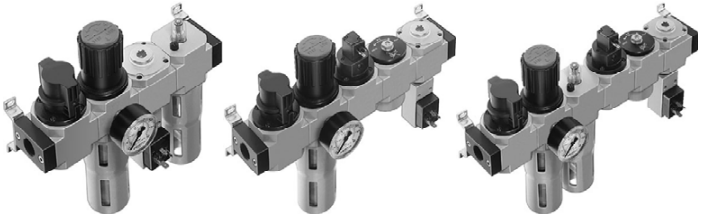


Abbildung 2-5: Beispiel einer generisch-quantitativen Modularisierung. Bild: [Festo AG]

Direkte Folgen von auftragsbezogenen Informationsdefiziten sind Verzögerungen der Fertigstellung, infolge von Warten auf die Informationen, sowie erhöhter Aufwand für die Informationsbeschaffung. Dabei besteht die Gefahr, dass Mitarbeiter diesen Aufwand scheuen und die Informationen nicht oder nur in veralteten Ständen nutzen [Lang07, S. 114]. Teilweise werden Tätigkeiten trotz eines Informationsdefizits ausgeführt, mit der regelmäßigen Konsequenz von Zusatzkosten durch Mehraufwände, z. B. für die Suche nach Arbeitsmitteln oder Materialien, wegen Nacharbeit und Qualitätseinbußen [Slama04, S. 131; Hacker08, S. 23].

Den stellenweise erheblichen Aufwand für die Informationsbeschaffung bei KIP belegt eine Untersuchung bei einem Maschinenbauunternehmen. Abbildung 2-6 zeigt das dieser Untersuchung entnommene Beispiel der Vormontage einer pneumatischen Wartungseinheit. Zur Vormontage benötigt werden ein fünfseitiger Fertigungsauftrag, sieben Seiten CAD-Zeichnungen und ein Umlagerbeleg. Die Sichtung der Unterlagen durch den Werker dauert 10 Minuten, obwohl nur ein geringer Teil der Informationen tatsächlich benötigt werden. Die eigentliche Montagezeit ist mit 12 Minuten veranschlagt. Somit fallen in dem Beispiel für die Informationsbeschaffung 45 % der gesamten Montagezeit an [Kammermeier09, S. 276].

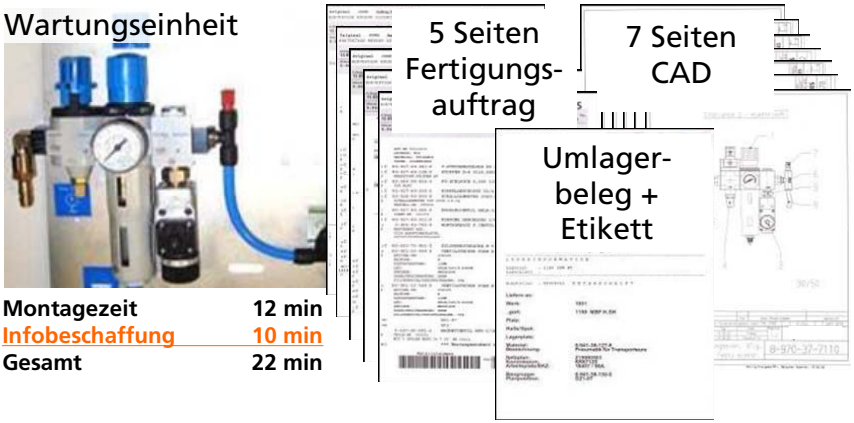


Abbildung 2-6: Beispielhafter Aufwand für Informationsbeschaffung. Quelle: [Kammermeier09, S. 268]

2.2 Informationsbedarfe produktionsnaher Teams

In unmittelbarer Konsequenz der informationstechnischen Durchdringung und Vernetzung kundenindividueller Geschäftsprozesse sowie der erforderlichen flexiblen Arbeitsorganisation ergibt sich ein drastisch geänderter und gestiegener Informationsbedarf seitens des Produktionsbereichs und der Arbeitsgruppen [Piller06, S. 104], der im folgenden Abschnitt näher ausgeführt wird.

Allgemein besteht eine **Information** im *„…Anteil einer Nachricht, der für den Empfänger neu ist. Durch Informationen werden beim Wirtschaftssubjekt bestehende Wahrscheinlichkeitsurteile bezüglich entscheidungsrelevanter Daten oder Ereignisse … verändert"* [Gabler04, S. 1473]. Informationen sind gemäß dieser Definition neu für den Nachfrager, Bekanntes stellt keine Information dar. Unter einem **Informationsbedarf** sind die *„Art, der Umfang und die Qualität aller Informationsressourcen, die zur zielgerichteten Bewältigung einer problematischen Situation (Durchführung einer bestimmten Aufgabe, Treffen einer Entscheidung) zu einer bestimmten Zeit und an einem bestimmten Ort benötigt werden"* [Kuhlen04, S. 56] zu verstehen. Informationsbedarfe lassen sich in den Dimensionen Informationsinhalt, Bedarfszeitpunkt, Bedarfskontext und Darstellungsform beschreiben [Schwarze98, S. 90f]. Für den Produktionsbereich charakteristische Informationsinhalte werden anschließend in einem Informationsraums dargelegt. Danach zeigt ein Intensitätsportfolio Zeitpunkt und Kontext der Informationsbedarfe. Die Beschreibung der Darstellungsform erfolgt als Anforderungskatalog an das Informationsangebot.

2.2.1 Informationsraum der kundenindividuellen Produktion

Der Informationsraum besteht aus einer Aufstellung der in der Produktion typischerweise nachgefragten Informationsinhalte und einer übergeordneten Gliederung. Grundlage ist eine Erhebung in der Literatur beschriebener Informationsbedarfe produktionsnaher Arbeitsgruppen, s. Tabelle 2-1. Eine Aufstellung der verwendeten Literatur ist in Anhang 9.1 gegeben. Für die Gliederung sind die ermittelten Informationsinhalte in zwei Schritten zu fünf Informationsclustern verdichtet [Gerlach09, S. 164]:

- Unternehmens- und Umfeldinformationen beschreiben die an der Wertschöpfungskette beteiligten Akteure. Weiter enthalten sind produktionsferne Informationen mit administrativen und organisatorischen Inhalten.
- Produktionsumfeldinformationen zeigen die Ausgestaltung und Ausstattung der Produktionseinrichtungen und das zu deren Betrieb benötigte Wissen.
- Auftragsabwicklungsinformationen betreffen die kundenindividuellen Produktspezifikationen und die erforderlichen Produktionsprozesse.
- Produktionsplanung und –steuerung sind Informationen über den zeitlichen und mengenmäßigen Einsatz der Produktionsressourcen im Rahmen der Auftragsabwicklung.
- Kennzahlen belegen das Zusammenwirken der Prozesse und Ressourcen.

Der Informationsraum zeigt die Vielfältigkeit und Vielgestaltigkeit des Informationsbedarfs. Die Oberbegriffe der Verdichtungsebenen vermitteln zunächst ein homogenes Bild der Informationsbedarfe, wie es in nahezu allen Unternehmen vorliegt. Dagegen zeigen die zugeordneten Informationsinhalte, wie unterschiedlich ausgestaltet die generischen Oberbegriffe tatsächlich sind. Weitere, noch nicht berücksichtigte Differenzierungsmöglichkeiten ergeben sich aus der Darstellungsform der einzelnen Informationsinhalte. In der

Praxis zerfasern so die Informationscluster in eine Vielzahl von Informationsinhalten mit unterschiedlichster Interpretation in den Unternehmen.

Infocluster	Informationsinhalt
Unternehmen und Umfeld	
Unternehmen & Organisation	Unternehmenspräsentation, Unternehmenskultur und Leitbild, Geschäftsbericht, Beteiligungen, Infotafel, Ankündigungen
Geschäftsstrategie	Vision und Ziele, Kundenbedürfnisse, Absatzplanung
Kunden	Kundenliste, Kontaktpersonen, Verkaufsbedingungen, Rahmenaufträge, Klassifikation, Teillieferungserlaubnis, Anlieferstellen, -zeiten, -modalitäten, Frachtinfo.
Lieferanten	Lieferantenliste, Kontaktpersonen, Kontakthistorie, Produktübersicht, Spezifikationen von Zukaufteilen, Vertragsbedingungen, Rahmenverträge, Konditionen
Mitarbeiter	Mitarbeiterverzeichnis, Telefonliste, Funktion, Bild, Kenntnisse, Biografie
Persönliches	Entlohnungsdaten, Individual- und Gruppenziele, Prämienstand, Personenstammdaten, Zugriffskontrolldaten & Passwörter, *(Anwesenheitszeiten, Schichtpläne s. PPS)*
Administration	Aufbauorganisation, Organigramm, Karrieremöglichkeiten, Stellenausschreibungen, Reisekostenabrechnung, Urlaubsantrag, Schulungsinformationen und –unterlagen
Produktionsumgebung	
Arbeitsumfeld	Arbeitsplatzgestaltung, Arbeitssicherheit
Ressourcen, Betriebsmittel	Betriebsmittelliste, Maschinendaten, Werkzeuginformationen, NC-Programme, Wartungspläne, Instandhaltung, alternative Ressourcen
Qualitätssicherung	Fehler- und Mängeldaten, Maschinenstörungsinformationen, Messwerte, Toleranzen, Prüfpläne, Übersichten für Prüf- und Freigabeteile, QM-Handbuch
Kontinuier. Verbesser. Prozess KVP	Verbesserungsvorschläge (Anzahl, Effekt), Themenspeicher, Maßnahmendefinition, -verfolgung, -wirksamkeit, Selbstbewertungen, Eigenaudits
Know-how	IT-gestützte Mitarbeiterkommunikation, Groupware, FAQ, ausgewählte Informationsquellen (Wikipedia), Prozessdokumentationen, Checklisten
Auftragsabwicklung	
Auftragsspezifikation	Lastenheft, Pflichtenheft, Termine, Produktkonfiguration, Auftragswiederholung
Auftragsänderung	Prüfung Auftragsstatus und Preisrelevanz, Verhandlung, Änderungsmanagement
Auftragsklärung	Machbarkeitsprüfung, Rücksprache bei fehlerhaften Auftragsspezifikationen
Preiskalkulation	Materialkosten, Produktionskosten, Preislisten
Produktunterlagen	Zeichnungen, Stücklisten, wachsende Stücklisten
Arbeitsplan	Arbeitsgänge, Arbeitsanweisungen, Vorgabezeiten, Rüstzeiten, Liegezeiten, Übergangszeiten, Werkzeuge, Spannmittel, Maße, Einstellwerte
Projektsteuerung	Projektpläne, Projektstatus, Projektkosten
Auftragsdokumentation	Produktabänderungen (ergänzte Auftragsstücklisten, Zeichnungen, Schaltpläne, Fotos, Skizzen, Texte), Ausführende, Messdaten, Prüfprotokoll
Auftragsverfolgung	Auftragsstatus, Liefertermine, Betriebsdatenerfassung (BDE: Fertigmeldung, Zeiten, Gutteile, Ausschuss)

Nachkalkulation	Materialverbrauch, Maschinenzeiten, Personalzeiten, BDE-Rückmeldungen
Produktionsplanung und –steuerung (PPS)	
Fertigungsaufträge	Auftragsbestand, Mengen, Losgrößen, Endtermine, Kommissionierlisten
Feinplanung	Auftragsfreigabe, Teileverfügbarkeit, Termine, Beistellteile
Ressourcenplan	Kapazitätsdaten, Ressourcenbelegung (Tag, Woche), Rüstzustand, Warteschlangen
Personaleinsatz-plan	Personalverfügbarkeit, An-, Abwesenheitszeiten, Urlaubspläne, Stundenkonten, Schichtpläne, Qualifikationsanforderungen, Qualifikationsstand Mitarbeiter
Materialdisposition	Materialbestand, -bedarf, -bestellung, Dispositionsart, Losgröße, Mindestbestand, Gebindeeinheit, Wiederbeschaffungszeit, Kanbanmenge, Lagerorte, Teileverwendung, Materialstammdaten (Stapelhöhe, Verfallsdatum, Gefahrgutklasse, Gewicht)
Kennzahlen	
Auftrags-abwicklung	Durchlaufzeit, Termintreue, Ausbringung, Kosten, Qualität, Fehlmengen, Reklamationen
Materialwirtschaft	Bestand, Reichweite, Fehlteilehäufigkeit, Materialumschlag, Einkaufsvolumen
Unternehmens-performance	Kundenzufriedenheit, Umsatzwachstum, Innovationsfähigkeit, Time to Market, Mitarbeiterzufriedenheit, internes Benchmarking, Projekte mit Kundenintegration
Ressourcen	Auslastung, Kapazitätsbedarf (Mitarbeiter), Qualitätsmanagement und KVP

Tabelle 2-1: Informationsraum für produktionsnahe Informationsbedarfe [Gerlach09, S. 166ff]

2.2.2 Intensität des Informationsbedarfs

Neben den Informationsinhalten ist die Nutzungsintensität ein maßgebliches Merkmal der Informationsbedarfe. Eine qualitative Charakterisierung der Nutzungsintensität ist möglich anhand eines Portfolios, mit den Achsen Nutzungszweck und Nutzungsanstoß. Der **Nutzungszweck** folgt der Arbeitsaufgabe. Zu unterscheiden sind Kernaufgaben von Komplementäraufgaben. *Kernaufgaben* bilden den Schwerpunkt der Aktivitäten des Informationsnachfragers, im Betrachtungsbereich sind dies Fertigungs- und Montageoperationen. Diese lassen sich weiter unterscheiden. Standardisierte Kernaufgaben sind variantenneutral und werden für jeden Auftrag identisch ausgeführt. Individuelle Kernaufgaben sind variantenbildend und auftragsspezifisch ausgeprägt. *Komplementäraufgaben* ergänzen und unterstützen die Kernaufgaben. Sie entstehen in aller Regel durch eine Arbeitserweiterung und Arbeitsanreicherung in Arbeitsgruppen und betreffen administrative oder operative Vorgänge. Die zweite Einflussgröße, der **Nutzungsanstoß** erfolgt aus einer Regeltätigkeit oder einer Bedarfstätigkeit heraus. *Regeltätigkeiten* sind planbar, vorhersehbar und permanent oder zumindest periodisch auszuführen. Im Gegensatz dazu sind *Bedarfstätigkeiten* ungeplant, unvorhersehbar und fallen sporadisch an. Sie sind nur selten oder gelegentlich auszuführen. Die Nutzungsintensität der Informationscluster ist am höchsten zu erwarten im Kontext von permanenten Regeltätigkeiten für individuelle Kernaufgaben. Im Umkehrschluss ist die erwartete Nutzungsintensität einer Information eher gering im Kontext von sporadischen Bedarfstätigkeiten für Komplementäraufgaben [Spath09b]. Abbildung 2-7 verdeutlicht die zu erwartende Nutzungsintensität der Informationscluster des Informationsraums. Die Einstufung der Informationscluster im Portfolio erfolgt im Hinblick auf direkt operative Arbeitsplätze bei kundenindividueller Produktion. Sie entspricht einer Situation, die bei vier untersuchten Unternehmen vorgefunden wurde und durchaus typisch ist für die kundenindividuelle Produktion, s. Kapitel 6.1.2.

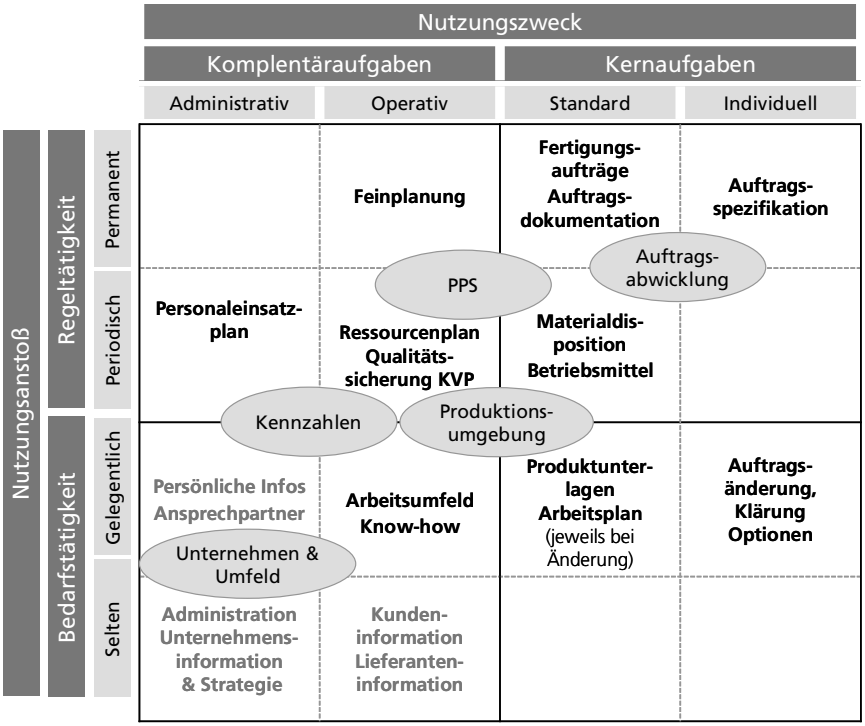


Abbildung 2-7: Portfolio der Intensität des Informationsbedarfs. Nach: [Spath09b]

Das Portfolio zeigt die große Bandbreite der erwarteten Nutzungsintensität der Informationscluster. Eine hohe Nutzungsintensität ist für Regeltätigkeiten gegeben. In wie weit sowohl Kern- als auch Komplementäraufgaben als Teil der Regeltätigkeiten an einem Arbeitsplatz wahrzunehmen sind, ist nur im konkreten Einzelfall zu beurteilen. Grundsätzlich ist eine Divergenz festzustellen hinsichtlich der Organisation gewerblicher, produktionsnaher Arbeit. So werden die Arbeitsinhalte in immer mehr Produktionsbereichen mit Komplementäraufgaben für dispositive, organisatorische und koordinierende Zwecke angereichert. Gleichzeitig ist für die Kernaufgaben durch die starke Verbreitung der Lean Prinzipien und der Produktionssysteme eine Tendenz zu standardisierten und stark repetitiven Tätigkeiten festzustellen, insbesondere in der Montage [Lacher06, S. 9]. Ein derart umfangreiches Aufgabenspektrum ist durchaus in vielen Unternehmen anzutreffen, insbesondere bei KIP. Jeder einzelne Mitarbeiter und Arbeitsplatz ist dabei durch divergente Anforderungen geprägt, die einerseits eine hohe Standardisierung und Produktivität der Kernaufgaben verlangen, andererseits komplexe und anspruchsvolle Komplementäraufgaben zur Ablaufoptimierung, zur Qualitätsverbesserung und zur Ressourcendisposition vorsehen. Als Folge daraus sind Arbeitsgruppen bei KIP häufig konfrontiert mit der gesamten Breite des Informationsraums als Informationsbedarf, bei gleichzeitig hohen Ansprüchen an eine effiziente, aufwandsarme Versorgung mit Kerninformationen.

2.3 Informationsangebot für produktionsnahe Teams

Die Informationsbedarfe in der kundenindividuellen Produktion unterscheiden sich signifikant von serienorientierten und kundenneutralen Produktionsprozessen, sowohl in den Inhalten, vor allem aber in der Nutzungsintensität und der Relevanz für die Produktionsschritte. Im Folgenden wird untersucht, wie ein adäquates Informationsangebot gestaltet sein sollte, welche Realisierungsmöglichkeiten bestehen und wie verbreitet diese in der Praxis sind.

2.3.1 Anforderungen an die Informationsbereitstellung

2.3.1.1 Arbeitsplatzspezifische Kerninformationen

In Arbeitsgruppen steht ein vielfältiger Informationsbedarf an, mit jedem zu produzierenden Teil oder Auftrag. Informationen sollten daher unmittelbar am Arbeitsplatz im Zugriff sein, wobei aus Effizienzgründen nur eine notwendige, minimale Informationsdichte anzustreben ist [Kuba97, S. 56; Slama04, S. 134]. Der Kontext der kundenindividuellen Produktion erlaubt ein differenziertes Bild in Bezug auf einzelne Arbeitsstationen. Aus dem Betrachtungsfokus von industriell organisierten, arbeitsteiligen Produktionsstrukturen folgt, dass die Arbeitsumfänge an den einzelnen Arbeitsplätzen begrenzt sind. Weiterhin ist von einer Modularisierung und Standardisierung des Produktaufbaus in einem Baukasten auszugehen. Die Arbeitsumfänge weisen somit variantenneutrale und variantenbildende Anteile auf. Dementsprechend selektiv sind die Informationsbedarfe. Die variantenneutralen Arbeitsinhalte sind für jeden Auftrag auszuführen und hinreichend bekannt. Diesbezügliche Informationen sind den Werkern nicht neu und somit nur von geringem Nachrichtenwert. An Arbeitsstationen werden also nur Informationsinhalte permanent im Zugriff benötigt, die unmittelbar variantenbildende Arbeitsinhalte betreffen. Diese variantendefinierenden Informationsinhalte sind die essenziellen Anteile des elementaren Produktionsfaktors Information. Zur Betonung ihrer hohen Relevanz für die Kernaufgaben werden sie in der vorliegenden Arbeit auch als *Kerninformationen* bezeichnet.

Die Gestaltung der Bereitstellung von Kerninformationen verdient die gleiche Aufmerksamkeit, die allen Elementarfaktoren zukommt, und von Lotter für materielle Ressourcen folgendermaßen beschrieben wird: *„Wirtschaftliche Montagekonzepte zeichnen sich u. a. dadurch aus, dass sie während der Montage bei sämtlichen Handhabungs- und Fügevorgängen jede unnötige Bewegung des Montageobjekts, des Menschen und der Montagemittel vermeiden"* [Lotter06, S. 59]. In diesem Sinn vermeidet eine wirtschaftliche Bereitstellung von Kerninformationen alle unnötigen Bewegungen und Aktivitäten seitens der Mitarbeiter. Die wirtschaftliche Bereitstellung materieller Elementarfaktoren folgt den Prinzipien des Lean Managements, die Verschwendung mit der sogenannten 5S-Methode eliminieren [Spath03a; Takeda04, S. 19]. Die Bezeichnung 5S steht für die fünf Schritte Sortieren, Systematisieren, Säubern, Standardisieren und Selbstdisziplin. Alle zur Erfüllung einer Aufgabe nicht benötigten Ressourcen stören den Arbeitsablauf und werden vom Arbeitsplatz entfernt. Nur die notwendigen Einrichtungen und Materialien verbleiben griffbereit und entsprechend dem Arbeitsablauf übersichtlich angeordnet [Sekine95, S. 29]. Die Bereitstellung von Kerninformationen sollte ebenfalls nach den Prinzipien der 5S-Methode erfolgen. Das geforderte „5S-Informationsangebot" ist dazu auf die relevanten Informationsinhalte zu reduzieren, übersichtlich zu präsentieren, mit minimalen Interaktionen zu erreichen und wie die Anordnung der Ressourcen ebenfalls arbeitsplatzspezifisch zu gestalten. Ein derartiges Informationsangebot für Arbeitsplätze bei kundenindividueller

Produktion wird im Weiteren als *KIP-Infoset* bezeichnet. Es ist geprägt durch [Spath09a, S. 230]:

- Inhaltsfilter mit einem Kontext in den Dimensionen Arbeitsstation und Auftrag;
- produktionskonforme Interaktionen, z. B. mit Touchscreens, Barcodeleser, Fußschalter, Taster;
- Visualisierungen und Darstellungen, z. B. Layouts von Arbeitsplätzen und Materialanordnungen, vgl. dazu [Sternemann00, S. 122];
- partielle Darstellungen von Auftragsoptionen und hierfür notwendigen Tätigkeiten.

2.3.1.2 Flankierende Komplementärinformationen

Die Informationsanalyse zeigt weitergehende Informationsbedarfe, die aus produktionsnahen Komplementäraufgaben resultieren oder kundenneutrale Arbeitsumfänge und Teile betreffen und im Kontext der aktuellen Aufträge oder Arbeitsumgebungen anfallen. Beispiele für Informationsbedarfe im Auftragskontext sind Montageanweisungen, Prüfpläne und vollständige Stücklisten. Im Kontext der Arbeitsumgebung zu sehen sind Wartungsvorschriften oder Werkzeugdaten. Der gemeinsame Kontext legt es nahe, arbeitsplatzspezifische KIP-Infosets mit weiterführenden Komplementärinformationen gezielt anzureichern und in einem einheitlichen System zu präsentieren. Bei Bedarf sind die Komplementärinformationen aus dem Informationsangebot für die Kernaufgaben in nur einem Navigationsschritt effizient zu erreichen. Eine derartige, „flankierende" Bereitstellung von Komplementärinformationen vermeidet die Navigation in mehreren Informationssystemen und erleichtert den Werkern die Informationsbeschaffung [Spath03c; Spath05b]. Ein Informationsangebot mit arbeitsplatzspezifischen Kern- und flankierenden Komplementärinformationen besteht aus [Gerlach09, S. 176]:

- einer Einstiegsseite mit einem zentralen Navigationsknoten;
- der Bereitstellung aller für die Auftragsausführung notwendigen, essenziellen und variantendefinierenden Kerninformationen und Erfassungsfunktionen;
- einfachen Möglichkeiten für den Abruf bedarfsweise benötigter Informationen.

2.3.1.3 Sporadische Notifikationsfunktionen

Trotz allen Anstrengungen zur Reduzierung der inneren Varianz ist eine Zunahme des Spektrums der Teile und der Tätigkeiten bei KIP in der Regel nicht zu vermeiden. Zwangsläufig ergibt sich eine geringe Wiederholrate für seltene Produktoptionen und Tätigkeiten. Zwischen der Definition einer Produktoption in der Entwicklung und ihrer ersten Ausführung in der Produktion kann eine längere Zeit vergehen. Auch ist zwischen den weiteren Ausführungen mit Änderungen der Informationsinhalte zu rechnen. Eine – heute übliche - Verteilung im Push-Verfahren von neuen oder geänderten Informationsinhalten zum Zeitpunkt ihrer Entstehung kann zu beträchtlichen Latenzzeiten führen, zwischen dem Erhalt einer Information und ihrer Verwendung. Dagegen ermöglicht eine intranetbasierte Informationsbereitstellung den aktiven und zeitpunktgerechten Informationsabruf im „Pull-Verfahren" [Hoffmann01, S. 115]. Allerdings wird nicht jeder Informationsbedarf tatsächlich erkannt. Beispiele sind vom Werker unbemerkte Teileänderungen oder aktualisierte Arbeitsanweisungen. Daher bedarf es Funktionen zur Erregung und Lenkung der Aufmerksamkeit der Werker auf vorhandene, aktuelle Informationen. Die Aufmerksamkeitserregung sollte ebenfalls flankierend erfolgen. Die Kerninformationen sind mit kontextsensitiven Notifikationsfunktionen zu versehen, die Werkern die Existenz neuer

bzw. aktualisierter Informationsangebote aktiv anzeigen und so sporadische Informationsbedarfe stimulieren. Eine Push-Notifikation ergänzt die Pull-Informationsabrufe und stellt die Aktualität der Informationsstände der Werker sicher.

2.3.1.4 Bidirektionale Kommunikation

In Folge der Arbeitsanreicherung sind die Mitarbeiter in der Produktion Träger wichtigen Know-hows, das es zu erfassen und vorzuhalten gilt, insbesondere in variantenreichen und individualisierten Produktionen [Westkämper09, S. 190; Strauss02, S.200]. Sie optimieren die Arbeitsabläufe in kontinuierlichen Verbesserungsprozessen, treffen dispositive Entscheidungen oder sind in Qualitätszirkeln aktiv. Der Austausch des entstehenden Wissens und die Erfassung von produktionsrelevanten Ereignissen erfordern eine intensive und vor allem bidirektionale Kommunikation, die alle Mitarbeiter initiieren können [Kuba97, S. 3 & 44]. Zur informationstechnischen Unterstützung eignen sich Intranetanwendungen und Portale, die mit ihren strukturierten und schnellen Übermittlungsmöglichkeiten eine aufgabenbezogene, horizontale und asynchrone Kommunikation ermöglichen [Hoffmann01, S. 266]. Wie bei Komplementärinformationen ist die flankierende Bereitstellung der Kommunikationsfunktionen anzustreben. Sie ermöglichen:

- das Erfassen und Editieren von Texten;
- die Anreicherung mit multimedialen Inhalten, Spracheingaben und Bildarchiven;
- eine Gliederung entsprechend den Elementen und Objekten im Shopfloor.

2.3.1.5 Personalisierte Informationsangebote

Die Vorteilhaftigkeit der Arbeitsorganisation bei KIP in selbstregulierenden Arbeitsgruppen mit einem ständigen Arbeitsplatzwechsel wurde bereits ausgeführt, s. Kapitel 2.1.2. Beides resultiert in variierenden Vorkenntnissen, Fertigkeiten und Routinegrade der Mitarbeiter an den Arbeitsplätzen. Für die Informationsbereitstellung sind daher differierende Informationsbedarfe zu erwarten, denen ein einheitliches Informationsangebot für alle Mitarbeiter nicht gerecht wird. Vielmehr bedarf es einer personalisierten Informationsbereitstellung [Huang03, S. 43]. Personalisierung ist die Anpassung des Informationsangebots an die individuellen Fähigkeiten, Bedürfnisse und Aufgaben eines Benutzers, mit dem Ziel der Steigerung der Produktivität und Effizienz bei der Problemlösung und Aufgabenerfüllung [Bruse05, S. 79]. Für die Produktion sind hauptsächlich die angebotenen Informationsinhalte an den Informationsständen der Mitarbeiter auszurichten. Bekannte Informationen sind von geringem Wert, sie stören in einem auf Wesentliches reduzierten KIP-Infoset. Von einem personalisierten Informationsangebot ist zu erwarten, dass:

- Informationsinhalte zu Informationsstand und Routinegrad der Mitarbeiter passen;
- bekannte Informationen nur untergeordnet oder auf Verlangen präsentiert werden;
- sporadische Notifikationen persönliche Informationsstände von Mitarbeitern berücksichtigen und nach Kenntnisnahme der neuen Informationsinhalte unterbleiben.

2.3.1.6 Gewichtung der Anforderungen im Informationsraum

Die genannten Anforderungen an ein Informationsangebot gelten nicht für alle Informationsinhalte gleichermaßen. Abbildung 2-8 ordnet die aufgestellten Anforderungen den Informationsclustern zu. Ein arbeitsplatzspezifisches Informationsangebot zu realisieren ist nur für permanent benötigte Kerninformationen lohnend. Alle weiteren, Regeltätigkeiten

betreffende Informationen sind flankierend und kontextgerecht bereitzustellen. Kommunikations- und Notifikationsfunktionen bieten sich vorwiegend für gelegentliche oder komplementäre Informationsbedarfe an. Die Personalisierung von Informationsangeboten ist ein softwaretechnisch und administrativ aufwendiges Unterfangen und sollte aus Effizienzgründen nur in einem sinnvollen und tatsächlich notwendigen Umfang erfolgen [Bruse05, S. 162]. Aus diesem Grund ist eine Personalisierung nur für intensiv genutzte Informationen für Kernaufgaben angebracht.

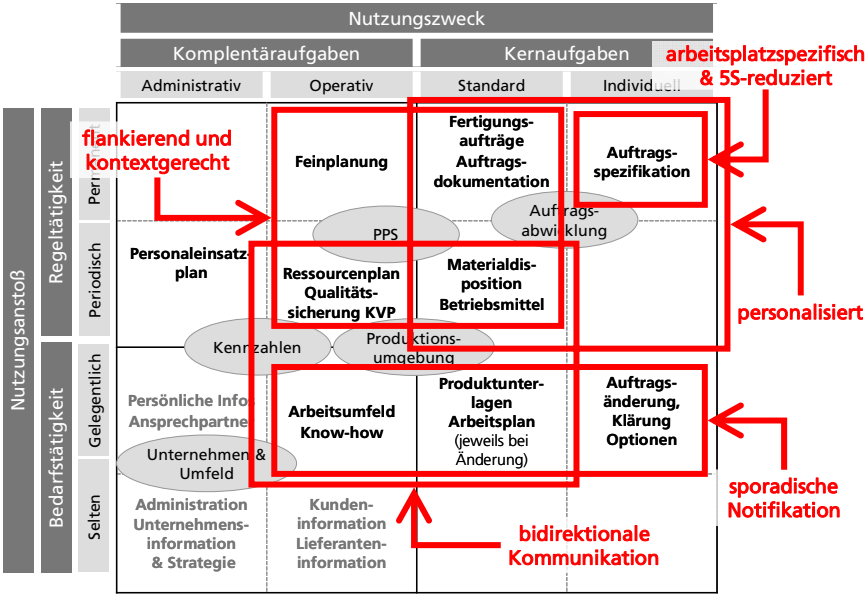


Abbildung 2-8: Anforderungen an das Informationsangebot im Informationsraum

Die Anforderungsbereiche sind überlappend und nicht exakt, sondern nur tendenziell und im Hinblick auf den produktionsnahen Einsatz zu sehen. Die höchste Relevanz und die höchsten Anforderungen bestehen für die Bereitstellung von Kerninformationen bezüglich der Auftragsabwicklung. Diese sind mit den Informationsclustern aus den angrenzenden Intensitätsbereichen eng zu verzahnen. In der weiteren Arbeit werden somit Informationen der Auftragsabwicklung, der PPS, der Produktionsumgebung und zu Kennzahlen betrachtet. Zusammenfassend ist von einem effizienten und bedarfsgerechten Informationsangebot bei KIP zu erwarten, dass:

- Kerninformationen arbeitsplatzspezifisch selektiert, auf die notwendigen Inhalte reduziert und mit einer interaktionsarmen Navigation präsentiert werden;
- Komplementärinformationen flankierend und kontextgerecht angeboten werden;
- sporadische Notifikationsfunktionen auf neue Informationsstände hinweisen;
- bidirektionale Kommunikationsfunktionen für Werker initiierbar und nutzbar sind;
- Personalisierungsmöglichkeiten bestehen für mitarbeiterspezifische Informationsangebote entsprechend den individuellen Vorkenntnissen, Fähigkeiten und Informationsstände der Mitarbeiter.

2.3.2 Realisierungsformen und Technologien

Informationsangebote der geforderten Form sind technisch darstellbar. Nachfolgend werden die Voraussetzungen, Technologien und Realisierungsmöglichkeiten vorgestellt.

Ein KIP-Infoset ist Teil einer geschlossenen Informationskette zur Abwicklung kundenindividueller Aufträge und in die Informationsarchitektur eines Unternehmens zu integrieren. Informationsarchitekturen beschreiben die zur Abbildung der Unternehmensprozesse eingesetzten Softwareanwendungen, deren Beziehungen und die technischen Infrastrukturen mit Hardwarekomponenten [Scheer09, S. 550; Mertens04, S. 20; Huber00, S. 17]. Abbildung 2-9 zeigt die typischerweise eingesetzten betrieblichen Standardanwendungen, deren Informationen in ein KIP-Infoset einfließen können. Entsprechend den Aufgaben und Funktionsbereichen kommen im Zuge der Produktausreifung entwicklungsorientierte Systeme zum Einsatz. In der Produktion finden sich Steuerungs- und Kontrollanwendungen. Die produktionsnahen Enterprise Resource Planning Systeme (ERP) für die Auftragsabwicklung und Produktionsplanung verbinden beide Sichten und Datenwelten.

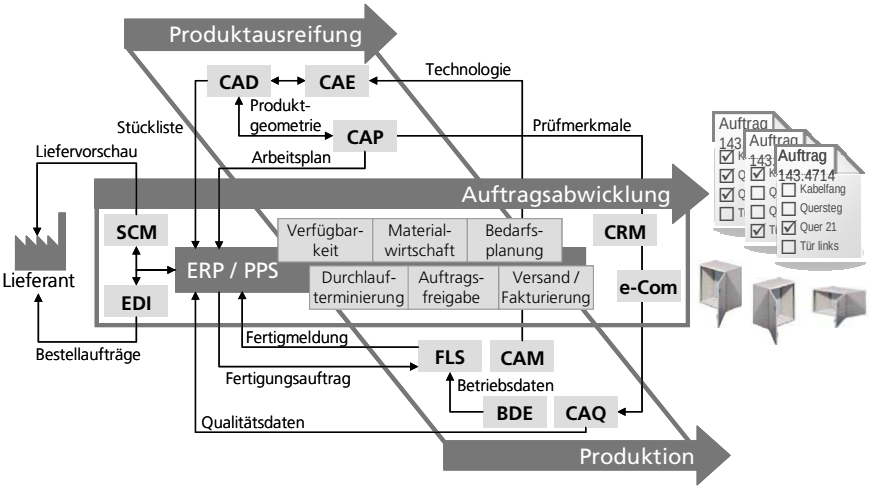


Abbildung 2-9: Informationsarchitektur der Auftragsabwicklung. Nach: [Mertens04, S. 267]

Informationsarchitekturen unterscheiden sich entweder im Einsatz vieler kleiner, kooperierender Komponenten oder durch den Aufbau großer, monolithischer Anwendungen [Mertens04, S. 20]. Dabei hat sich eine Integration über alle Funktionsbereiche hinweg als nicht praktikabel und instabil erwiesen [Mertens04, S. 7]. Neben den ERP-Systemen sind daher weitere Applikationen im Einsatz, z. B. CRM, SCM und e-Commerce, durchaus von unterschiedlichen Herstellern [Huber00, S. 39]. Infolgedessen ist eine effiziente Abwicklung kundenindividueller Aufträge auf mehrere kooperierende Systeme verteilt. Hierbei kommen zur Vernetzung häufig explizite, sogenannte EAI-Produkte zum Einsatz [Schelp02, S.8]. In der Praxis erweisen sich Informationsarchitekturen als heterogene Systemlandschaften, die unterschiedliche Technologien, Individual- und Standardsoftware sowie schwer zu ersetzende Altsysteme, sogenannte *Legacy-Software,* vereinigen [Eriksson00, S. 371].

Die Heterogenität und Vielfalt der Systeme äußert sich auch in ihrer inhomogenen, komplexen und aufwendigen Bedienung, die die Informationsbeschaffung seitens der Mitarbeiter in der Produktion erschwert. Abhilfe versprechen Informationsportale. Portale sind zentrale, webbasierte und personalisierbare Einstiegspunkte in vielschichtige Informationsangebote [Aneja00, S. 2; Bach05, S. 38; Mertens04, S. 4], die üblicherweise heterogenen und verteilten Informationsquellen entstammen [Davydov01, S. 127; Chan02, S. 115]. Grundsätzlich lassen sich Webportale von Unternehmensportalen unterscheiden. Webportale richten sich an ein anonymes und massenhaftes Publikum aus dem Internet [Dias01, S. 276]. Unternehmensportale bieten eine prozess- und aufgabenorientierte Informationsaufbereitung mit einem rollenspezifischen Zugriff für autorisierte Mitarbeiter oder Geschäftspartner von Unternehmen [Röhricht01, S.163]. Dies setzt den Datenaustausch mit den vorhandenen Systemen und eine Integration der Portale in die Informationsarchitekturen voraus [Spath05a, S. 11; Mertens04, S. 4]. Neben der Prozessintegration ist die Personalisierung der Informationsangebote ein charakteristisches Merkmal von Portalen [Bach05, S. 40]. Unternehmensportale lassen sich nach den Zielgruppen unterschieden in Mitarbeiter- (B2E), Endkunden- (B2C), Geschäftskunden- und Lieferantenportale (B2B) [Gurzki03, S. 10; Stelzer04, S. 23; Chan02, S.116]. Sie werden vornehmlich eingesetzt zum Zweck der Entscheidungsunterstützung, Kollaboration, Informationspublikation und der Prozessausführung [Großmann05, S. 34ff; Dias01, S. 274f; Huang03], siehe Abbildung 2-10. Tendenziell wachsen Portale zu umfassenden Anwendungen zusammen, die mehrere Zielgruppen und Funktionalitäten abdecken [Gurzki03, S. 10].

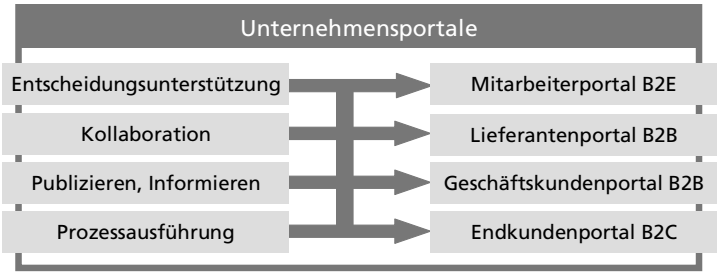


Abbildung 2-10: Anwendungsbereiche von Unternehmensportalen

2.3.3 Praxis der produktionsnahen Informationsbereitstellung

Zum Abschluss der Informationsanalyse wird den Anforderungen an die Gestaltung der Informationsangebote die Praxis der produktionsnahen Informationsbereitstellung gegenübergestellt. Die Nutzung produktionsbegleitender Informationen war mehrfach Untersuchungsgegenstand von Studien. Spath und Wilhelm untersuchten 2003 in einer schriftlichen Befragung von 159 Unternehmen die Nutzung von Informations- und Kommunikationsmedien durch gewerbliche Mitarbeiter in der Produktion. Betrachtete Informationsinhalte waren Arbeitspläne, Prüfpläne, Umfeldinformationen und Ad-hoc-Inhalte, z. B. für Störungsmeldungen oder Qualitätsmängel [Spath03b, S. 5f]. Die Befragung zeigt, dass Produktionsinformationen überwiegend papierbasiert sind und im „Push-Prinzip“ verteilt werden. Auch Ad-hoc-Informationen werden nach wie vor persönlich oder telefonisch weitergeleitet, s. Abbildung 2-11 oben. Eine produktionsnahe Onlinebereitstellung von Auftragsinformationen erfolgt vorwiegend in den Systemen, mit denen die Informationen erstellt wurden [Spath03b, S. 33].

Eine von Lang 2007 publizierte, ähnliche Erhebung bei 51 produzierenden Unternehmen bestätigt dieses Ergebnis. Abgefragt wurde die Nutzung von Umfeldinformationen und von Arbeitsanweisungen. Lang stellt ebenfalls fest, dass Papier bei den Informationsmedien mit Abstand dominiert. Zur Informationsdistribution wird das Holprinzip nur selten eingesetzt. Meistens erfolgt eine Informationsverteilung, öfter auch in Kombination mit einer Verpflichtung zum Quittieren der erhaltenen Information durch die Werker, s. Abbildung 2-11 unten. Analoge Aussagen finden sich in weiteren Untersuchungen. Slama konstatiert 2004, dass an den Arbeitsplätzen verfügbare Auftragsbegleitkarten und Montageanleitungen überwiegend papiergebunden sind [Slama04, S. 133]. Kuba findet bereits 1997 bei einer Untersuchung von benutzten Informationsmedien in der Endmontage in 59 % der Fälle papierbasierte Informationsträger, in 23 % mündliche Kommunikation und nur in 18 % informationstechnische Medien [Kuba97, S. 36]. Auffallend ist der über die Jahre konstant hohe Anteil an papierbasierter Informationsverteilung, trotz einer mittlerweile tiefen Durchdringung der Produktion mit einer informationstechnischen Infrastruktur sowie Bildschirmen an den Arbeitsplätzen.

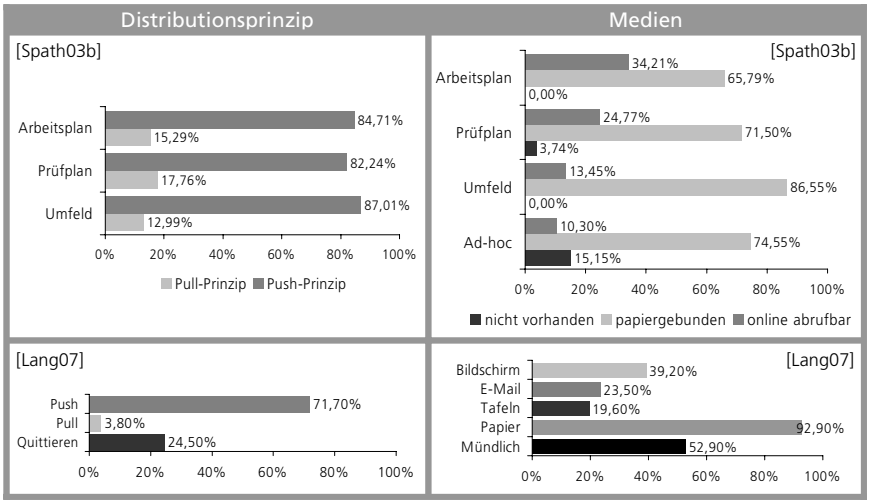


Abbildung 2-11: Distribution und Medien. Quellen: [Spath03b, S. 31ff; Lang07, S. 30f].

Intranetanwendungen nutzen Unternehmen in der Produktion vorwiegend zur Verbreitung von aktuellen Mitteilungen und Unternehmensnachrichten sowie die Verteilung organisatorischer Informationen [Hoffmann01, S. 266 & S. 197]. Weitere Anwendungen bestehen als Lernmedien zur Mitarbeiterqualifizierung und den Know-how-Transfer [Glander06, S. 187f]. Mitarbeiterportale werden häufig verstanden und eingesetzt im Sinne eines *Employee-Self-Services*, der administrative und personalwirtschaftliche Funktionen vorhält [Schäffer-Külz05, S. 21; Bach05, S. 76]. Anwendungsmöglichkeiten für Portale in der Produktion werden von Huang und Mak aufgezeigt zur Entscheidungsunterstützung für die Disposition und Arbeitsvorbereitung [Huang03, S. 7f]. Einsatzmöglichkeiten zur Verteilung operativer Informationen zeigen sie nicht auf.

Die Untersuchungen zeigen, dass Intranet- und Portalanwendungen als produktionsbegleitende Systeme genutzt werden, mit einem zusätzlichen Angebot an digitalisierten In-

formationen, die zuvor papierbasierte Dokumente ersetzten. Eine Integration mit den produktionsnahen, betrieblichen Informationen und Daten findet selten statt. Bezüge zwischen den koexistierenden Informationsbeständen sind manuell herzustellen, zum Beispiel durch Ablage einer Wartungsanleitung im Content Management System (CMS) des Intranets unter der vom ERP-System vergebenen Materialnummer des Betriebsmittels.

2.3.4 Schlussfolgerungen aus der Informationsanalyse

Die Informationsanalyse der produktionsnahen Informationsbereitstellung bei kundenindividueller Auftragsabwicklung lieferte folgende wesentliche Erkenntnisse:

- Informationen über die Auftrags- und Produktspezifikationen werden zu elementaren Produktionsfaktoren für variantenbildende Produktionsschritte.
- Im Detail betrachtet erweisen sich die Informationsinhalte heterogen ausgeprägt und die Informationsbedarfe selektiv seitens der Mitarbeiter und Arbeitsplätze.
- Die Informationsrelevanz in Verbindung mit der Nutzungsintensität rechtfertigt den Aufbau von KIP-Infosets mit arbeitsplatz- und mitarbeiterspezifischen Informationsangeboten.
- KIP-Infosets sind wie alle elementaren Produktionsfaktoren sorgfältig auszuwählen, auf die Arbeitsplätze abzustimmen und für die Mitarbeiter aufzubereiten.
- Weiter sind sie in die Abwicklung der Geschäftsprozesse eingebunden und bedienen sich einer Vielzahl von Informationen aus verschiedenen Systemen.
- In der Praxis werden nur selten adäquate Informationsmedien eingesetzt.

KIP-Infosets decken, im Sinne des 5S-Prinzips, exakt die individuellen Informationsbedarfe eines Mitarbeiters an einer Arbeitsstation - und zwar nur diese. Im Unterschied dazu warten Standardanwendungen mit standardisierten Informationsangeboten auf, sind unspezifisch, universell einsetzbar und mit generalisierten Informationen überfrachtet. Folglich genügen vordefinierte und standardisierte Informationsangebote aus Standardanwendungen nicht den Ansprüchen an ein KIP-Infoset, das für jeden Einsatzfall individuell auszuführen ist.

Prinzipielle Möglichkeiten zur Realisierung individuell gestalteter Informationsangebote bestehen in der Parametrisierung von Standardsoftware, der Erweiterung von Standardsoftware oder der Realisierung einer Individualsoftware [Hinderer 2005, S. 107]. Bedingt durch die Breite und Spezifität des Informationsbedarfs erscheint die reine Parametrisierung von Standardsoftware nicht praktikabel, da üblicherweise systemübergreifende Informationsquellen vorliegen. Eine Realisierung als Individualsoftware verbietet sich aus Aufwandsgründen und widerspricht zudem der Einbindung in die Geschäftsprozesse und die vorhandenen operativen Informationssysteme. Somit bleibt zur Realisierung eines KIP-Infosets die Erweiterung und Konfiguration der vorhandenen Standardsoftware im Rahmen der Informationsarchitektur des Unternehmens.

Intranetanwendungen und Unternehmensportale verfügen über die notwendigen technischen Grundfunktionen zur Erweiterung bestehender Anwendungen. Die Eignung des Intranets zum multimedialen Arbeitsgruppeninformationssystem mit umfassenden Produktionsdaten wurde bereits mehrfach nachgewiesen und mit prototypischen Anwendungen belegt, z. B. in [Bullinger00; Gerlach00; Neuschwinger03; Witte03; Slama04; Lang07], vgl. hierzu Kapitel 2.4.6.1. Auch mobile Geräte und Formen der Informationsbereitstellung sind mittels der Intranettechnologie realisierbar [Spath03d]. Trotzdem sind

derartige, intranetbasierte Werkerinformationssysteme in der betrieblichen Praxis nur vereinzelt anzutreffen [Spath09a, S. 230]. Die nach wie vor zu konstatierende Dominanz papiergebundener Informationsnutzung in der Produktion – trotzt der vorhandenen informationstechnischen Infrastrukturen – legt die Vermutung nahe, dass die verfügbaren Onlineinformationsangebote nicht bedarfsgerecht ausgeführt oder aufwendig zu realisieren sind.

Zur Verbesserung der effizienten und somit bedarfsgerechten Informationsversorgung bei KIP - der Zielsetzung der vorliegenden Arbeit – sind arbeitsplatz- und mitarbeiterspezifische, produktionsnahe Informationsportale gezielt für jeden Einsatzfall zu erstellen. Portalanwendungen basieren üblicherweise auf kommerziellen Portalsoftwarepaketen, ergänzt mit individuell erstellten Softwarebausteinen [Amberg05, S. 7]. Portalentwicklungen beinhalten somit in der Regel einen Anteil an Aktivitäten zur Konfiguration und zur Programmierung und sollten methodisch unterstützt erfolgen [Gaedke99]. Von den Methoden ist insbesondere eine Unterstützung bei der bedarfsgerechten Gestaltung und Konzeption der produktionsnahen Informationsangebote zu fordern. Wie alle Produktionsressourcen unterliegen auch Portale einem kontinuierlichen Verbesserungsprozess. Regelmäßig sind die Informationsangebote den wandelnden Produktionsaufgaben und Produktionsprozessen anzupassen. Portalerstellungsmethoden sollten daher auch die Anpassung und Wiederverwendung bestehender Informationsangebote unterstützen.

2.4 Erstellung produktionsnaher Informationsportale

Im Folgenden werden Methoden zur Erstellung von Informationsportalen vorgestellt und auf ihre Eignung für den Aufbau produktionsnaher Informationsangebote untersucht.

2.4.1 Abgrenzung des Untersuchungsbereichs für Portalerstellungsmethoden

2.4.1.1 Charakteristik und Merkmale der Domäne Informationsportale

Informationsportale markieren eine spezielle Domäne für Softwareanwendungen. Unter einer Domäne wird in der Informatik ein abgeschlossenes Anwendungsgebiet von Software verstanden, in dem besondere Prinzipien, Problemstellungen und Annahmen durchgängig gelten. [Schütze99, S. 81]. Häufig kommen domänenspezifische Methoden zum Einsatz, z. B. erweiterte Notationen, Referenzarchitekturen, Komponentenbibliotheken oder Frameworks [Schütze99, S. 29]. Hinsichtlich der Merkmale einer Domäne ist zu unterscheiden in funktionale und nicht-funktionale Anforderungen [Conallen99, S. 116]. Funktionale Anforderungen beschreiben den Leistungsumfang einer Software [Balzert01, S. 258ff]. Nicht-funktionale Anforderungen bestehen im Hinblick auf die gesamte Nutzungsdauer eines Softwareartefakts. Dazu zählen die Verfügbarkeit, Skalierbarkeit, Sicherheit, Wartbarkeit, Flexibilität und Portabilität [Dyson04, S. 15ff]. Obwohl die Portalentwicklung viele Vorgehensweisen mit der allgemeinen Softwareentwicklung teilt, bestehen doch Unterschiede, die eine domänenspezifische Disziplin rechtfertigen [Ginige01, S. 16]. Abweichende Charakteristika von Portalanwendungen sind: anonyme und heterogene Benutzer, Content-Orientierung, Verwendung von multimedialen und redaktionellen Inhalten, Sicherheit bei Verwendung von öffentlichen Kommunikationsstrecken, Integration von Legacy-Anwendungen, Betonung von Gestaltung der Präsentation, nichtlineare Navigation und heterogene Front-End-Clients (Webbrowser) [Murugesan01, S. 7ff]. Diese Charakteristika gelten auch für das Anwendungsgebiet der vorliegenden Ar-

beit, wenn auch einige wenige nur in abgeschwächter Form. So sind in Intranetanwendungen die Benutzer und die verwendeten Front-End-Clients bekannt. Auch sind die Sicherheitsanforderungen in einer kontrollierten Intranetumgebung nicht vorrangig.

2.4.1.2 Elemente von Methoden zur Erstellung von Informationsportalen

Vor der weiteren Untersuchung ist festzulegen, welche Unterstützung bei der Portalentwicklung von einer Methode zu erwarten ist. Da Informationsportale Softwareartefakte sind, werden die methodischen Elemente des Softwareengineering herangezogen. Softwareengineering ist eine ingenieurmäßige Disziplin, die eine zielorientierte und arbeitsteilige Vorgehensweise auf die Entwicklung umfangreicher Softwareprodukte überträgt [Balzert00, S. 36]. Sie stellt Vorgehensweisen, Werkzeuge und Modelle bereit [Schütze99, S. 18ff], s. Abbildung 2-12. Vorgehensweisen gliedern und ordnen die Entwicklungsaktivitäten und werden durch Werkzeuge unterstützt. Modelle sind formalisierte Notationen der entstehenden Produkte oder der benutzten Werkzeuge und Vorgehensweisen.

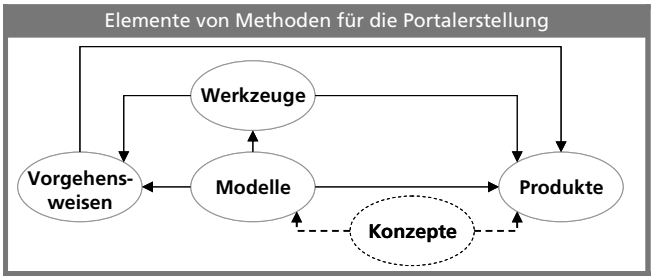


Abbildung 2-12: Elemente einer unterstützenden Methode. Nach: [Schütze99, S. 18]

Erfolgreiches Softwareengineering beruht auf Prinzipien, d. h. abstrakten, empirisch bestätigten Grundsätzen des Handelns. Beispielsweise gelten für die ergonomische Gestaltung von Bildschirmarbeitsplätzen die Prinzipien der Aufgabenangemessenheit, Selbstbeschreibungsfähigkeit, Steuerbarkeit, Erwartungskonformität und Fehlerrobustheit. Prinzipien sagen jedoch nichts darüber aus, wie sie erreicht werden [Balzert00, S. 36]. Ähnlich verhält es sich mit Vorgehensweisen. Sie sind Anleitungen um die Aufgabenstellungen strukturiert und zielorientiert zu bewältigen, führen jedoch nicht zwangsläufig zu „guten“ Softwareprodukten [Schütze99, S. 53]. Eine erfolgreiche Produktentwicklung ist maßgeblich von der Erfahrung der Entwickler bestimmt, die auf bekannte und bewährte Lösungskonzepte zurückgreifen [Buschmann98, S. 1]. Lösungskonzepte zeigen Ausführungsformen für die Produktgestalt. Im Unterschied zu abstrakten Prinzipien sind Lösungskonzepte konkrete Vorlagen. Von handlungsleitenden Vorgehensweisen unterscheidet sie ihre konstruktive, gestaltgebende Natur. Die methodischen Elemente des Softwareengineerings werden daher um das Element *Konzept* erweitert, das konkrete Lösungsvorschläge bezeichnet, s. Abbildung 2-12. Eine methodische Unterstützung der Portalentwicklung besteht somit aus geeigneten Werkzeugen, Vorgehensweisen, Modellen und Konzepten. Nachfolgend werden die hierfür bekannten Ansätze vorgestellt.

2.4.2 Werkzeuge zur Realisierung von Informationsportalen

Die Grundstruktur von Portalanwendungen ergibt sich aus der Internettechnologie. Resultat ist eine Trennung in einen Front-End-Webbrowser, mehreren Back-End Anwendungen mit Datenquellen und einer Portal-Engine, die Basisdienste für den Betrieb, die Portaladministration sowie die Inhaltspräsentation bereitstellt [Huang03; Davydov01, S. 172]. Realisiert werden sie mehrheitlich mittels Portalwerkzeugen. Eine Marktstudie mit 51 verfügbaren Werkzeugen bietet [Spath05a], eine Übersicht der Technologien zur Realisierung von Webanwendungen, Portalen und Integrationslösungen geben [Dumke03; Jablonski04; Keller02]. Die Basisdienste sehen üblicherweise vorkonfigurierte, universelle Anwendungen, Unterstützungsfunktionen und technische Funktionen vor. Anwendungen und Unterstützungsfunktionen sind für den Benutzer sichtbar. Dabei stiften Anwendungen Nutzen, während Unterstützungsfunktionen nur der Administration oder der Anpassung von Portalen dienen. Technische Funktionen bleiben dem Benutzer verborgen, sind für den Betrieb eines Portals allerdings notwendig, z. B. Integrations- und Transaktionsdienste, Datenzugriff und Konsolidierung. Ebenfalls Teil der technischen Funktionen sind Programmierschnittstellen zur Realisierung spezieller Anwendungen. Abbildung 2-13 zeigt die technische Grundstruktur mit einer Übersicht der typischerweise genannten Basisdienste, siehe [Röhricht01, S. 171ff; Davydov01; Rösch00, S. 253ff; Spath05a; Huang03; Bach05, S. 67ff].

Zur Realisierung von Informationsportalen stehen mittlerweile etablierte kommerzielle Werkzeuge zur Verfügung. Bedingt durch die vergleichsweise junge Web-Technologie und den Wettstreit der Hersteller, existieren mehrere alternative Implementierungspfade [Jablonski04, S. 8]. Die Entwickler von produktionsnahen Informationsportalen sind somit konfrontiert mit einer heterogenen, betrieblichen Informationsarchitektur und einem breiten Spektrum an Implementierungstechnologien.

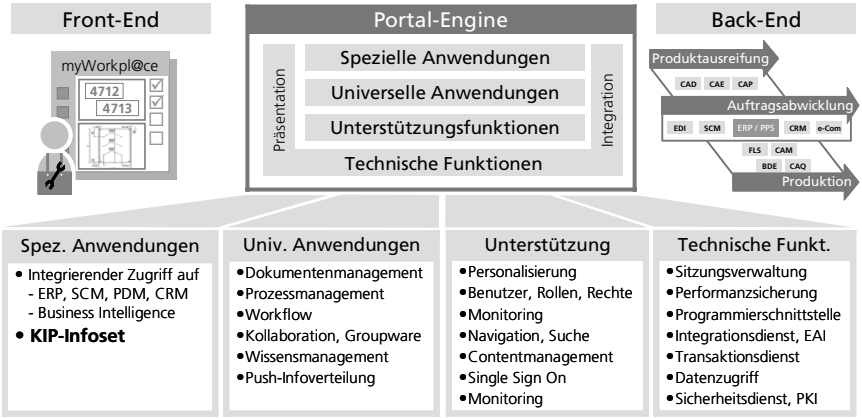


Abbildung 2-13: Basisdienste von Portalanwendungen. Nach: [Huang03]

2.4.3 Vorgehensweisen zur Portalentwicklung

Zur Erstellung von Informationsportalen, die zugleich Webanwendungen sind, dienen die domänenspezifischen Vorgehensweisen des Web-Engineerings [Ceri03; Murugesan01; Kappel04; Mendes06; Rossi07; Brandon08; Pressman09]. Dessen Definition orientiert sich am Softwareengineering und zielt auf den Einsatz systematischer und quantifizierbarer Verfahren zur effizienten Entwicklung qualitativ hochwertiger Web-Anwendungen [Lowe99, S. 90; Kappel04, S. 4; Mendes06, S. 2]. Web-Engineering ist ausgeprägt interdisziplinär. Neben Softwareentwicklern sind Mediengestalter, Informationsdesigner und Usability-Experten beteiligt [Mendes06, S. 9; Kappel04, S. 18]. Typisch sind zudem agile Vorgehensweisen, die besser mit kurzen Entwicklungszyklen, wechselnden Anforderungen und der ausgeprägten Wiederverwendung vorhandener Softwarebausteine harmonieren [Pressman09, S. 19; Kappel04, S. 245]. Web- und Softwareengineering teilen sich die grundsätzlichen Aktivitäten, die in der Literatur in unterschiedlicher Granularität beschrieben sind, vgl. z. B. [Eriksson00, S. 354]. Stets auszuführen sind die Schritte Analyse, Konzeption und Implementierung, die korrespondierende Problemsichten einnehmen. Aufgabe der Analyse ist die Erhebung der fachlich-funktionalen Anforderungen aus Anwendersicht. Die Konzeption überführt diese in einen Entwurf. Bei der Implementierung rücken softwaretechnische Aspekte in den Fokus, z. B. Systemintegration, Wiederverwendung oder Systemlast [Schütze99, S. 20f; Hahsler01, S. 18; Fowler98, S. 62].

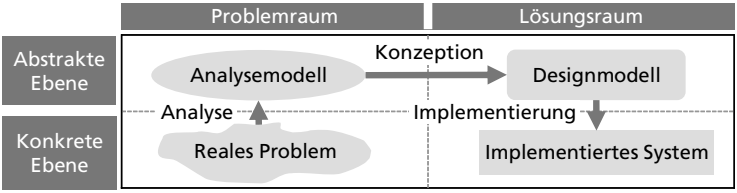


Abbildung 2-14: Aufgaben und Sichten der Softwareentwicklung. Nach: [Hahsler01, S. 18]

Ausgehend von den Entwicklungsaktivitäten sind verschiedene Vorgehensweisen bekannt. Dabei sind phasenorientierte bzw. iterative Abläufe sowie schwergewichtige bzw. leichtgewichtige Prozesse zu unterscheiden [Balzert01, S. 98; Hagen05, S. 34]. In phasenorientierten Prozessen erfolgt die Softwareentwicklung in einem Durchgang. Bei iterativen Entwicklungen werden Rumpfapplikationen frühzeitig evaluiert und die Erkenntnisse in nachfolgenden Iterationszyklen verwertet. Iterative Prozesse eignen sich unter anderem bei unklaren Vorstellungen des Auftraggebers: *„I can´t tell you what I want, but I´ll know it when I see it"* [Boehm88, S. 63]. Schwergewichtige Prozesse sehen eine detaillierte Gliederung und Dokumentation der Aktivitäten und Produkte im Softwarelebenszyklus vor [Balzert01, S. 101]. Neuere Bestrebungen tendieren zu agilen und leichtgewichtigen Prozessen. Diese vermeiden bürokratische, überspezifizierte Koordinationsaktivitäten und konzentrieren sich auf die effiziente Produktentwicklung in sehr kurzen Iterationszyklen und hoch-kommunikativen, kleinen Teams [Beck00, S. 54 ff]. So unterschiedlich die Vorgehensweisen sind, allen gemeinsam ist ein Mikroprozess mit der Abfolge von Anforderungsdefinition, Konzeption und Implementierung: *„Fundamentally, you first need to do some analysis, then some design, then some coding. There is no way around it, and there is not a single piece of software that can justifiably be developed any other way."* [Ambler98, S. 18f]. Für die vorliegende Arbeit wird davon ausgegangen, dass jede Softwareentwicklung immer in den genannten Schritten erfolgt, unabhängig von der übergeordneten Vorgehensweise.

2.4.4 Modelle für das Web-Engineering

Formale Modelle unterstützen eine systematische und quantifizierbare Entwicklung von Web-Anwendungen [Wissen05, S. 18]. Sie reduzieren die Problemkomplexität, dokumentieren Entwurfsentscheidungen und erleichtern die Kommunikation zwischen Entwicklern [Schwinger04, S. 50]. Im Rahmen des Web-Engineerings sind zahlreiche, domänenspezifische Modellierungsformen entstanden. Eine grobe Einteilung kann nach datenorientierten und objektorientierten Modellen erfolgen [Schwinger04, S. 68]. Datenorientierte Modelle gründen auf der Entity Relationship Notation (ER), erweitern diese um Navigations- und Hypertextstrukturen und legen den Fokus auf vorwiegend datengetriebene Web-Anwendungen, z. B. die Web Modelling Language (WebML) [Ceri03]. Objektorientierte Modelle verwenden Klassendiagramme zur Domänenmodellierung und verbinden diese ebenfalls mit Navigationsstrukturen und einem abstrakten Oberflächenentwurf, z. B. die Object-Oriented Hypermedia Design Method (OOHDM) [Rossi06]. Häufig sind objektorientierte Modelle Derivate der Unified Modelling Language (UML) mit Stereotypen für die Web-Entwicklung. Beispiele hierfür sind UML-based Web Engineering (UWE) [Schwinger04] und die Web Application Extensions (WAE) [Conallen99]. Neuere Entwicklungen basieren auf Ontologien und sehen hierfür eine explizite Analyse und Modellierung der Fachdomäne vor. Dabei wird ein formales Begriffssystem der im Anwendungsbereich eingesetzten Objekte, Konzepte und Daten erstellt, die zumindest teilweise in die Web-Anwendung eingehen [Wissen05, S. 38]. Einen Überblick und Vergleich der bekannten Modellierungsformen bieten z. B. [Escalona04; Woukeu03; Rossi07].

2.4.5 Konzepte zur Gestaltung von Informationsportalen

Als letztes und im Rahmen der Arbeit bedeutsamstes methodisches Element wird der Stand der Forschung hinsichtlich bekannter Konzepte für die arbeitsplatz- und mitarbeiterspezifische Gestaltung produktionsnaher Informationsportale dargelegt. Zur gegenseitigen Abgrenzung der Konzepte wird zuvor ein Ordnungsrahmen festgelegt.

2.4.5.1 Ordnungsrahmen für Konzepte

Generell sind die zueinander komplementären Problem- und Lösungsdomänen zu unterscheiden [Buschmann07, S. 218]. Eine **Problemdomäne** bezieht sich auf Aktivitäten oder Nutzungsinteressen des Anwenders und beschreibt den Einsatzbereich und das Fachgebiet der Softwareanwendung [Evans04, S. 2]. Die Elemente der hier betrachteten Problemdomäne entsprechen dem erhobenen Informationsbedarf. Als vorrangig erkannt wurden Informationen bezüglich des Produktionsumfelds, der Auftragsabwicklung, der PPS und Kennzahlen, s. Kapitel 2.2, die in einem Portal als Teil der speziellen Anwendungen zu implementieren sind. Weitere Elemente ergeben sich aus den universellen Anwendungen und den Unterstützungsfunktionen von Portalen, s. Kapitel 2.3.2.3. **Lösungsdomänen** adressieren die innere Struktur einer Software. Sie entsprechen Querschnittsfunktionen, die in vielen Problemdomänen benötigt werden. Die Lösungsdomäne für funktional umfangreiche Webanwendungen mit einem nennenswerten Softwareanteil, denen Portale zuzurechnen sind, lässt sich gliedern nach einem zweidimensionalen Schema [Austaller04, S. 103f], s. Abbildung 2-15.

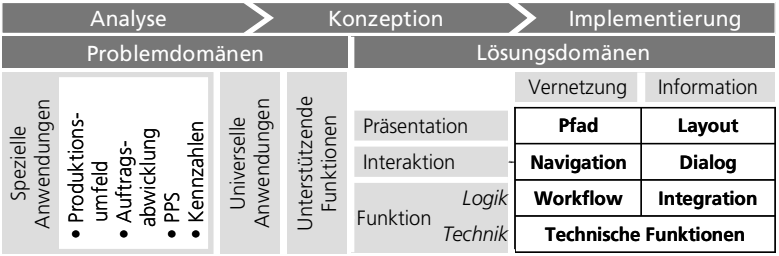


Abbildung 2-15: Domänenprofil für Informationsportale. Nach: [Austaller04]

Die vertikale Ebenengliederung in Präsentation, Interaktion und Funktion ist in vielen interaktiven Anwendungen anzutreffen. Charakteristisch für Web-Anwendungen ist eine weitergehende, horizontale Trennung dieser Ebenen in die zwei Säulen Vernetzung und Information. Die Präsentation der Vernetzung erfordert Angaben zur Position im Netz und zum Pfad, auf dem diese Position erreicht wurde. Eine Interaktion mit der Vernetzung erfolgt zum Zweck der Navigation und ermöglicht das aktive Aufsuchen anderer Informationsinhalte. Verarbeitungsfunktionen werden in Portalen zu Workflows vernetzt und orchestriert. Im Block Informationspräsentation findet die Layoutgestaltung und multimediale Darstellung der Inhalte statt. Interaktionen mit den Informationen initiieren Eingabe- und Verarbeitungsdialoge. Die eigentliche Informationsverarbeitung erfolgt bei Webanwendungen üblicherweise in externen Komponenten. Daher enthält die Funktionsebene vorwiegend Aufgaben zu deren Einbindung [Austaller04, S. 115]. Neben den Integrationsfunktionen fallen für Portale weitere portalspezifische, technische Aufgaben an, z. B. Single-Sign-On oder Caching, vgl. Kapitel 2.4.2. Eine Trennung der Ebene Funktion in zwei Ebenen mit fachlich-logischen und technischen Funktionen sorgt hier für mehr Klarheit.

Alle Anwendungen für die Problemdomäne sind mit Funktionsbausteinen der Lösungsdomäne zu realisieren, wobei häufig viele oder gar alle der Bausteine benötigt werden. Problem- und Lösungsdomäne können bei dieser Definition als orthogonal angesehen werden. Wie ausgeführt, erfordert die Realisierung eines Portals immer die Schritte Analyse, Entwurf und Implementierung. Naturgemäß stehen in der Analyse stärker die Elemente der Problemdomäne im Vordergrund, während in der Konzeption und Implementierung die Lösungsbausteine betrachtet werden. Ausgehend von dem definierten Ordnungsrahmen werden nachfolgend die bekannten konzeptionellen Ansätze vorgestellt.

2.4.5.2 Problemorientierte Konzepte für produktionsnahe Informationsangebote

Gemäß ihrer Bedeutung wurde die Thematik der werker- und anforderungsgerechten Informationsbereitstellung in der Produktion bereits frühzeitig und öfter aufgegriffen. Die folgende Übersicht enthält Veröffentlichungen, die intranetbasierte Anwendungen forcieren und somit nach ca. 1995 erschienen. Ältere Ansätze berücksichtigen die seither stattgefundene technologische Entwicklung und die daraus resultierenden visuellen und ergonomischen Gestaltungsmöglichkeiten nicht ausreichend in ihrer Konzeption.

Kuba entwickelt ein objektorientiertes Informations- und Kommunikationsmodell der Produktion unter expliziter Berücksichtigung von Mitarbeitern als Modellelemente [Kuba97, S. 88f]. Dabei werden hauptsächlich das Kommunikationsverhalten und die Kommunika-

tionszustände abgebildet [Kuba97, S. 115ff]. Anforderungen der Mitarbeiter an die Informationsdarstellung und die Gestaltung der Benutzungsschnittstelle werden von Kuba zwar genannt, allerdings nur in Form allgemeiner Prinzipien, z. B. der DIN-Norm 66234 zur Gestaltung von Bildschirmarbeitsplätzen [Kuba97, S. 60]. Eine Umsetzung der Anforderungen wird nicht gezeigt.

Neuschwinger fokussiert auf die Erfassung und den Austausch des Erfahrungswissens der Facharbeiter in produktionsnahen Arbeitsgruppen, das zusammen mit den gebräuchlichen Fertigungsunterlagen und den Fertigungsaufträgen in einem möglichst einheitlichen Informationssystem bereitgestellt wird [Neuschwinger03, S. 4, 41]. Das prototypisch realisierte System ermöglicht den Anwendern, Fertigungswissen in multimedialer Form zu erfassen, Fertigungsaufträgen zuzuordnen, nach Bearbeitungsverfahren, Betriebsmittel und Teilebezeichnung zu klassifizieren und später zu recherchieren [Neuschwinger03, S. 125]. Eine wesentliche Nutzungsvoraussetzung eines derartigen Werkzeugs sieht Neuschwinger in der ergonomischen und individualisierbaren Benutzungsoberfläche [Neuschwinger03, S. 45]. Die ergonomische Gestaltung des Systems orientiert sich an den üblichen Gestaltungsgrundsätzen, z. B. der Verwendung von Gruppierungen und Hervorhebungen von Informationsinhalten [Neuschwinger03, S. 97]. Zur Anpassung der Benutzungsoberfläche an mitarbeiterspezifische Bedürfnisse sind die Darstellungselemente der Bildschirmmasken fallweise ein- bzw. ausblendbar [Neuschwinger03, S. 111].

Feldmann und Slama beschreiben einen ganzheitlichen Ansatz für die Gestaltung von modularen und flexiblen Arbeitssystemen für die variantenreiche Kleinserienmontage, der auch den Einsatz eines multimedialen Mitarbeiterinformationssystems vorsieht. Zielsetzung ist die Fehlervermeidung, die Steigerung der Prozesseffizienz sowie die Unterstützung der Selbstorganisation von Montagegruppen [Feldmann04, S. 234; Slama04, S. 130]. Das Informationsangebot umfasst neben auftragsneutralen Arbeitsanweisungen, Montageunterlagen und Qualitätsdaten auch einen Auftragsvorrat für den Arbeitsplatz nebst Rückmeldefunktionen. Von der Informationsbereitstellung wird ein ergonomisches, möglichst einfaches Bedienkonzept mit berührungssensitiven Bildschirmen und Barcodelesegeräten verlangt [Feldmann04, S. 236f; Slama04, S. 134f]. Weiterhin wird gefordert, den Detaillierungsgrad der Informationen dem Qualifikationsniveau des Mitarbeiters anzupassen. Konkrete Vorschläge zur Gestaltung der Informationstiefe beschränken sich allerdings auf die Filterung des Auftragsvorrats nach dem Qualifikationsstand des Mitarbeiters sowie der Verwendung von Videosequenzen zum Einlernen neuer Mitarbeiter [Feldmann04, S. 242; Slama04, S. 138]. Der Wissensstand der Mitarbeiter wird ebenfalls bei der Auftragsanmeldung geprüft. Aktualisierte Arbeitsanweisungen oder sonstige aktuelle Dokumente erhält der Mitarbeiter angezeigt und muss die Kenntnisnahme durch seinen Vorgesetzten bestätigen lassen [Slama04, S. 160]. Das Konzept für das Montageinformationssystem sieht eine grobe Architektur und eine Anbindung an externe Standardanwendungen, z. B. PPS-Systeme, vor [Slama04, S. 136f]. Eine mitarbeiter- oder aufgabenspezifische Aufbereitung von Informationen ist nicht vorgesehen, lediglich die Filterung der Auftragsliste nach der Arbeitsstation.

Lang entwickelt die Konzeption von Feldmann und Slama weiter und erstellt einen Leitfaden für die ergonomische Gestaltung von Arbeitsanweisungen hinsichtlich multimedialer Informationspräsentation, Verständlichkeit und Nutzbarkeit [Lang07, S. 41]. Im Hinblick auf eine rechnergestützte Informationsbereitstellung empfiehlt er, Arbeitsanweisungen nach dem Prinzip „Keep it simple and stupid“ (KISS) zu gestalten und bringt in knapper Form noch einige wenige Grundsätze zum Bildschirmaufbau und zur Navigation [Lang07,

S. 63 & 70]. Für die rechnerunterstützte Bereitstellung wird ein intranetbasiertes Werkzeug auf Basis der Konzeption von Feldmann und Slama prototypisch realisiert, das ebenfalls eine Rückmeldemöglichkeit, stationsabhängige Auftragsvorratslisten, Kenntnisstandprüfung und eine Informationstiefe entsprechend der Mitarbeiterqualifikation vorsieht. Außer der bereits genannten Verwendung von Videosequenzen zur Mitarbeiterschulung werden auch hier keine weiteren Gestaltungsmöglichkeiten der Informationstiefe genannt [Lang07, S. 122]. Den Schwerpunkt der Arbeit von Lang bildet ein Regelwerk für die Gestaltung von Arbeitsanweisungen sowie deren teilautomatisierte und wissensbasierte Generierung aus Vorranggraphen. Die Umsetzung in ein produktionsnahes Informationssystem wird nur exemplarisch gezeigt. Hierfür erforderliche, technisch orientierte Konzepte fehlen.

Raupach betrachtet wie Lang die ergonomische Gestaltung multimedialer Arbeitsanweisungen [Raupach08, S. 7]. Ausgehend von den Grundsätzen der Ergonomie – Ausführbarkeit, Schädigungslosigkeit, Beeinträchtigungsfreiheit und Persönlichkeitsförderlichkeit [Raupach08, S. 21] – leitet Raupach 37 Prüf- und Beurteilungskriterien für multimediale Arbeitsanweisungen ab und führt zwei Versuchsreihen mit jeweils unterschiedlich strukturierten Arbeitsplänen durch. Die Versuchergebnisse dokumentiert sie in einem Regelkanon mit Gestaltungsempfehlungen für Arbeitsanweisungen [Raupach08, S. 120ff]. Die Gestaltungsempfehlungen sind allgemeiner Art, eine Berücksichtigung aufgaben- oder mitarbeiterspezifisch ausgeprägter Informationsbedarfe erfolgt nicht.

Gaida untersucht die Anwendbarkeit, den Nutzen und die Benutzerakzeptanz von Informationssystemen für ein Total Quality Management auf der Ebene von Arbeitsgruppen [Gaida00, S. 37]. Dazu wurden die zu unterstützenden, qualitätsrelevanten Gruppenprozesse und deren Informationsbedarf erhoben und anschließend in einer iterativen Vorgehensweise mehrere explorative und funktionale Prototypen erstellt [Gaida00, S. 93]. Das realisierte Informationsangebot deckt u. a. Qualitätskennzahlen und Trainingsfunktionen sowie die Koordination und die Kommunikation in Qualitätsgruppen ab. [Gaida00, S. 44]. Die Evaluation durch IT-Experten und durch Anwender in Unternehmen erfolgte auch im Hinblick auf die Gestaltung der Benutzungsoberfläche und der Softwareergonomie [Gaida00, S. 99]. Insgesamt konnte Gaida mit der Vorgehensweise die kritischen Erfolgsfaktoren für die Einführung von Qualitätsinformationssystemen identifizieren und die generelle Vorteilhaftigkeit belegen [Gaida00, S. 115]. Außer einer Anforderungsdefinition und einigen allgemeinen Grundsätzen für die Oberflächengestaltung beschreibt er jedoch keine konkreten Konzepte für die Ausführung einer derartigen Software.

Gerber erhebt den Informationsbedarf für ein umfassendes Qualitätsinformationssystem und entwickelt eine Vorgehensweise für die Anforderungsdefinition vor dem Hintergrund der speziellen Situation von kleinen Unternehmen [Gerber08, S. 4]. Ergebnis ist ein Pflichtenheft für die Auswahl und Realisierung eines Qualitätsinformationssystems, das auch Kommunikationsbeziehungen und Rollenkonzepte aufzeigt [Gerber08, S. 104ff]. Weitergehende Anforderungen an mitarbeiter- oder arbeitsplatzspezifische Informationsbedarfe werden nicht aufgezeigt, ebenso finden sich keine konzeptähnlichen Vorgaben an die Systemrealisierung.

Wilhelm fordert eine Content-Orientierung im Qualitätsmanagement von produzierenden Unternehmen und beschreibt ein Verfahren zur Einführung eines intranetbasierten Content Managements im Produktionsbereich. Content-Orientierung bedeutet dabei die standardisierte Erzeugung und Speicherung aller Informationsträger getrennt nach Inhalt,

Struktur und Format. Ziel ist die wirtschaftliche, flexible und bedarfsgerechte Wiederverwendung und Weiterverarbeitung der Informationen [Wilhelm06, S. 23]. Betrachtungsgegenstände sind Arbeitsanweisungen, Prüfpläne, Betriebsvorschriften, Zeichnungen, Stücklisten aber auch Ad-hoc Informationen für die Mitarbeiterkommunikation [Wilhelm06, S. 67f]. Das beschriebene Verfahren erlaubt die Verankerung und Unterstützung der Content-Orientierung in allen Vorgängen zur Informationserzeugung. Die Thematik der Informationspräsentation und –aufbereitung ist nicht im Fokus der Arbeit.

Sternemann sucht für Gruppenarbeitsstrukturen eine neue Form der dezentralen Bereitstellung von Informations- und Wissensbeständen aus originären Datenbeständen [Sternemann00, S. 7]. Exemplarisch betrachtet werden prozessrelevante Dokumente, Informationen und Wissen aus dem Umfeld von genehmigungspflichtigen, verfahrenstechnischen Anlagen, z. B. Prozessparameter, Auflagen, Genehmigungen und Anleitungen. Das konzipierte intranetbasierte Teaminformationssystem unterstützt die Vernetzung vorhandenen Wissens in einem Informationsraum mit den Dimensionen Prozess, Wirkungsbereich und Informationsaspekt [Sternemann00, S. 67]. Der Informationsraum entspricht einer Verwaltungsstruktur für bestehende, externe Informationsobjekte, die mittels eines Strukturmanagers manuell im Informationsraum verzeichnet, klassifiziert und mit Zugriffsrecht, Status und Verfallsdatum versehen werden [Sternemann00, S. 72, 86, 92]. Eine Navigation entlang der Achsen des Informationsraums anhand von Strukturdiagrammen und über Informationsbeziehungen erlaubt die Recherche nach Informationsobjekten [Sternemann00, S. 114ff]. Sternemann zeigt einen Ansatz für Wissensmanagement in der Produktion durch die explizite Vernetzung von bestehenden Informationsobjekten, aber keine Anforderungen hinsichtlich der Informationspräsentation.

Linsenmaier zeigt die Realisierung eines integrierten Montageinformationssystems, unter Verwendung von Business Objekten als Schicht zwischen den Applikationssystemen und den zur Informationsbereitstellung und Dialogsteuerung eingesetzten Intranetdiensten [Linsenmaier99, S. 83]. Schwerpunkte des beschriebenen Konzepts bilden die Architektur der Integrationsschicht und der technische Aufbau der Business Objekte mittels CORBA. Fragen der Informationsgestaltung und Präsentation werden nicht adressiert.

Gelhaar beschreibt detailliert den erweiterten Informationsbedarf von Arbeitsgruppen und Facharbeitern in der Fertigung mit Mehrmaschinenbedienung. Wesentliche Informationsmodule betreffen die Auftragsreihenfolge, die Werkzeugverwaltung, die Maschinensteuerung und –überwachung, die Betriebsdatenerfassung sowie die Prozessdokumentation und schichtübergreifende Abstimmung [Gelhaar95, S. 5]. Ein objektorientierter Applikationsgenerator konfiguriert aus diesen Modulen ein arbeitsplatzspezifisches Informationssystem. Dabei werden auch die Menüstruktur und die Funktionstastenbelegung gemäß den Wünschen des Anwenders konfiguriert [Gelhaar95, S. 142]. Der Bedarf nach einem arbeitsplatz- und mitarbeiterspezifischen Informationsangebot wird von Gelhaar zwar adressiert. Die Umsetzung orientiert sich jedoch an den typischen Gestaltungskonventionen einer werkstattorientierten Programmierung und Steuerung.

Keines der Konzepte für produktionsnahe Informationsangebote basiert explizit auf Informationsportalen. Daher werden zusätzlich Konzepte für Portalanwendungen in produktionsfernen Bereichen herangezogen.

2.4.5.3 Problemorientierte Konzepte für Informationsportale

Für Informationsportale wurde eine Reihe von problemorientierten Konzepten in Form von Portalarchitekturen beschrieben. Architekturen strukturieren eine Software, regeln die Zuständigkeiten und Anwendungsfunktionen der einzelnen Bausteine, zeigen deren Beziehungen und Schnittstellen und überführen dadurch Anforderungen in einen Entwurf [Eichinger04, S. 78]. Der Grad der Detaillierung und Konkretisierung der Architekturen ist stark unterschiedlich. So beschränken sich die Arbeiten von [Collins01], [Fingar00], [Hinderer03] und [Röhricht01] auf Blöcke von Basis- und Universalfunktionen für B2E- und B2B-Portale. Beziehungen zwischen den Funktionsbausteinen werden nicht gezeigt. Die Arbeiten entsprechen somit im Wesentlichen einem Anforderungskatalog der Problemdomäne. Konkreter ausgestaltet sind die Architekturen von [Davydov01], [Galic04], [Grossmann05], [Gurzki06], [Puschmann03] und [Sullivan04]. Die Konzepte berücksichtigen die Anwendungsfunktionen zusammen mit grundlegenden Technikkomponenten und stellen einen Wirkungszusammenhang her, sodass sich die Aufgaben und Zuständigkeiten der aktiven Komponenten eines Portals erkennen lassen. In Abbildung 2-16 ist ein Beispiel einer Portalarchitektur aus [Galic04] wiedergegeben.

Wie alle der hier genannten Arbeiten, ist auch die gezeigte Architektur von Galic als Blockschaltbild ausgeführt und verwendet kein formales Modell mit einer definierten Semantik. Daraus ist ersichtlich, dass Architekturen auf dieser hohen Aggregationsstufe bestenfalls eine Zuständigkeitsverteilung zeigen, aber noch kein realisierbares Lösungskonzept darstellen.

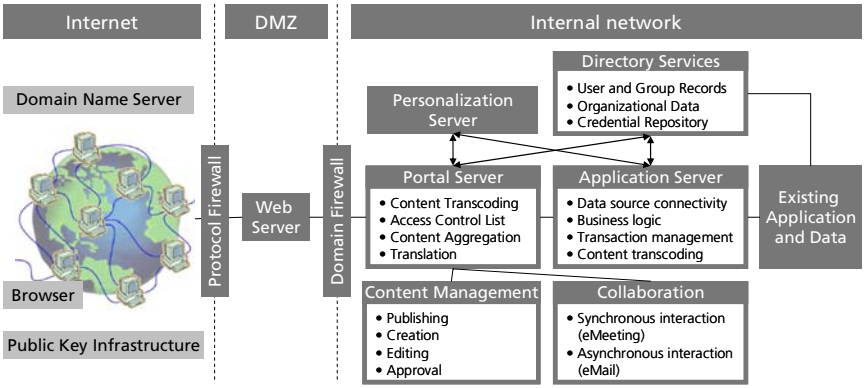


Abbildung 2-16: Beispielhafte Portalarchitektur. Nach: [Galic04, S. 90]

2.4.5.4 Lösungsorientierte Konzepte für Informationsportale

Die Verfahren des Web-Engineerings beinhalten oft domänenspezifische, lösungsorientierte Konzepte für Webanwendungen und Informationsportale. Insbesondere die Arbeiten von [Ceri03; Kappel04; Pressman09] bieten eine breite Sicht der bekannten Konzepte für die Lösungsdomäne. Weitere Arbeiten vertiefen technische Entwurfsaufgaben und zeigen hierfür Konzepte und Architekturen auf. Beispiele sind Referenzarchitekturen und Gestaltungsansätze für performante Web-Anwendungen [Anastopoulos01; Menasce00], für verteilte Anwendungen und Datenhaltung [Goodyear00] oder für die Konsolidierung

umfangreicher Datenbestände in Business Intelligence Lösungen [Finkelstein00]. Häufig orientieren sich die genannten Konzepte an nicht-funktionalen Anforderungen für Web-Anwendungen [Eichinger04, S. 77]. Beispielsweise werden in dem in Abbildung 2-23 dargestellten Konzept zur Verteilung von Datenbanken die Wechselwirkungen mit Kriterien der Sicherheit, Flexibilität, Skalierbarkeit etc. ausgeführt.

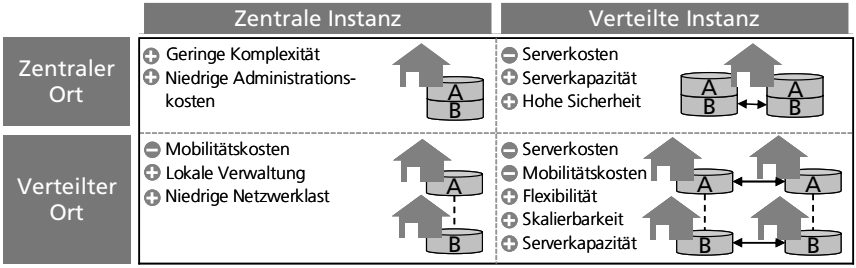


Abbildung 2-17: Beispiel für ein lösungsorientiertes Konzept. Nach: [Goodyear00, S. 17/31]

2.5 Vergleich und Bewertung der Methoden

In den vorhergehenden Abschnitten wurden bekannte Werkzeuge, Vorgehensweisen, Modelle und Konzepte für die Erstellung von Informationsportalen beschrieben. Die Erhebung ergab, dass etablierte **Werkzeuge, Vorgehensweisen** und **Modelle** zur Realisierung von Informationsportalen verfügbar sind. Ohne diese im Detail darzustellen, kann davon ausgegangen werden, dass die Technologien auch im Produktionsumfeld einsetzbar und produktionsnahe Informationsportale technisch realisierbar sind.

Es bleibt die **Konzepte** zu bewerten, hinsichtlich ihrer Anwendbarkeit für produktionsnahe Informationsportale. Die Bewertungskriterien für Konzepte ergeben sich aus dem Informationsraum, den Anforderungen an ein Informationsangebot und dem Ordnungsrahmen für Konzepte, s. Tabelle 2-2.

Anforderung	Erläuterung
Komplett	Konzepte für den gesamten produktionsnahen Informationsbedarf, sowohl für Kern- als auch Komplementäraufgaben. *(Produktionsumgebung, Auftragsinformationen, PPS, Kennzahlen)*
Problemorientiert konzeptionell	Konzepte für ein selektives Informationsangebot nach dem Prinzip 5S. *(arbeitsplatzspezifische Kerninformationen, flankierende Komplementärinformationen, sporadische Notifikation, bidirektionale Kommunikation, Personalisierung)*
Lösungsorientiert konzeptionell	Konzepte für die Realisierung eines produktionsnahen Portals und dessen Integration in die vorhandene Informationsarchitektur. *(Präsentation, Interaktion und Funktion von Vernetzung und Information)*
Konstruktiv	Unterstützung aller Entwicklungssichten. *(Analyse, Entwurf, Implementierung)*

Tabelle 2-2: Anforderungen an Konzepte zur Realisierung produktionsnaher Informationsportale

Direkt verwendbar für den Gegenstandsbereich der vorliegenden Arbeit sind die **problemorientierten Konzepte für produktionsnahe Informationsangebote**, s. Kapitel 2.4.5.2. Alle genannten Arbeiten zeigen Konzepte nur für fokussierte Informationsbedarfe oder einzelne Gestaltungsaspekte auf. So betrachten Gaida, Gelhaar, Gerber und Wilhelm Informationsbedarfe des Qualitätsmanagements. Lang, Neuschwinger und Slama stellen Informationsangebote für auftragsneutrale Produktionsunterlagen, Arbeitspläne, Verfahrensanweisungen, Zeichnungen, Betriebsmitteldaten etc. in den Mittelpunkt der Betrachtungen. Den Schwerpunkt auf Gestaltungsaspekte des Informationsangebots legen Kuba mit Kommunikationsfunktionen, Linsenmaier mit Integrationsfunktionen, Raupach mit Ergonomiekriterien sowie Sternemann mit Vernetzungskonzepten für originäre Datenbestände. Forderungen nach der Einbindung der Werker in den Informationsfluss – auch bidirektional zur Kommunikation - sowie der ergonomischen und mitarbeiterspezifischen Gestaltung der Informationsangebote finden sich öfter. Diesbezügliche Konzepte zur Informationsaufbereitung sind jedoch zumeist nur rudimentäre Filterfunktionen oder auf ausgewählte Informationsbedarfe abgestellte Konzepte, z. B. das Regelwerk zur Formulierung von Arbeitsplänen von Lang. Keine der Arbeiten berücksichtigt die Abwicklung von kundenindividuell ausgeprägten Aufträgen und die daraus resultierenden Informationsbedarfe. Die Notwendigkeit einer arbeitsplatzspezifischen Ausprägung der Informationsinhalte wurde weder untersucht noch mit entsprechenden Konzepten hinterlegt. Implementierungsaspekte werden nur stark verkürzt behandelt.

Die **problemorientierten Konzepte für Informationsportale** beziehen sich auf Anwendungen für die externe Informationsbereitstellung im Bereich B2B oder B2C oder auf personalwirtschaftlich orientierte Mitarbeiterinformationsportale (B2E). Häufig liegt der Schwerpunkt eher auf der Prozessgestaltung und dem Business Engineering mittels Portalen, weniger auf der Portalgestaltung. Direkt verwendbare Konzepte für ein selektives und reduziertes Angebot an kundenindividuellen Auftragsinformationen im produktionsnahen Bereich bietet keine der Arbeiten.

Gleiches gilt für die **lösungsorientierten Konzepte für Informationsportale**. Die primär auf die Beschreibung des Web-Engineering ausgerichteten Arbeiten beschränken sich auf Übersichten der bekannten Konzepte, vertiefen diese aber nicht. Detaillierte Konzepte finden sich nur für fokussierte, technische Fragestellungen, zum Beispiel zur Konsolidierung von verteilten Datenbeständen oder für anwendungsneutrale Portalbasisfunktionen. Tabelle 2-3 zeigt die Erfüllung der Anforderungen durch die genannten Konzepte.

Zusammenfassend ist festzustellen, dass Modelle, Werkzeuge und Vorgehensweisen vorhanden sind, es aber an durchgängigen und vollständigen Problem- und Lösungskonzepten mangelt, s. Abbildung 2-18. Keine der Arbeiten bietet eine komplette, konzeptionelle und konstruktive Unterstützung der gewünschten Form. Für jede produktionsnahe Portalanwendung muss „das Rad neu erfunden“ werden. Einzelaspekte sind durchaus konzeptionell unterlegt. Jedoch ist kein Konzept bekannt, das für alle erforderlichen Sichten und Aufgaben der Realisierung von produktionsnahen Informationsportalen eine geschlossene, abgestimmte Lösung liefert. Um im obigen Bild zu bleiben: Einzelne Bauteile des Rades sind vorhanden, aber es existiert kein Bauplan.

Autor (Konzept)	Anforderungen: Kompletten Informationsbedarf						Problemorientierte Konzepte					Lösungsorientierte Konzepte							Konstruktiv			Objektbereich
	KIP-Info				Portal							Vernetz.										
	Produktionsumgebung	**Auftragsabwicklung**	**PPS**	**Kennzahlen**	Universell	Basis	**Arbeitsplatzspezifisch**	**Flankierende Information**	**Sporadische Notifikation**	**Bidirektionale Komm.**	**Personalisierung**	**Pfad**	**Navigation**	Workflow	**Präsentation**	**Interaktion**	**Integration**	**Technik**	**Analyse**	**Konzeption**	**Implementierung**	
Fokus auf problemorientierte Konzepte für produktionsnahes Informationsangebot																						
[Gaida00]	◑			◑	○					◐					○				◑			TQM für Gruppen
[Gelhaar95]	◑		◑		○					○	○		○						◑			Fertigungsmaschinen
[Gerber]	◐																		●			Pflichtenheft für QIS
[Kuba97]		○	○		○			○		◐					○				◑	◑		Kommunikation in Produktion
[Lang07]	●	○	◐	○	○			○		◐	○		○		◑					◑	○	Arbeitsanweisungen
[Linsenmaier99]	○													◑			◑			○	◑	Integrationsarchitektur mit BO
[Neuschw.03]	◑		○		◑					●	◑		○		◐				○	◑	○	Wissensdokumentation Fertigung
[Raupach08]	○														◐					◑		Ergonomie
[Slama04]	◐	○	◐	○	○			○		○	○		○		○				○	○		Montageunterlagen
[Sternemann99]	○												◑							◑	◑	Informations- und Wissensraum
[Wilhelm06]	○				◑					◑									◑	◑	◑	Content & ad-hoc-Info im QM
Fokus auf problemorientierte Konzepte für Portale, produktionsfern																						
[Collins01]						○													○			Anforderungskatalog für B2E
[Davydov01]					○	○								○			○	○		○		Übersicht Portaltechnik für B2B
[Fingar00]						○													○			Anforderungskatalog für B2B
[Galic05]						○								○			○	○		○		Plattform IBM Websphere
[Großmann05]					○	◐								○				○		○		Anforderungskatalog Portale
[Gurzki06]					◐	◐								○			○		○	○		Technischer Vertrieb
[Hinderer03]					○	○													○			Anwendungsbausteine Portale
[Puschmann03]						◐								◐			◐			○		Prozessportale
[Röhricht01]					○	○													○			B2B Anwendungsszenarien
[Sullivan04]													◑	○				○		○		Technikübersicht
Fokus auf lösungsorientierte Konzepte für Portale, produktionsfern																						
[Anastop.01]																		◑			◑	Performanz
[Brandon08]														◑			◑			◑		Modellbasierte Web-Entwicklung
[Ceri03]												○	◐	○	○	◐	◐	○	◐	◑	○	Webentwicklung mit WebML
[Finkelstein00]																	◑		◑	◑	◑	Reporting, Business Intelligence
[Goodyear00]						○									○		◐	◐		◐		Verteilte Daten & Applikationen
[Kappel04]													○	○	○	○	◐	◐	◑	◐		Übersicht Web-Engineering
[Menasce00]																	◑	◑	◑	◑		Performanz
[Mendes06]													○			○		◐	○			QS von Web-App. & Projekt
[Murugesan01]													○				○	○	◑	○		Thematisiert Web-Engineering
[Pressman09]													◐		○	◐	○		◑	◐		Isoliertes Web ohne ext. Komp.
[Rossi07]													○			○			◑	○		Formale Modelle für Web

● Breite Abdeckung, vertiefte Konzepte
◐ Breite Abdeckung, einfache Konzepte
◑ Teilweise Abdeckung, vertiefte Konzepte
○ Geringe Abdeckung, einfache Konzepte

Tabelle 2-3: Vergleich der Konzepte

Es fehlen daher umfassende Konzepte zur Realisierung effektiv nutzbarer, produktionsnaher Informationsportale bei kundenindividueller Produktion. Dabei verlangen zum einen

die Häufigkeit der notwendigen Erstellung von Portalseiten, zum anderen die hohen Gestaltungsansprüche an das Informationsangebot den Einsatz unterstützender Methoden. Somit ist ein Handlungsbedarf festzustellen hinsichtlich einer Methode, die die bestehenden Vorgehensweisen, Modelle und Werkzeuge ergänzt um eine Unterstützung durch umfassende Problem- und Lösungskonzepte.

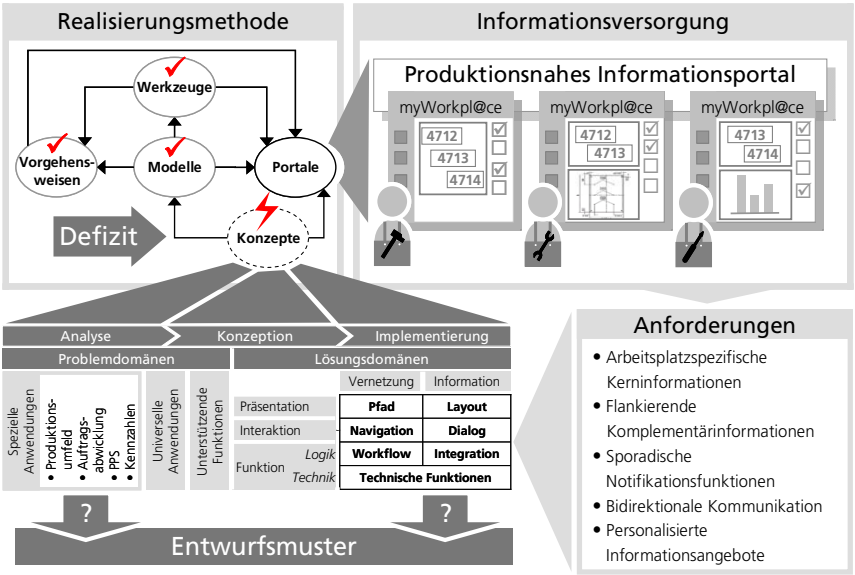


Abbildung 2-18: Defizit bei der Realisierung produktionsnaher Informationsportale

Ein vielversprechender Ansatz für eine konzeptionelle Unterstützung sind Entwurfsmuster. Sie sind weitgehend unabhängig von Systemplattformen und werden so der Heterogenität der betrieblichen Informationsarchitekturen gerecht. Sie ergänzen die Modelle, Werkzeuge und Vorgehensweisen des Web-Engineering und unterstützen Web-Entwickler mit erprobten und passenden Lösungskonzepten [Pressmann09, S. 327f; Kamthan08, S. 116]. Ihr Einsatz folgt dem Gedanken der Prozessqualität und verankert im Entwicklungsprozess die Fähigkeit, qualitativ hochwertige Web-Applikationen zu erzeugen [Kamthan08, S. 100].

Im weiteren Verlauf der vorliegenden Arbeit wird eine neue Methode auf Basis von Entwurfsmustern entwickelt, die den Aufbau qualitativ hochwertiger und bedarfsgerechter KIP-Infosets als Medien für eine effiziente Informationsversorgung unterstützt. Zuvor werden im nächsten Kapitel die grundlegenden Begriffe von Entwurfsmustern vorgestellt, bekannte Entwurfsmustersprachen mit Bezug zur Domäne aufgezeigt und ihre Eignung für eine konzeptionelle Unterstützung zum Aufbau produktionsnaher Informationsportale diskutiert.

3 Grundlagen der Entwurfsmuster

Seit Mitte der 90er Jahre wird insbesondere in der objektorientierten Softwaretechnik der Einsatz von Entwurfsmustern zur Unterstützung der Softwareentwicklung intensiv diskutiert. Zwischenzeitlich gelten sie als probates Mittel, um – plakativ formuliert – schneller zu besseren Ergebnissen zu gelangen [Hagen05, S. 10].

3.1 Grundlegende Begriffe zu Entwurfsmustern

3.1.1 Definition und Wesen von Entwurfsmustern

Entwurfsmuster sind informell beschriebene, erprobte Konzepte für wiederkehrende Entwurfsprobleme.

Die Idee, Lösungskonzepte für Entwurfsaufgaben in Form von Mustern zu explizieren, geht auf Christopher Alexander zurück, der eine Mustersprache für die Gebäudeplanung formulierte. Alle bekannten Muster beziehen sich auf seine Definition: *„Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem…"* [Alexander77, S. 10]. Alexander betont, dass ein Muster ein in ähnlicher Weise wiederkehrendes Problem adressiert und hierfür eine generalisierte Lösung aufzeigt. Der generalistische Anspruch unterscheidet Muster von Fallbeispielen [Hahsler01, S. 23]. Typisch für Muster ist ihre gestaltgebende Natur. Sie zeigen zu einer Entwurfsaufgabe die Form einer unmittelbar umsetzbaren Lösung. Beispielhaft zeigt Abbildung 3-1 Ausschnitte des Musters CORNER DOORS, das Positionierungslösungen für Türen in Wohnräumen enthält.

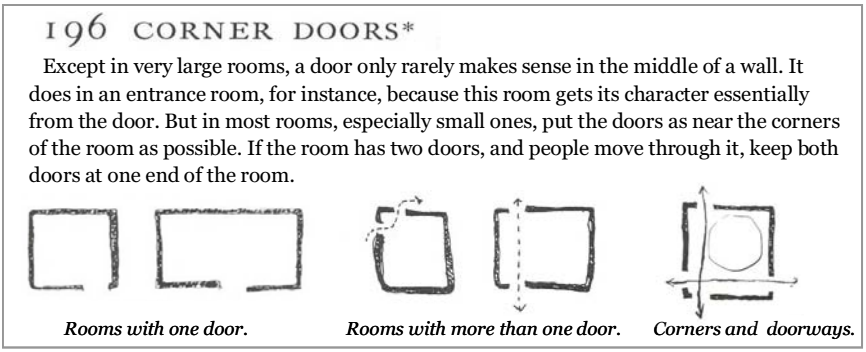


Abbildung 3-1: Musterbeispiel aus der Architektur. Quelle: [Alexander77, S. 904]

Mit den größten Erfolg erlangte Alexanders Idee im Bereich der Wiederverwendung von Lösungsstrukturen bei der Softwareentwicklung. Softwareentwurfsmuster liefern Vorlagen für Form und Inhalt zu erstellender Programme [Klose02, S. 107]. Sie zeigen bewährte Lösungen für Entwurfsprobleme, indem sie Komponenten, deren Verantwortlichkeiten, Beziehungen und Zusammenwirken darstellen [Buschmann07, S. 8]. Dabei ist die Bewährungsprobe immer zu sehen im Hinblick auf die Entwurfsaufgaben vorgegebenen, typischerweise konfligierenden Zielstellungen: Performance gegen Kosten, Einfachheit gegen Wartbarkeit etc. [Keller99]. Innerhalb dieses Spannungsfelds der Ziele ist eine Lösung zu

finden, s. Abbildung 3-2. Entwurfsmuster vermitteln, welche Ziele mehr, welche weniger befriedigt werden, erklären das Spannungsfeld der Ziele und tragen so zu einem tieferen Verständnis der Entwurfsaufgabe und der Lösung bei [Gabriel07].

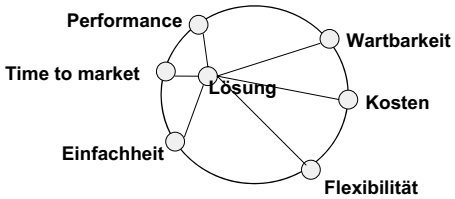


Abbildung 3-2: Balance von Zielkonflikten durch Musterlösungen. Quelle: [Keller99]

Entwurfsmusterbeschreibungen erfolgen in Prosa. An ihre Ausarbeitung bestehen hohe Ansprüche. Eine sorgfältige Formulierung und ausführliche Dokumentation gelten als ebenso substanzielle Bestandteile eines Musters, wie die Lösung selbst: *„…good practice poorly communicated ist the same as no practice at all."* [Buschmann07, S. 92]. Dabei tragen standardisierte Gliederungen zur Lesbarkeit, Erlernbarkeit und Vergleichbarkeit bei [Hahsler01, S. 29]. Eine Musterbeschreibung sollte aus folgenden Teilen bestehen [Coplien96, S. 7ff; Fowler97, S. 6f]:

- **Name** – jedes Muster sollte über einen aussagekräftigen, eindeutigen Namen verfügen, der das Vokabular der Entwickler bereichert und so die effiziente Kommunikation unterstützt.
- **Problem und Kontext** – explizite Beschreibung der Entwurfsaufgabe und der Vorbedingungen, unter denen die Lösung valide ist. Die Vorbedingungen können sich aus Problemumfang, zuvor angewandten Entwurfsmustern etc. ergeben. Die Kontextbeschreibung sollte alle Einflussgrößen auf die Validität der Musterlösung aufzeigen.
- **Lösung** – präzise und detaillierte Beschreibung der Lösung, die durch einen Entwickler direkt umsetzbar ist.
- **Entwurfsziele** – in dem Kontext typischerweise auftretende Entwurfsziele, deren Kriterien und die durch das Entwurfsmuster erzielte Balance der bestehenden Zielkonflikte. In Entwurfsmustern werden die auf eine Lösung einwirkenden Zielsetzungen üblicherweise als „Kräfte" beziehungsweise „Forces" bezeichnet.
- **Konsequenzen** – ergeben sich aus der Anwendung eines Musters und zeigen resultierende Entwurfsaufgaben und Kontexte, die Anlass zur Anwendung weiterer Entwurfsmuster geben.

Ein Beispiel für eine Entwurfsmusterbeschreibung in obiger Form findet sich in Anhang 9.3. Zusammenfassend lässt sich die Aufgabe von Entwurfsmustern charakterisieren, einem Entwickler bewährte Lösungen aufzuzeigen und darzulegen, wann, wie und warum die Lösungen anwendbar sind [vanWelie00, S. 1]. Sie müssen dafür sorgfältig und ausführlich dargestellt sein und ihre Wirksamkeit anhand mehrerer Implementierungen belegen - Entwurfsmuster werden nicht entwickelt, sondern gefunden. Mittlerweile gibt es viele Arten von Entwurfsmustern in der Softwareentwicklung.

- **Implementierungsmuster** nehmen unmittelbaren Einfluss auf die Gestalt von Softwareartefakten und zeigen, wie diese realisiert werden. Sie können auf unterschiedli-

chen Abstraktionsebenen vorliegen und sowohl Architekturen strukturieren als auch Detaillösungen aufzeigen [Porter05, S. 232; Buschmann98, S. 12, Appleton98, S. 4].

- **Protopatterns** sind Vorschläge für Entwurfsmuster, die noch nicht vollständig ausformuliert, deren essenzielles Lösungsprinzip noch nicht exakt erkannt oder deren Wirksamkeit nur in Einzelfällen belegt ist [Buschmann07, S. 395]. Zwar gibt es keine Regel für die erforderliche Anzahl der Nachweise. Ihre Wirksamkeit muss aber in jedem Fall durch mehrere erfolgreiche und praxisrelevante Implementierungen belegt sein [Cunningham04].
- **Antipatterns** beschreiben vermeintlich gute Lösungen, die aber mehr Probleme schaffen, als sie lösen [Cunningham06]. Antipatterns beruhen auf erfolglosen Softwareentwicklungsprojekten und zeigen, woran diese Projekte scheiterten [McCormick98]. ARCHITECTS DONT CODE ist ein Beispiel für ein Antipattern [Cunningham08]. Es tritt auf, wenn hoch qualifizierte Experten Systemarchitekturen entwerfen und verantworten, wegen des Zeit- und Kostendrucks aber weder an der Implementierung beteiligt sind noch in den letzten Jahren Programmiertätigkeiten ausführten. Häufig mündet dies in Entwürfen, die sich als nicht realisierbar erweisen.
- **Analysemuster** zeigen wiederkehrende Strukturen bei der Analyse von betrieblichen Abläufen in Unternehmen [Hay96, S. 4; Fowler97, S. 8]. Sie beschreiben charakteristische fachliche Prozesse, identifizieren die daran beteiligten Akteure und Informationen und zeigen die essenziellen Beziehungen und Strukturen zwischen diesen.
- **Usability-Muster** unterstützen die Gestaltung von Benutzungsoberflächen, Navigationsstrukturen und Präsentationselementen [vanWelie00] oder beschreiben Interaktionsformen einzelner Personen oder Gruppen von Personen mittels informationstechnischen Anwendungen [Martin01]. Die Muster können abstrakter Natur sein [Tidwell99] oder konkrete und fokussierte Entwurfsaufgaben behandeln, z. B. Muster für Benutzungsoberflächen für Suchanfragen [Wellhausen05].

Muster existieren ebenso für administrative und nachgelagerte Aufgaben der Softwareentwicklung [Coplien95]. Zum Beispiel für bewährte Managementtechniken und Organisationsstrukturen zur Softwareentwicklung [Ambler98; Dikel01], für das Testen, die Inbetriebnahme und die Wartung von Software [Ambler99, S. 17ff] oder auch für die sogenannte „Refaktorierung“, d. h. die funktionstreue, interne Umgestaltung bestehender Software, mit dem Ziel nicht-funktionale Eigenschaften zu verbessern [Fowler99, S. 41].

3.1.2 Entwurfsmustersammlungen

Entwurfsmuster bestehen für eine Vielzahl von Entwurfsaufgaben der Softwareentwicklung und werden häufig in Sammlungen zusammengestellt. So führt z. B. der Pattern Almanach über 1000 Entwurfsmuster für verschiedene Domänen auf [Rising00].

3.1.2.1 Musterorganisation

Mustersammlungen sollten geeignet klassifiziert und organisiert sein, wobei umfangreiche Mustersammlungen zum Teil mehrere Ordnungskategorien überlagern.

Häufig erfolgt die Organisation einer Mustersammlung nach der **Intention und Aufgabenstellung** der enthaltenen Muster. Beispielsweise gliedert Tichy nach der abstrakten Aufgabenstellung in Kategorien wie Entkopplung, Zustandsmanipulation, Objektauswahl, virtuelle Maschinen etc. [Tichy 98]. Van Duyne et al. gliedern ihre Mustersprache zur Gestaltung von Webanwendungen in zwölf Problemkategorien, u. a. die Erstellung von Navi-

gationsstrukturen, die Gestaltung von effektiven Seitenlayouts und die Reduzierung von Ladezeiten für Webseiten. Die Gliederung einer Mustersammlung nach der Intention ermöglicht eine einsichtige und nachvollziehbare Zuordnung der Entwurfsmuster zu einer Ordnungskategorie. Sie erlaubt ein effizientes Auffinden anwendbarer Entwurfsmuster, ist allerdings durch den Autor der Entwurfsmustersprache subjektiv geprägt und nicht immer eindeutig [Buschmann07, S. 224].

Ebenfalls verbreitet ist die Organisation von Entwurfsmustersammlungen anhand des **Abstraktionsgrads**. Gebräuchlich ist die Unterscheidung in Architekturmuster, Designmuster und Quellcodemuster [Buschmann98, S. 364]. Problematisch bei der Klassifikation nach dem Abstraktionsgrad ist die mehr oder weniger willkürlich gewählte Trichotomie. Die Abstraktionsgrade von Entwurfsmustern sind nicht scharf abgegrenzt, ihre Zuordnung zu den Klassen ist nicht eindeutig [Coplien96, S. 25]. Zudem ist der Abstraktionsgrad eines Entwurfsmusters kein sinnvolles Auswahlkriterium bei der Suche nach einer Lösung für eine Entwurfsaufgabe.

Ein weiterer struktureller Ansatz für Mustersammlungen besteht in der situativen Gliederung nach den **Phasen im Softwareentwicklungsprozess**. Bender verwendet die Entwicklungsphasen Analyse, Entwurf und Implementierung zur Klassifikation von Entwurfsmustern [Bender99, S. 29].

3.1.2.2 Musterkomposition

Komplexe Aufgabenstellungen erfordern bis zu ihrer vollständigen Implementierung die Anwendung mehrerer Entwurfsmuster, die zu gesamthaften Lösungen kombiniert und überlagert werden und sich gegenseitig unterstützen oder verfeinern [Gamma96, S. 15]. **Musterbeziehungen** zeigen konzeptionelle Abhängigkeiten und strukturelle Verfeinerungen zwischen Mustern auf und leiten die Komposition von Entwurfsmustern [Porter05, S. 238]. Sie sind Teil der Musterbeschreibung und häufig in einem Abschnitt enthalten, der alternative oder weiterführende Muster benennt [Hagen05, S. 25]. Häufig werden die Beschreibungen durch semantische Beziehungsdiagramme ergänzt. Dabei werden Kombinationsmöglichkeiten von Mustern durch Pfeile dargestellt und der Wirkzusammenhang angedeutet, siehe beispielhaft Abbildung 3-3.

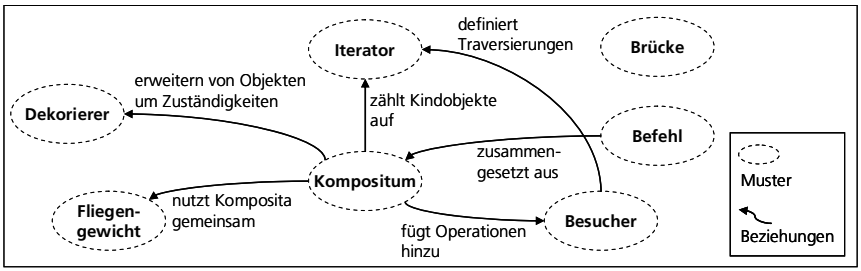


Abbildung 3-3: Ausschnitt Beziehungsdiagramm. Nach: [Gamma96, S. 16]

Einige Autoren verwenden stringentere Semantiken für Musterbeziehungen. Zimmer reduziert die Möglichkeiten auf drei Verwendungsbeziehungen: *X verwendet Y, X ist kombinierbar mit Y, X ist ähnlich zu Y* [Zimmer95, S. 6]. Noble geht weiter und unterscheidet primäre von sekundären Beziehungen. Primäre Beziehungen sind *Uses, Refines* und

Conflicts. Sekundäre Beziehungen können aus primären Beziehungen abgeleitet werden: *Used by, Refined by, Variants, Variant Uses, Similar, Combines, Require, Tiling* und *Sequence of Elaboration* [Noble98]. Dyson und Longshaw unterscheiden ebenfalls zwei Arten von Beziehungen [Dyson04, S. 52ff]. Direkte Beziehungen zwischen Mustern charakterisieren funktionale Abhängigkeiten der Form *Kombination, Alternative* und *Verwendung*. Indirekte Beziehungen ergeben sich aus Wechselwirkungen der Entwurfsmuster bezüglich nicht-funktionaler Merkmale, z. B. im Hinblick auf die Verfügbarkeit oder Skalierbarkeit einer Software.

3.1.2.3 Entwurfsmustersprachen

Mit zunehmender Kohäsion entstehen aus Mustersammlungen Musterkataloge, Mustersysteme und letztendlich Mustersprachen [Hagen05, S. 18; Klose02, S. 109ff].

Musterkataloge sind kategorisierte Mustersammlungen, wobei die Muster mit einem gemeinsamen Anwendungsbereich oder Einsatzzweck lose verbunden sein können, häufig aber auch losgelöst voneinander sind [Hagen05, S.18].

Mustersysteme enthalten kooperierende Muster zur Lösung einer gemeinsamen, komplexen Aufgabenstellung und sind charakterisiert durch [Buschmann98, S. 359ff]:
- eine einheitliche Beschreibungsform und geeignete Organisation der Muster,
- eine Beschreibung von Kompositionsbeziehungen zwischen den Mustern,
- die Beantwortung übergreifender Fragestellungen. Mustersysteme sind mehr als die bloße Summe der Muster [Meszaros97].

Mustersprachen bieten wie Mustersysteme Lösungen für komplexe Probleme, die einzelne Muster nicht lösen können. Darüber hinaus erheben sie den Anspruch der Vollständigkeit. Von Mustersprachen ist zu erwarten:
- die ausreichende Überdeckung der Probleme und Aspekte der Domäne mit Mustern. Alle typischerweise zu erwartenden Entwurfsaufgaben müssen durch die Mustersprache gelöst werden [Hagen05, S. 19; Buschmann98, S.360; Klose02, S.110].
- Ein hohes Maß an Kohäsion durch eine enge Kopplung der Entwurfsmuster. Die Mustersprache soll zusammenhängend sein und keine isolierten Muster enthalten [Buschmann07, S. 269 & 250; Salingaros00, S. 157].
- Eine stetige und erkennbare Progression bei der Lösungsfindung durch eine Vernetzung der Muster dergestalt, dass die Probleme und Aufgaben in der richtigen Reihenfolge angegangen und gelöst werden [Beck87; Coplien98, S. 2].
- Das Aufzeigen und Darlegen von allen Entwurfsaufgaben und Problemstellungen, die in dem Anwendungsgebiet bestehen [Beck87], sowie die Identifikation von Lücken in der Mustersammlung für eine Domäne [Zdun06, S. 988].

Die Begriffe Musterkatalog, Mustersystem und Mustersprache sind in der Literatur eingeführt, allerdings nicht verbindlich definiert und uneinheitlich verwendet [Hagen05, S. 19]. Auch wird gelegentlich die Begriffsbildung kritisiert, da weder Mustersysteme ein System im wissenschaftlichen Sinn darstellen, noch Mustersprachen dem Verständnis der Informatik von einer formalen Sprache entsprechen [Riehle96b, S. 10]. Riehle und Züllighoven verwenden daher den Begriff des Musterhandbuchs, wie er auch aus anderen Fachdisziplinen bekannt ist [Riehle96b, S. 10]. Jedoch hat sich diese Bezeichnung nicht etabliert.

3.1.2.4 Mustersequenzen

In jüngerer Zeit werden verstärkt Mustersequenzen diskutiert, z. B. in [Porter05; Zdun06; Henney06; Buschmann07; Siddle07]. Mustersequenzen beschreiben Reihenfolgen von Musteranwendungen in Softwareartefakten und sind somit temporale Strukturen auf Mustersammlungen [Porter05, S. 235ff]. Mustersequenzen und Musterbeziehungen stehen in gegenseitiger Wechselwirkung, sind aber nicht äquivalent. Eine Mustersequenz darf die Musterbeziehungen nicht verletzen. Muster zwischen denen Folgebeziehungen bestehen, müssen auch in Mustersequenzen aufeinanderfolgen, wenn auch nicht unmittelbar. Mustersequenzen können jedoch Muster sinnvoll anordnen, zwischen den keine Beziehungen bestehen [Siddle07, S. 17; Zdun06, S. 990]. Den Zusammenhang zwischen Musterbeziehungen und Mustersequenzen verdeutlicht eine in Anhang 9.4 beschriebene Mustersprache mit vier Entwurfsmustern. Abbildung 3-4 zeigt das Beziehungsdiagramm angereichert mit Sequenzierungsbedingungen [Henney06]. Offene Kreise stehen für optionale Folgen, gefüllte Kreise für notwendige Folgen und Kreisbögen zeigen ergänzende Folgen an, wobei eine oder mehrere der verbundenen Folgen benötigt werden. Aus dem Diagramm lassen sich mit den Musterbeziehungen verträgliche Mustersequenzen ableiten. In der Abbildung ist beispielsweise aufgeführt die leere Sequenz, die einem nicht passenden Musterkontext entspricht. Ebenfalls zulässig sind einfache Realisierungen mit wenigen Mustern oder umfangreichere Anwendungen mit mehreren Mustern. Auch die mehrfache Anwendung eines Musters ist in einer Mustersequenz darstellbar, wie im letzten Beispiel der zulässigen Sequenzen gezeigt.

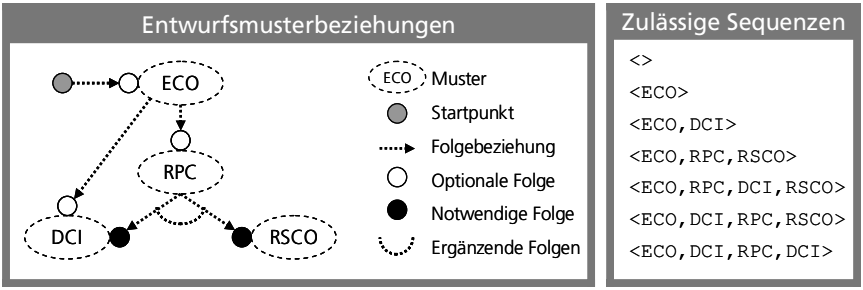


Abbildung 3-4: Vergleich von Musterbeziehungen und Mustersequenzen. Nach: [Henney06, S. D1-25]

3.1.2.5 Notation von Entwurfsmustersprachen

Entwurfsmuster und Mustersprachen sind ein eher informales Konzept, und nicht - wie in der Informatik üblich – durch Grammatiken formal definiert. [Hagen05, S. 19]. Von Beginn an finden sich Ansätze für formalisierte Notationen, so z. B. in [Riehle96a; Schütze99; Hagen05; Sand06]. Auch für Mustersequenzen finden sich Ansätze für formale Notationen. Porter et al. zeigen einen Ansatz für einen Formalismus zur Kombination von Mustersprachen mittels halbgeordneten Mustersequenzen, verbunden mit der Anmerkung, dass momentan keine Nutzenabschätzung für eine vollständige Formalisierung des Ansatzes möglich ist [Porter05, S. 246ff]. Henney beschreibt eine Grammatik für die zulässigen Sequenzen der oben vorgestellte Mustersprache ECO, s. Abbildung 3-5. Weitere Notationsmöglichkeiten zeigen Buschmann et al. [Buschmann07, S. 283f].

Alle Ansätze beschränken sich auf Einzelarbeiten. Verbreitete oder allgemein akzeptierte formale Notationen für Entwurfsmustersprachen finden sich nicht in der Literatur. Zurzeit ist die Formalisierung von Mustersprachen noch Gegenstand der Forschung in der Informatik. Auch finden sich in neueren Veröffentlichungen ambivalente Aussagen bezüglich des Sinns der vollständigen Formalisierung von Mustersprachen. So konstatiert Jenifer Tidwell: "*How not to write a Pattern Catalog: Treat it as an fun theoretical exercise*" [Tidwell07, S. 12]. Buschmann et al. bemerken *„Formalizing patterns is a popular sport, particularly in academia."* [Buschmann07, S. 85]. Als weiteres Argument gegen eine vollständige Formalisierung nennt Buschmann et al., dass der Einsatz von Entwurfsmustern teilweise auch situativ und subjektiv bedingt ist. Beispielsweise folgt die Auswahl zwischen konkurrierenden Entwurfsmustern durchaus den in Unternehmen eingeübten Programmierstilen. Derartige, durch die Unternehmenskultur vorgegebene Auswahlkriterien lassen sich nicht auf rationale Formalismen reduzieren [Buschmann07, S. 154]. Viele Sprachen verwenden daher eine Mischung aus intentionaler Gliederung, Prosa und lokalen Beziehungsdiagrammen [Buschmann07, S. 283f].

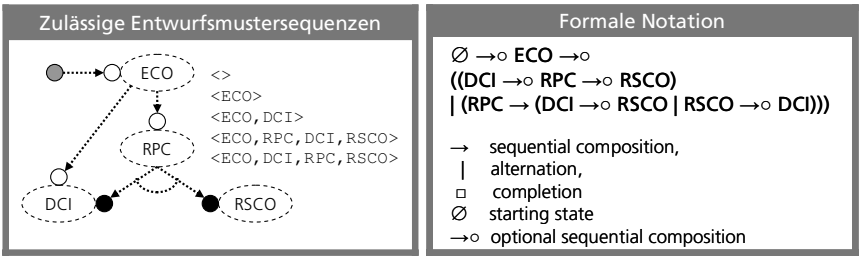


Abbildung 3-5: Beispiel einer formalen Notation von Mustersequenzen. Nach: [Henney06, S. D1-31]

3.2 Entwurfsmustersprachen für produktionsnahe Informationsportale

Entwurfsmuster unterstützen viele Gestaltungsaspekte, Phasen und Aufgaben der Softwareentwicklung. Dementsprechend gibt es keinen vollständigen Entwurfsmusterkatalog. Insbesondere Entwurfsmustersprachen sind häufig auf fokussierte Domänen ausgerichtet [Buschmann07, S. 267]. Im Weiteren werden potenzielle Entwurfsmustersprachen für produktionsnahe Informationsportale vorgestellt.

Analyse

Eine Reihe von problemorientierten Mustersprachen zeigen Entwurfsmuster für die Analyse des Gegenstandsbereichs. [Eriksson00] beschreibt Muster für die Modellierung von Geschäftsprozessen mittels UML. [Hay96] und [Fowler97] legen grundlegende und wieder verwendbare Datenstrukturen für Business Objekte dar. [Schümmer07] deckt den Bereich der Kommunikation und Interaktion mit unterstützenden Entwurfsmustern ab. Ein Beispiel für ein Analysemuster aus [Fowler97, S. 42] ist in Abbildung 3-6 gezeigt. Das Analysemuster OBSERVATION wird bei der Erfassung von Messwerten eingesetzt und ist eine vom beobachteten Objekt unabhängige Beschreibung einer variablen Anzahl von erfassten Werten. Folgende Informationselemente werden benötigt: Beobachtung protokolliert die Umstände der Erfassung des Werts und ordnet diese einem Beobachtungsgegenstand zu, Quantität bzw. Qualität repräsentieren den beobachteten Wert, Merkmalstyp ist das hinter der Beobachtung stehende Konzept der Messgröße. Das Muster eignet sich im

Produktionsbereich zum Beispiel für die Observation temperaturempfindlicher Artikel. Eine Beobachtung erfolgt mit dem Anlass „Wareneingang“ zum Zeitpunkt „12.10.2008, 22:34“. Gemessen wird der Merkmaltyp „Temperatur“ mit der Quantität „-18“ und der Messung „Grad Celsius“. Offensichtlich kann die Observation öfter durchgeführt werden, um z. B. eine unterbrechungsfreie Kühlkette während eines Transports zu belegen. Das Beispiel zeigt, wie Analysemuster die Detektion von Informationselementen in einem Anwendungsfall unterstützen.

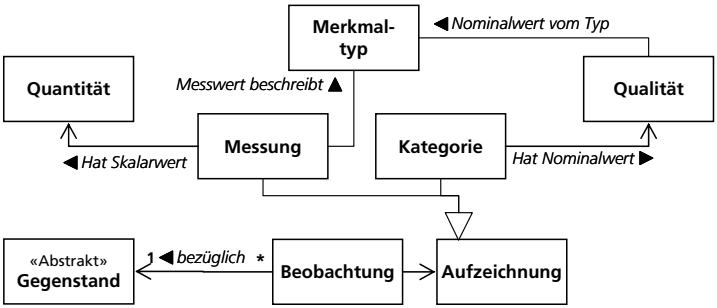


Abbildung 3-6: Analysemuster OBSERVATION. Nach: [Fowler97, S. 42]

Konzeption

Infolge der hohen Gestaltungsansprüche an ein produktionsnahes Informationsangebot verlangt eine konzeptionelle Unterstützung Usability-Muster zur Informationsaufbereitung und Navigation. Umfangreiche Sammlungen von Usability-Mustern mit konkreten Gestaltungsvorschlägen für das Web-Design und die visuelle Gestaltung existieren von [vanDuyne02; vanWelie07; Tidwell06]. Abstrakter sind die Mustersammlungen von [Griffiths98; Laakso03; Tidwell99] mit einem ausgeprägten Fokus auf das Interaktionsdesign und die Dialoggestaltung. Insgesamt weisen die Mustersammlungen starke Überschneidungen auf, auch mit weiteren unvollständigen Sammlungen von Usability-Mustern. Ein Beispiel für ein Usability-Muster ist TWO-PANEL-SELECTOR, auch bekannt unter der Bezeichnung OVERVIEW BY DETAIL. Das Muster eignet sich für die Präsentation einer Menge von Objekten mit jeweils detaillierten Zusatzinformationen. Zur Darstellung wird eine Übersichtsleiste für die Objekte mit einem Anzeigebereich für die Details kombiniert [Tidwell06]. Abbildung 3-11 illustriert die Anwendung des Musters.

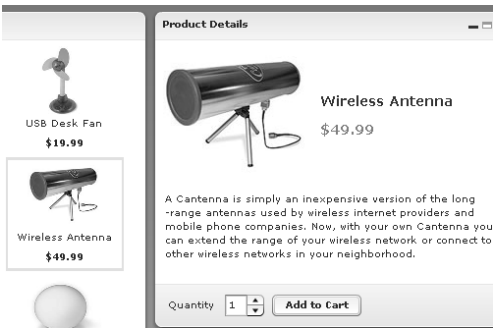


Abbildung 3-7: Usability-Muster OVERVIEW BY DETAIL. Quelle: [vanWelie07]

Implementierung
Für programmiertechnische Entwurfsaufgaben bei der Realisierung von Informationsportalen existieren ebenfalls Mustersammlungen. [Fowler02] bietet eine umfassende Sammlung von Implementierungsmustern für die Systemarchitektur von verteilten betrieblichen Web-Anwendungen. [Keller98] beschreibt eine Mustersprache für den Entwurf von Schnittstellen zwischen objektorientierten Sprachen, die überwiegend zur Realisierung von Web-Applikationen eingesetzt werden, und relationalen Datenbanken. Maßnahmen zur Berücksichtigung von Lastanforderungen und zur Anpassung der Leistungsstärke von Web-Applikationen zeigen [Anastopoulos01; Rüping07].

Generell ist zu konstatieren, dass viele Entwurfsmuster für die Entwicklung von Informationsportalen existieren. Ein Vergleich mit dem Domänenprofil offenbart die breite Abdeckung sowohl der Entwicklungsphasen als auch der Problem- und Lösungsdomänen, s. Abbildung 3-8.

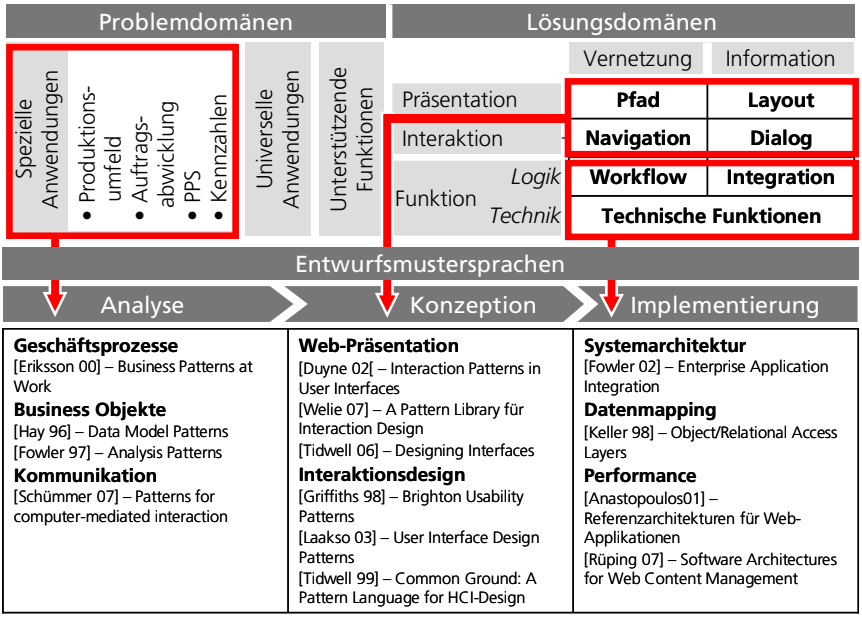


Abbildung 3-8: Übersicht der bekannten Mustersprachen für Informationsportale

3.3 Eignung des Lösungsansatzes

Nach der Schilderung des Stands der Technik bezüglich Entwurfsmuster wird abschließend ihre Wahl als Lösungsansatz für eine unterstützende Methode zum Aufbau von Informationsportalen begründet.

3.3.1 Alternative Formen konzeptioneller Unterstützung

Entwurfsmuster sind nicht die einzige Form einer konzeptionellen Unterstützung bei der Entwicklung von Informationsportalen. Durch die Wiederverwendung von Halbzeugen, die wie Entwurfsmuster vorhandene Erfahrungen in ihren Codefragmenten bergen, lassen sich ebenfalls Lösungskonzepte transportieren. Übliche Halbzeuge sind Komponentenbibliotheken, Frameworks oder Codegeneratoren [Schütze99, S. 30ff; Klose02, S. 118f; Hahsler01, S. 36]. Halbzeuge ermöglichen die direkte Wiederverwendung von Quellcode und sind deutlich konkreter als Entwurfsmuster. Gleichwohl sind sie proprietär und auf eine Systemplattform festgelegt [Eichinger04, S. 69]. Im Unterschied dazu sind Entwurfsmuster technologieneutral und in vielen Systemplattformen einsetzbar. In Anbetracht der Heterogenität der Informationsarchitekturen in Produktionsunternehmen wird in der vorliegenden Arbeit der technologieneutrale Lösungsansatz mit Entwurfsmustern der Verwendung von Halbzeugen vorgezogen.

3.3.2 Nutzen von Entwurfsmustern

Die Untersuchung des Stands der Forschung zeigte eine unvollständige konzeptionelle Unterstützung bei der Erstellung von produktionsnahen Informationsportalen, s. Kapitel 2.5. Entwurfsmuster greifen dieses Defizit auf und ergänzen die bestehenden Methoden zur Portalerstellung:

- **Vermittlung konstruktiven Lösungswissens***:* Entwurfsmuster zeigen, wie, wann und warum eine Lösung eingesetzt werden kann [Appleton98, S. 12], beschreiben Best Practices [Buschmann07, S. 8ff], dokumentieren und explizieren Designwissen von erfahrenen Softwareentwicklern und ermöglichen so den Aufbau von Bibliotheken mit Lösungswissen [Klose02, S. 5 & 114].
- **Dokumentation von Softwareartefakten**: Muster fördern und erleichtern das Verständnis der inneren Struktur eines Softwareartefakts und dokumentieren die zugrunde liegenden Entwurfsentscheidungen [Johnson92; Schütze99, S. 63; Klose02, S. 119; Hahsler01, S. 54].
- **Gemeinsames Entwicklungsvokabular***:* Entwurfsmuster geben Lösungskonzepten einen Namen und liefern eine Terminologie für eine effiziente Kommunikation zwischen Entwicklern [Gamma96, S. 432; Buschmann07, S. 8ff; Fowler97, S. 7]. Der Verzicht auf eine formale Modellierung und die informale, prosaische Beschreibungsform unterstützen interdisziplinäre Designprozesse und ermöglichen holistische Sichten auf Systeme, Entwurfsaufgaben und Lösungen [Klose02, S. 114; Coplien97, S. 38f].

Mit dem Einsatz von Entwurfsmustern verbindet sich eine Reihe von Nutzenerwartungen. Im Allgemeinen werden ihnen folgende positiven Auswirkungen zugesprochen:

- **Höhere Effizienz** des Entwicklungsprozesses durch Transfer von Erfahrungswissen, Vorgabe von Lösungsansätzen [Hahsler01, S. 55; Klose02, S. 114] und die Einschränkung des Entwurfsraums auf sinnvolle Teillösungen [Schütze99, S. 63].

- **Leichtere Wiederverwendung** von Modellen und Lösungskonzepten sowie darauf beruhender Software und Quellcode [Hahsler01, S. 54; Schütze99, S. 63]. Sogar zwischen Programmierparadigmen und –technologien ist eine Wiederverwendung möglich, die Halbzeuge nicht zulassen [Schmidt95, S. 5; Buschmann07, S. 11].
- **Steigerung der Softwarequalität** und der Benutzerakzeptanz als unmittelbare Konsequenz aus der Wiederverwendung von erprobten Lösungsansätzen.

Der Idee und dem Wesen von Entwurfsmustern immanent ist, dass ihr Nutzen nur empirisch ermittelt werden kann. Prechelt et al. haben diesbezügliche Studien durchgeführt [Prechelt02]. In den Versuchsreihen bekamen zwei Gruppen von insgesamt 96 Studenten mehrere Wartungsaufgaben gestellt, an Programmstücken mit 340 – 560 Codezeilen, die jeweils mit, beziehungsweise ohne Musterkommentare versehen waren. Es wurde nachgewiesen, dass die Probanden an Programmstücken mit Musterkommentaren mehr Aufgaben lösten und diese zudem schneller erreichten, s. Abbildung 3-5.

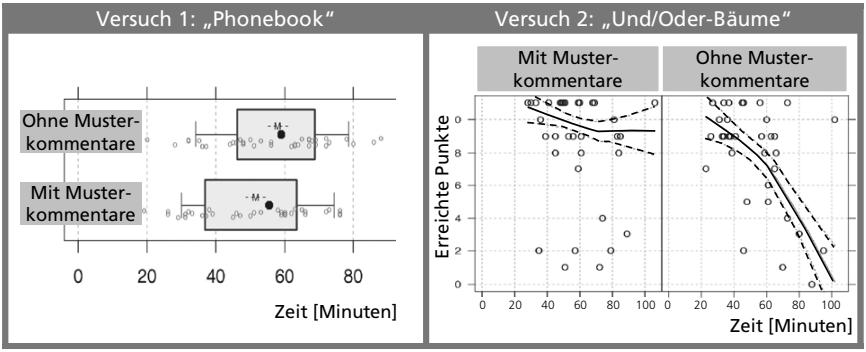


Abbildung 3-9: Vergleich von Lösungszeit und Lösungsqualität. Quelle [Prechelt02]

Der Nutzen von Entwurfsmustern lässt sich somit in abgesicherten, experimentellen Versuchsreihen durchaus nachweisen. Die Aussagen zum allgemeinen Nutzen sind unmittelbar übertragbar auf die Entwicklung von produktionsnahen Informationsportalen. Die höhere Effizienz im Entwicklungsprozess und die Wiederverwendung von Elementen von Portalseiten reduzieren den Aufwand für die Erstellung und kontinuierliche Pflege von Informationsportalen. Vor allem ist eine Steigerung der Softwarequalität zu erwarten im Hinblick auf die bedarfsgerechte, produktionskonforme Gestaltung der Informationsangebote.

3.3.3 Bewertung und Handlungsbedarf

Entwurfsmuster bieten einen vielversprechenden Ansatz für eine konzeptionelle Unterstützung bei der Erstellung von produktionsnahen Informationsportalen entsprechend den in Kapitel 2.5.1 aufgestellten Anforderungen: Sie sind konstruktiv und gestaltgebend, vermitteln Problem- und Lösungskonzepte und sind für den kompletten Informationsbedarf einsetzbar. Entwurfsmuster lösen Probleme, die durch die Modelle, Werkzeuge und Vorgehensweisen des Softwareengineerings nicht adressiert werden können [Buschmann07, S. 22; Coplien97]. Sie ergänzen bestehende Methoden zur Portalentwicklung, steigern die Softwarequalität und reduzieren den Entwicklungsaufwand.

Allerdings ist keine vollständige Mustersprache bekannt, die alle Phasen sowie Problem- und Lösungsdomänen abdeckt. Portalentwickler sind konfrontiert mit einer Fülle von Entwurfsmustern und Mustersprachen. Deren Anwendung verlangt eine umfangreiche Expertise sowohl hinsichtlich ihrer Problem- und Lösungsdomänen als auch im Anwendungsgebiet [Zdun06, S. 985f]. Tatsächlich fühlen sich Softwareentwickler häufig überfordert, Entwurfsmuster zu suchen und anzuwenden. So gaben in einer Studie zum Einsatz von Entwurfsmustern 85 % aller Befragten an, die Suche nach passenden Mustern problematisch zu empfinden. 50 % der Befragten bezeichneten die Suche gar als sehr oder ziemlich problematisch [Czichy01, S. 102]. Diese Aussagen gelten unverändert für die Web- und Portalentwicklung. Viele Web-Entwickler anerkennen Muster als nützlich, vermissen aber Verfahren und Werkzeuge für eine effiziente, systematische Auswahl von Entwurfsmustern [Pressmann09, S. 331], s. Abbildung 3-10.

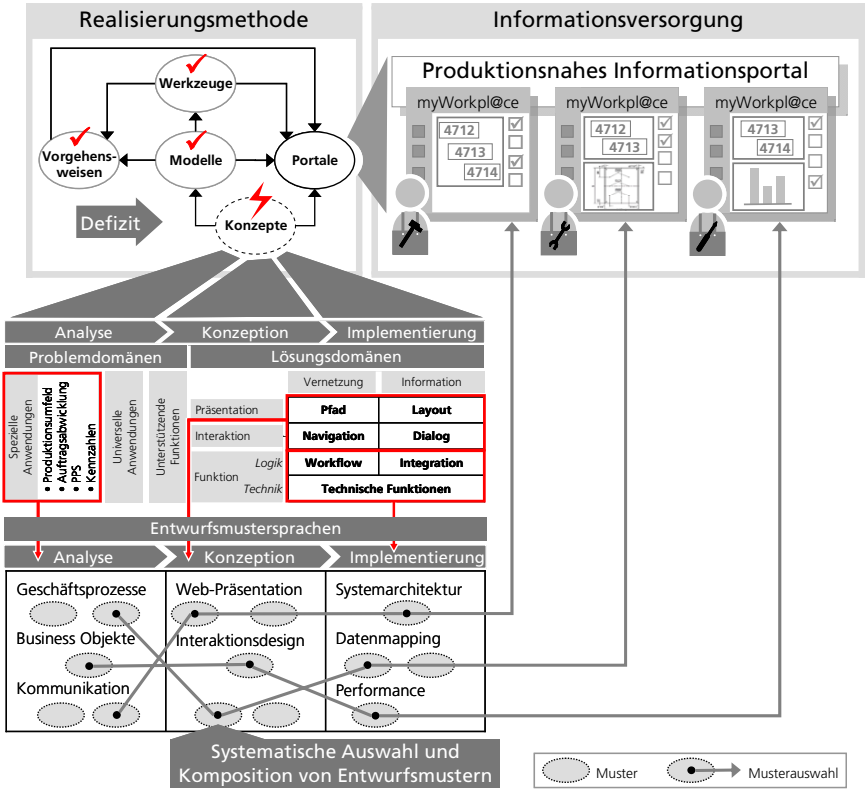


Abbildung 3-10: Sprachübergreifende Musterauswahl und Kombination

Neben der Auswahl bereitet auch die Komposition der Entwurfsmuster Probleme. Mustersprachen sind üblicherweise auf eng gefasste Implementierungsaspekte ausgerichtet [Porter05, S. 234]. Die Erstellung von Software erfordert dagegen die Lösung einer Vielzahl von Entwurfsfragen, die nicht zwingend voneinander abhängen, aber sich gegenseitig beeinflussen können. Es bedarf somit der Anwendung von Mustern aus unterschiedli-

chen Domänen und mehreren Mustersprachen [Porter05, S. 233f; Zdun06, S. 987]. Auch für die Realisierung produktionsnaher Informationsportale sind punktuelle Entwurfsmuster für nahezu jeden Aspekt des Domänenprofils bekannt. Jedoch ist nicht geklärt, wie die Muster aus verschiedenen Mustersprachen zusammenfinden.

Für die gesuchte Methode zur Unterstützung des Aufbaus von produktionsnahen Informationsportalen bedarf es einer systematischen Auswahl und Komposition von Entwurfsmustern aus vielen Mustersprachen mit Bezug zu den beteiligten Domänen und Entwicklungsphasen. Dazu wird im Anschluss eine neue, vollständige Mustersprache entwickelt, die alle benötigten Entwurfsmuster beinhaltet, in einer neuen Struktur verbindet und mit einer effizienten Auswahlmöglichkeit versieht. Der Aufbau der neuen Entwurfsmustersprache wird im folgenden Kapitel erläutert.

4 Entwurfsmustersprache für produktionsnahe Informationsportale

4.1 Übersicht der neuen Entwurfsmustersprache

4.1.1 Struktureller Aufbau

Ein wesentliches Merkmal der neuen Entwurfsmustersprache liegt in der Kombination bekannter Muster aus mehreren bestehenden Mustersprachen in einer neuen Struktur. Zur Unterscheidung werden die Neue als kombinierte Mustersprache und die Bestehenden als originäre Mustersprachen bezeichnet. Für eine kombinierte Mustersprache sind die Synergien der Muster der originären Domänen im neuen und gemeinsamen Kontext der produktionsnahen Informationsportale aufzuzeigen. Dazu bedarf es einer Musterauswahl, Mustervernetzung und Mustersequenzierung, s. Abbildung 4-1.

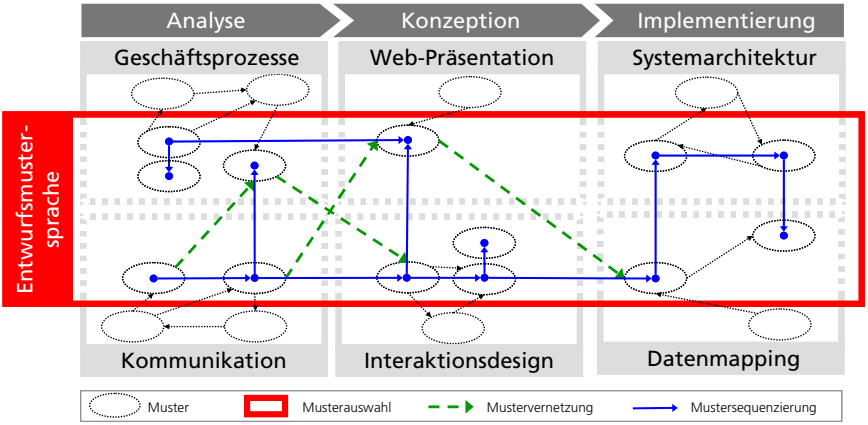


Abbildung 4-1: Vorgehensweise zur Entwicklung der neuen Mustersprache

Die **Musterauswahl** reduziert die Muster der originären Mustersprachen auf die Teilmengen der anwendbaren Muster. Beispielsweise enthalten Mustersprachen für Webanwendungen Muster für die Portalnavigation, die Personalisierung und die Realisierung von Webshops. Während die Portalnavigation und Personalisierung auch für produktionsnahe Informationsportale notwendig sind, ist die Implementierung von Webshops sinnlos, diesbezügliche Muster entfallen in der kombinierten Mustersprache. Die **Mustervernetzung** erfolgt unabhängig von den originären Mustersprachen, die Musterbeziehungen häufig in Form eines Abschnitts über die Konsequenzen der Musteranwendung als Teil der Musterbeschreibung enthalten. Die bestehenden Musterbeziehungen sind auf die originären Mustersprachen beschränkt. Neue Beziehungen zur Vernetzung der Muster verschiedener Sprachen sind daher explizit zu formulieren. Die **Mustersequenzierung** definiert eine Reihenfolge für die Anwendung der Muster und beschreibt einen systematischen Lösungsprozess. Mustersequenzen erfassen alle Muster der kombinierten Mustersprache und beteiligen insbesondere Muster an der Lösungsfindung, zwischen denen keine Folgebeziehungen bestehen.

Resultat der Musterauswahl ist ein Musterkatalog. Die darauf aufsetzende Mustervernetzung erzeugt ein Mustersystem. Die abschließende Mustersequenzierung führt zu einer Mustersprache für produktionsnahe Informationsportale. Im Anschluss ist der Aufbau der neuen Mustersprache in dieser Reihenfolge dargelegt. Den Ausführungen vorangestellt ist ein einleitendes Beispiel für den Einsatz von Entwurfsmustern bei der Entwicklung eines Informationsportals.

4.1.2 Fallbeispiel „Asynchrone Benachrichtigung"

Das nachfolgende Beispiel illustriert die Kombination von problem- und lösungsorientierten Entwurfsmustern zu einem Konzept für ein Informationsangebot, das die konstruktive Sichten abdeckt. Ausgangspunkt ist die Forderung nach einer bidirektionalen Kommunikationsfunktion für Werker zum Austausch produktionsnahen Know-hows bezüglich des Arbeitsumfelds. Die Kommunikationsfunktion soll in einem Portal die Mitarbeiter mit aktuellen Hinweisen zu neuen, verbesserten oder problematischen Betriebsmitteln, Arbeitsroutinen, Materialen etc. versorgen. Zur Umsetzung bietet sich die Form der asynchronen Benachrichtigung an [Hoffmann01, S. 266]. Dabei verfassen die Werker Hinweise in Form von Textnachrichten, die die Adressaten zu einem späteren Zeitpunkt bei Bedarf lesen.

Der Aufbau des Informationsangebots folgt mehreren Entwurfsmustern, s. Abbildung 4-2. Ausgangspunkt ist die Analyse des Informationsbedarfs. Das Muster THREADED DISCUSSION zeigt eine Form der asynchronen Kommunikation mittels kurzer Textnotizen, vergleichbar einem elektronischen Schwarzen Brett, das zusätzlich eine Antwortmöglichkeit bietet. Das Muster ist z. B. in Internetseiten häufig anzutreffenden Foren realisiert. SHARED ANNOTATION ist ein ergänzendes Muster, das vorschlägt, Anwendungsobjekte als Gliederungsregister für Dokumente und Notizen zu verwenden. Angewandt auf THREADED DISCUSSION führt dies zu vielen Diskussionsforen, jeweils eines je Anwendungsobjekt. Im Gegensatz dazu würde THREADED DISCUSSION ohne SHARED ANNOTATION bedeuten, nur ein Diskussionsforum einzurichten, das alle Textnotizen für alle Anwendungsobjekte umfasst. Im nächsten Schritt ist ein Informationsangebot zu konzipieren. SHARED ANNOTATION verweist auf das konzeptionelle Muster UNIFIED BROWSING HIERARCHY, das empfiehlt, die Navigation zu Webseiten, die Sachinformationen mit Kommunikationsfunktionen verbinden, nach den Sachinformationen zu gliedern. Folglich sind im Anwendungsbeispiel die Diskussionsforen in die Seiten der Anwendungsobjekte einzubetten. Jede Seite, die ein Anwendungsobjekt beschreibt, enthält eine Auflistung der aktuellen Themen des zugeordneten Diskussionsforums. Das vollständige Forum, mit ausführlicher Themenbeschreibung und allen Antworten, wird aus Platzgründen nicht innerhalb der Anwendungsseite aufgelistet. Vielmehr kann der Benutzer ein POP-UP-WINDOW öffnen, das alle Forumsbeiträge mit der Antwortstruktur in einer TREE TABLE anzeigt. Die Tabellenzeilen mit den Forumsbeiträgen sind als ENLARGED CLICK-AREAS ausgeführt, d. h., der gesamte Textblock ist zum Antworten antippbar gehalten. ENLARGED CLICK-AREA ist ein Muster, das die Bedienung von Touchscreens erheblich erleichtert, insbesondere im Produktionsumfeld. Abbildung 4-3 zeigt beispielhaft eine Anwendungsobjektseite mit eingebettetem Forum sowie einem POP-UP-WINDOW für das vollständige Forum.

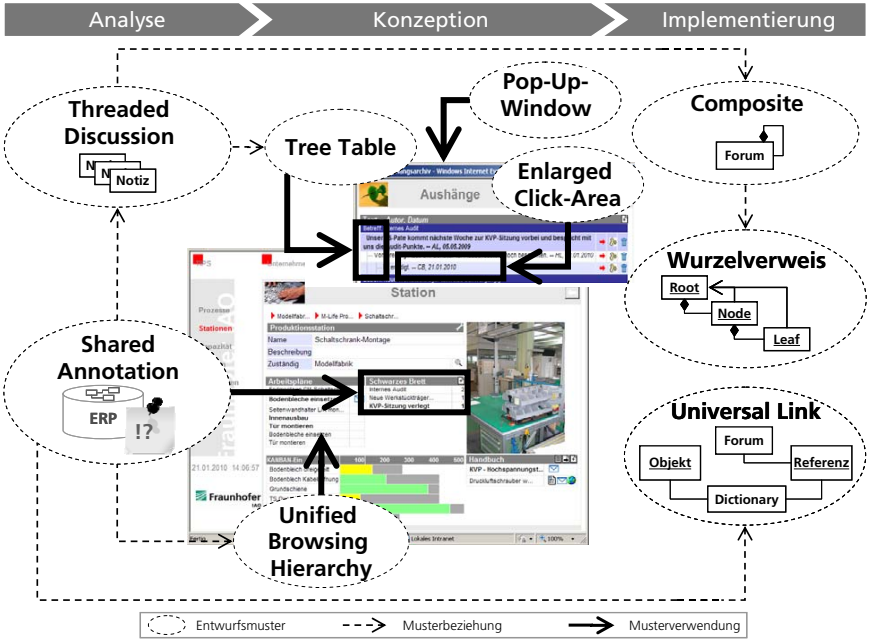


Abbildung 4-2: Entwurfsmuster für „Asynchrone Benachrichtigung"

Die oben genannten Anwendungsobjekte, z. B. Betriebsmittel, Routinen, Materialien sind typischerweise in ERP-Systemen verzeichnet. Programmtechnisch ist ein Diskussionsforum somit eine Baumstruktur für Textnotizen und Antworten, verlinkt mit beliebigen Anwendungsobjekten aus ERP-Systemen. Zur Implementierung des Anwendungsbeispiels kommen drei Muster zum Einsatz: COMPOSITE baut Hierarchiebäume durch Verweise auf das Vaterelement auf. COMPOSITE-Strukturen sind effizient zu pflegen und zu erweitern, in relationalen Datenbanken allerdings aufwendig auszulesen. Ein zusätzlicher WURZELVERWEIS bietet eine Möglichkeit, ein Forum trotzdem effizient mit einem Datenbankzugriff auszulesen, indem jeder Knoten einen Verweis auf die Baumwurzel führt. Zum Anheften der Foren an Anwendungsobjekte müssten vielfältige Verweise zu unterschiedlichen Tabellen in der Datenbank geführt werden. UNIVERSAL LINK ist ein Muster um Verweise aus einer Tabelle zu beliebigen anderen Datenbanktabellen mittels Indirektion über ein DATA DICTIONARY aufzubauen. Dadurch benötigen die Foren nur einen, einheitlichen Verweistyp für Verweise zu beliebigen Anwendungsobjekten.

Die gezeigten Entwurfsmuster nehmen maßgeblich Einfluss auf die Gestalt und Ausführung des Informationsangebots für eine bidirektionale Kommunikation mit einer asynchronen Benachrichtigungsfunktion. An dem Beispiel sind Analysemuster, Usability-Muster und Implementierungsmuster aus mehreren originären Mustersprachen beteiligt, die kooperieren und in Beziehung zueinanderstehen. Im Weiteren wird der vollständige Aufbau der neuen Entwurfsmustersprache ausgeführt, beginnend mit dem Entwurfsmusterkatalog.

Abbildung 4-3: Informationsangebot „Asynchrone Benachrichtigung“

4.2 Strukturierter Entwurfsmusterkatalog

4.2.1 Entwurfsmustersammlung

Grundlage des Entwurfsmusterkatalogs ist eine Sammlung von 203 Entwurfsmustern aus den in Kapitel 3.2 genannten originären Mustersprachen mit ausgeprägtem Domänenbezug. Für Detailprobleme wurden spezialisierte Mustersprachen recherchiert und aufgenommen. Insgesamt wurden 23 Originalquellen ausgewertet, s. Anhang 9.1.2. Durch die große Anzahl an Entwurfsmustern und deren hoher Überdeckung des Profils für Problem- und Lösungsdomänen, vgl. Abbildung 3-8, ist von einem ausreichenden Mustervorrat in der Entwurfsmustersammlung auszugehen. Jedes Muster ist mit Aufgabenstellung, Name, Lösungsprinzip und Quellenreferenz in der Sammlung verzeichnet. Sind zu einer Aufgabenstellung mehrere alternative oder ergänzende Entwurfsmuster bekannt, werden diese in einem Aufgabenbereich gebündelt. Auf eine ausführliche Beschreibung der Musterlösung, mit Begründung und Abwägung der Balance der Zielsetzungen, wurde in der Mustersammlung bewusst verzichtet. Diese elementaren Bestandteile eines Entwurfsmusters finden sich in den originären Mustersprachen. Der Fokus der Arbeit liegt auf einer übergreifenden Vernetzung der Muster zwischen den originären Mustersprachen. Eine vollständige Wiedergabe der Muster ist hierfür nicht notwendig. Tabelle 4-1 zeigt das Tabellenschema der Entwurfsmustersammlung anhand der Muster des Fallbeispiels. Die komplette Mustersammlung ist im Anhang 9.2 in Tabellenform aufgelistet.

	Muster	**Quelle**
Aufgabe	**Asynchrone Benachrichtigung**	
Name Lösungs-prinzip	**Threaded Discussion / Forum / Message Board** Schwarzes Brett, offen für alle Benutzer. In der "threaded" Version mit Antwortmöglichkeit als Diskussionforum.	[van Duyne 02] [Schümmer 07] [van Weliie 05]
Alternative	**FAQ / Frequently Asked Questions** Sammlung häufig wiederkehrender Fragen mit Lösung	[Schümmer 07] [van Duyne 02] [van Weliie 05]
Alternative	**Shared Annotation** Verwendung von Anwendungsobjekten als virtuelles Register für Dokumente, Notizen etc.	[Schümmer 07]
Aufgabe	**Ausgabeorganisation**	
	Pop-Up Windows / Pile of Working Surfaces Verteilung des Informationsangebots auf möglichst wenige Zusatzfenster für ergänzende Informationen.	[van Duyne 02] [Tidwell99]
Aufgabe	**Hierarchien darstellen**	
	Tree Table Hierarchiedarstellung in einer geordneten Tabelle mit Angabe der Stufe und Einrückung der Unterelemente.	[Tidwell06]
Aufgabe	**Aktionen auslösen**	
	Enlarged Clickarea Kleine Schaltflächen mit ebenfalls sensitiven, anklickbaren Bereichen umgeben, zur leichteren Bedienung.	[vanWelie05]
Aufgabe	**Datenmapping**	
	Composite Hierarchiebildung durch Verweise von den Kindern zum Vaterknoten.	[Gamma 96]

Tabelle 4-1: Ausschnitt Entwurfsmustersammlung

Die originären Mustersprachen sind nicht zwangsläufig disjunkt. Häufig enthalten sie Teilmengen der Muster anderer Sprachen, sowohl unter dem gleichen als auch unter einem eigenen Namen. Gleichartige Muster verschiedener Sprachen sind als ein Eintrag in der Mustersammlung berücksichtigt, unter Nennung aller Quellenreferenzen. Sind die Muster unter verschiedenen Namen bekannt, sind alle Namen aufgeführt. Gelegentlich sind Variationen von Grundmustern als eigenständige Muster in Mustersprachen ausgeführt. Der Übersichtlichkeit halber sind solche Variationen ebenfalls zusammengefasst und unter Nennung beider Namen in der Mustersammlung berücksichtigt, siehe oberstes Muster in Tabelle 4-1.

4.2.2 Ordnungsstruktur des Entwurfsmusterkatalogs

Zur effizienten Nutzung einer umfangreichen Mustersammlung bedarf es einer Ordnung und Klassifikation der enthaltenen Muster. Bekannte Ansätze kombinieren häufig zwei oder mehrere Kriterien für den Aufbau eines Ordnungssystems. Kombinationen aus situativen und intentionalen Gliederungen leiten die Anwender effizient zu den gesuchten Entwurfsmustern [Buschmann07, S. 225]. Dieser Ansatz wird daher auf die vorliegende Entwurfsmustersammlung übertragen. Der entstehende Entwurfsmusterkatalog sieht eine duale Kategorisierung vor, die eine rasche Lokalisierung der anwendbaren Muster ermöglicht und Lücken in der Mustersprache signalisiert. Abbildung 4-4 veranschaulicht die Ordnungsstruktur des Entwurfsmusterkatalogs.

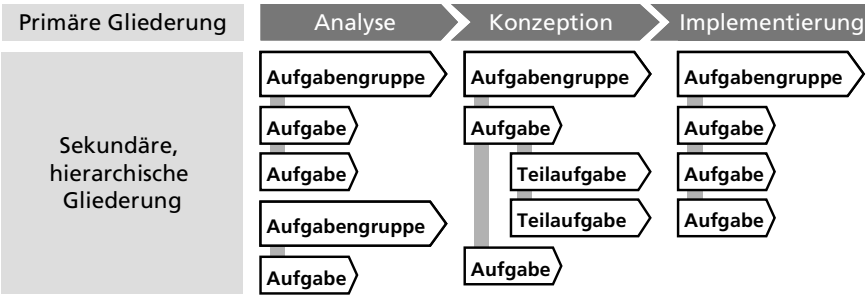


Abbildung 4-4: Duale Musterkategorisierung

4.2.2.1 Primäre Gliederung „Entwicklungsphasen und Sichten“

Das primäre Ordnungskriterium ist eine situative Gliederung nach den Phasen der Softwareentwicklung und unterscheidet in Analysemuster, Konzeptionsmuster und Implementierungsmuster. Die Phasen finden sich in den Mustersprachen wieder, die oft in einer korrespondieren Sicht geschrieben sind. Zum Teil drückt sich die Sichtweise einer Mustersprache bereits in der Benennung aus, z. B. „Analysis Patterns“ [Fowler97]. Die primäre Gliederung bündelt somit Muster mit vergleichbarer Sichtweise und Zweck, und korrespondiert zu den sichtspezifischen Anforderungen, Einflussgrößen und Gestaltungsfragen. Zudem sind bei der Internet- und Portalentwicklung agile und iterative Vorgehensweisen verbreitet [Dyson04, S. 287]. Der sukzessive Ausbau des Informationsangebots resultiert in einer ständigen Wiederholung des Entwicklungszyklus und der Schritte Analyse – Konzeption – Implementierung. Die primäre Gliederung des Musterkatalogs folgt diesem Zyklus und führt die Entwurfsmuster dem Entwickler passend zum situativen Unterstützungsbedarf zu.

4.2.2.2 Sekundäre Gliederung „Aufgaben“

Der Umfang der Mustersammlung verlangt eine sekundäre Ordnung des Musterkatalogs, die nach den in den jeweiligen Phasen zu erfüllenden Entwicklungsaufgaben erfolgt. Die sekundäre Gliederung ist hierarchisch aufgebaut. Aufgabengruppen werden in Aufgaben, Teilaufgaben etc. zerlegt. Die Gliederung muss dabei möglichst alle anfallenden Entwicklungsaufgaben aufzeigen, um den Entwurfsmusterkatalog anschießend zu einer Entwurfsmustersprache ausbauen zu können. Bei der Erhebung der Entwurfsaufgaben wurden sowohl die Referenzarchitekturen zur Portalentwicklung als auch die verwendeten originären Mustersprachen berücksichtigt. Weiterhin eingeflossen sind das Domänenprofil für produktionsnahe Informationsportale und die Erfahrungen aus der prototypischen Realisierung mehrerer produktionsnaher Informationsportale, s. Kapitel 6.2. Ein Vergleich mit den Vorgehensweisen des Web-Engineerings zeigt, dass die Aufgabenkategorien des Entwurfsmusterkatalogs detaillierter sind, und folglich ein umfassendes Bild der anfallenden Entwicklungsaufgaben bieten.

In Summe sieht das sekundäre Ordnungskriterium 79 Aufgaben zur Gliederung der 203 Entwurfsmuster vor. Die Verteilung der Muster auf Aufgaben ist sehr homogen. Jede Aufgabe ist durchschnittlich mit zwei bis fünf Entwurfsmustern abgedeckt. Die Granularität der Aufgaben kann somit als hinreichend für eine präzise und effiziente Lokalisierung von Entwurfsmustern angesehen werden. Die nachfolgenden, nach der primären Gliederung unterteilten Tabellen geben einen Überblick der Aufgabengruppen und deren Gliederung in Teilaufgaben.

Analyse		
Aufgabe	**Beschreibung**	**Teilaufgaben**
Geschäftsprozesse	Modellierung von Geschäftsprozessen zur Abgrenzung des Einsatzbereiches eines Informationsportals und zur Erhebung und Dokumentation des relevanten Einsatzkontexts, der Informationsflüsse und Kommunikationsbeziehungen.	
Business Objekte	Analyse der beteiligten Informationsinhalte und Datenquellen sowie ihrer benötigten Verarbeitungsfunktionen. Business Objekte abstrahieren in einer fachlich-logischen Schnittstelle von den realen Datenstrukturen der vorhandenen operativen Informationssysteme. Von ihrem Charakter her zu unterscheiden sind Stammdaten, Bewegungsdaten und temporale Daten, die Entwicklungen von Dateninhalten über die Zeit zeigen („Versionen“).	• Stammdaten • Bewegungsdaten • Temporale Daten
Kommunikation	Strukturierung von Kommunikationsbeziehungen. Aufzeigen von informationstechnisch unterstützten Kommunikationsformen, deren Einsatzmöglichkeiten und Randbedingungen. Produktionsnahe Einsatzmöglichkeiten bestehen vorwiegend zur asynchronen Benachrichtigung und für den Austausch von Erfahrungen und Know-how.	• Asynchrone Benachrichtigung • Erfahrungsaustausch
Personalisierung	Aufbau und Organisation von vorhandenen bzw. benötigten Merkmalen, Kriterien und Elementen für die Personalisierung von Informationsinhalten.	• Personalisierungsverfahren • Benutzerprofilierung • Contentprofilierung

Tabelle 4-2: Übersicht der Aufgabengliederung für die Analysemuster

Konzeption		
Aufgabe	**Beschreibung**	**Teilaufgaben**
Vernetzung		
Portal-navigation	Die Portalnavigation legt die Art und Weise fest, wie Benutzer die einzelnen Informationsinhalte erreichen können. Eine Navigationsstrategie beschreibt eine grundsätzliche Form der Navigation, z. B. Suchen oder Browsen. Ergänzend zum meistens obligatorischen Browsen sind eine Gliederungsstruktur und Navigationspfade festzulegen, über die Informationsinhalte erreicht werden.	• Navigationsstrategien festlegen • Inhalt gliedern • Navigationspfade festlegen
Navigations-präsentation	Die Navigationsmöglichkeiten sind dem Benutzer anzuzeigen. In umfangreichen und stark gegliederten Navigationsstrukturen benötigen die Benutzer Orientierungshilfen über die derzeitige Position im Navigationsgeflecht und über die Möglichkeiten zur relativen Navigation.	• Navigationswege darstellen • Orientierungshilfen vorsehen
Objekt-navigation	In produktionsnahen Portalen entstammt ein Großteil der Inhalte Datenbanken. Die übergeordnete Portalnavigation ist in vielen Fällen mit einer Objektnavigation zu kombinieren, die exaktes Lokalisieren einzelner Datensätze in umfangreichen Datentabellen gestattet. Bei der Objektnavigation sind Suchstrategien mit Filtern oder kontextorientierte Auswahlstrategien möglich.	• Suche • Auswahl
Information		
Portallayout	Für alle Portalseiten sind einheitliche Vorgaben für das Layout festzulegen. Berücksichtigt sind hier nur Entwurfsmuster für programmiertechnisch relevante Vorgaben. Dazu zählen Regeln für die Anordnung der Elemente und für das Verhalten der Elemente zur Anpassung an die verfügbare Bildschirmgröße.	• Raster festlegen • Größe festlegen
Seitendialog	Seitendialoge verbinden und organisieren Eingaben und Ausgaben von Informationsinhalten, die mehrere Bildschirmmasken oder Bildschirmseiten beanspruchen. Sie finden statt innerhalb eines abgeschlossenen Informationsangebots mit einer hohen fachlich-inhaltlichen Korrelation. Im Unterschied dazu erfordert eine Portalnavigation keine inhaltliche Korrelation der Angebote.	• Eingabeorganisation • Ausgabeorganisation - Heterogene Info. - Homogene Info.
Seitenlayout	Zur Präsentation von Informationsinhalten und von Eingabemöglichkeiten existiert eine Fülle von Mustern für die Gestaltung der Elemente und ihrer Funktionalität. Hierbei sind, wie bei der Gestaltung des Portallayouts, nicht die visuell-ästhetischen Gestaltungsaspekte angesprochen. Im Vordergrund der Betrachtungen stehen programmiertechnisch relevante Fragen, bezüglich des Verhaltens, der Vorbelegung, der Gruppierung, der Interaktivität etc. von Ein- und Ausgabeelementen. Angestrebt wird eine gute Benutzbarkeit und leichte Verständlichkeit der Inhalte.	• Eingabeelemente - Objektmethoden - Objektassoziationen - Objekthierarchie - Nominale Werte - Skalare Werte • Ausgabeelemente - Tabellen - Hierarchien - Grafiken - Beziehungen
Seiten-verarbeitung	Die Seitenverarbeitung umfasst alle Interaktionen, die zum bzw. unmittelbar nach dem Abschluss einer Eingabe zwischen dem Benutzer und dem informationstechnischen System stattfinden.	• Aktion auslösen • Eingabewerte kontrollieren • Rückmeldung geben

Tabelle 4-3: Übersicht der Aufgabengliederung für die Konzeptionsmuster

Implementierung		
Aufgabe	**Beschreibung**	**Teilaufgaben**
Portalarchitektur		
Programm-struktur	Zu Beginn der Implementierung eines Portals sind die Architektur und der grundlegende programmiertechnische Aufbau festzulegen. Das Gesamtsystem wird in den vielen Fällen nach dem MVC-Muster in eine Datenschicht, Präsentationsschicht und eine Interaktionssteuerung aufgeteilt. Für jede Schicht bestehen detaillierende Muster bezüglich des Funktionsprinzips.	• Gesamtsystem • Datenpersistierung • Präsentation • Interaktion
System-plattform	Portalarchitekturen orientieren sich an den verwendeten Systemplattformen, die häufig Entwurfsmuster für ihre korrekte Umsetzung kennen. In der vorliegenden Arbeit sind keine Entwurfsmuster berücksichtigt, um Abhängigkeiten von Plattformen zu vermeiden. Die Aufgabe ist als Platzhalter vorgesehen, für Muster im konkreten Einsatzfall.	
Anwendungsfunktionen		
Spezielle Anwendungen	Die Applikationslogik und die Verarbeitung der Daten bleiben üblicherweise den betrieblichen Informationssystemen der Back-End-Schicht vorbehalten. Portale übernehmen tendenziell Aufgaben zur Datenaufbereitung und Vernetzung. Spezielle Anwendungen benötigten vorwiegend Muster zur Erweiterung bestehender Applikationslogiken und zur Koordination von Nachrichtenflüssen in Anwendungen.	
Unterstützende Funktionen	Die unterstützenden Funktionen sind überwiegend durch die Portalplattform vorgegeben. In der Aufgabengruppe sind Muster über den sinnvollen Einsatz der Funktionen enthalten, die durch fallspezifische Regeln ergänzt werden können.	• Zugriffskontrolle • Portalmanagement • Benutzermanagement
Technische Funktionen		
Daten-mapping	Eine anspruchsvolle Aufgabe bei der Implementierung eines Portals besteht in der Abbildung der objektorientierten Business Objekte auf die gängigen relationalen Datenbanken in der Back-End-Schicht. Dabei sind die Abbildungen der in relationalen Datenmodellen unbekannten Vererbung und der Konzepte für komplexe Attribute, Hierarchien und Assoziationen zu klären. Typisch für Portale ist auch die Ergänzung der relationalen Datenelemente durch Objekte und Zusatzinformationen.	• Vererbung • Assoziationen • Komplexe Attribute • Hierarchie - Struktur - Effizientes Auslesen • Objektergänzung
Daten-handling	Alle Portalarchitekturen forcieren eine Kapselung des Datenzugriffs in einer eigenen Schicht, die ein explizites Datenhandling bietet. In jedem Fall zu klären sind die Fragen der Koordination nebenläufiger Datenzugriffe durch mehrere Anwender auf die gleichen Portalseiten und Datenobjekte, die Form der Zugriffsobjekte und des Datentransports zwischen objektorientierten und relationalen Programmteilen. Häufig empfiehlt sich die Einführung von Metadatenmodellen, zum Beispiel in Form von Data Dictionaries.	• Koordination Nebenläufigkeit • Zugriffsobjekt • Transporthandling • Metadaten
Interaktionsleistung	Funktional umfangreiche Portalanwendungen für eine größere Anzahl von Benutzern können erhebliche Systemlasten für die Portalserver erzeugen. In Abhängigkeit davon sind Fragen der Sitzungsverwaltung und des Aufrufhandlings zu beantworten. Für Portale mit einem sehr hohen Verkehrsaufkommen sind Maßnahmen zur Lastverteilung zu ergreifen.	• Sitzungsverwaltung • Aufrufhandling • Lastverteilung

Tabelle 4-4: Übersicht der Aufgabengliederung für die Implementierungsmuster

4.2.3 Status der Entwurfsmuster

Mit der situativen und intentionalen Gliederung des Musterkatalogs liegt eine ausreichend feine Kategorisierung vor, um Lücken im Musterkatalog zu identifizieren. Generell ist nicht zu erwarten, dass für alle Aufgaben Entwurfsmuster existieren. Im Einsatzfall müssen für eine Portalanwendung jedoch alle Entwicklungsaufgaben gelöst und in Softwareartefakte umgesetzt werden. Daher soll der Entwurfsmusterkatalog zumindest Regellösungen vorsehen, falls für Entwicklungsaufgaben keine Entwurfsmuster bekannt sind. Eine Regellösung zeigt eine beispielhafte Lösung, die für den Einsatzfall als vorteilhaft angesehen und zum Standard auserkoren wird. Regellösungen verzichten auf die strengen Kriterien, die ein Entwurfsmuster zu erfüllen hat: Sie müssen nicht mehrfach bewährt und nicht sorgfältig beschrieben sein. Zur Kennzeichnung wird ein Musterstatus eingeführt mit folgenden Unterscheidungen:

- **Muster** erfüllen alle Kriterien an Entwurfsmuster, sind üblicherweise veröffentlicht und mit einer Quellenangabe versehen;
- **Protopattern** sind allgemein erprobt und ausführlich dokumentiert, aber nicht in Musterform beschrieben;
- **Regellösungen** sind für den Einsatzfall beispielhaft gelöst, auf andere Informationsinhalte und Portalseiten übertragbar und eine einfache Dokumentation liegt vor;
- **Antipattern** beschreiben zu vermeidende Lösungen.

Damit ist es möglich, einen vollständigen Entwurfsmusterkatalog aufzustellen, der alle Entwicklungsaufgaben abdeckt[1]. Neben der Vollständigkeit des Entwurfsmusterkatalogs sind Regellösungen zudem sinnvoll für spezifische Lösungen von Entwurfsaufgaben, die auf konkrete Einsatzfälle ausgerichtet sind und die bekannten Entwurfsmuster ersetzen oder ergänzen. Die Dokumentation dieser spezifischen Regellösungen im Entwurfsmusterkatalog trägt zur Standardisierung der Portalanwendung bei.

Tabelle 4-5 zeigt beispielhaft die vollständige Struktur des Entwurfsmusterkatalogs. Die Tabelle verwendet den gleichen Ausschnitt der Entwurfssammlung wie Tabelle 4-1. Zusätzlich dargestellt sind die Ordnungskriterien „Phasen“ und „Aufgaben“, der Musterstatus und zwei Beispiele für Regellösungen. Zur leichteren Orientierung in dem umfangreichen Entwurfsmusterkatalog und zur eindeutigen Benennung sind den Aufgaben jeweils die Anfangsbuchstaben der Phasen und Aufgabengruppen vorangestellt. In der Erläuterungsspalte links sind die neuen Elemente fett hervorgehoben.

[1] Im vorliegenden Musterkatalog ist lediglich eine Entwurfsaufgabe, betreffend das effiziente Auslesen von Hierarchien, nicht durch mindestens ein Entwurfsmuster gelöst, sondern „nur“ durch zwei Regellösungen und ein Protopattern abgedeckt.

Phase	**Analyse**	
Aufgabe	**A.Kommunikation**	
Aufgabe	**A.K.Asynchrone Benachrichtigung**	
Muster / Quelle **Status**	**Threaded Discussion / Forum / Message Board** Schwarzes Brett, offen für alle Benutzer. In der "threaded" Version mit Antwortmöglichkeit als Diskussionforum.	[van Duyne 02] [Schümmer 07] [van Weliie 05] Muster
Muster / Quelle **Status**	**FAQ / Frequently Asked Questions** Sammlung häufig wiederkehrender Fragen mit Lösung	[Schümmer 07] etc. Muster
Muster / Quelle **Status**	**Shared Annotation** Verwendung von Anwendungsobjekten als virtuelles Register für Dokumente, Notizen etc.	[Schümmer 07] Muster
Phase	**Konzeption**	
Aufgabe	**K.Information**	
Aufgabe	**K.I.Seitendialog**	
Aufgabe	K.I.Seitendialog.Ausgabeorganisation	
Muster / Quelle **Status**	**Pop-Up Windows / Pile of Working Surfaces** Verteilung des Informationsangebots auf möglichst wenige Zusatzfenster für ergänzende Informationen.	[van Duyne 02] [Tidwell99] Muster
Aufgabe	**K.I.Seitenlayout**	
Aufgabe	K.I.Seitenlayout.Ausgabeelemente	
Aufgabe	K.I.Seitenlayout.Ausgabeelemente.Hierarchien darstellen	
Muster / Quelle **Status**	**Tree Table** Hierarchiedarstellung in einer geordneten Tabelle mit Angabe der Stufe und Einrückung der Unterelemente.	[Tidwell06] Muster
Aufgabe	**K.I.Seitenverarbeitung**	
Aufgabe	K.I.Seitenverarbeitung.Aktion auslösen	
Muster / Quelle **Status**	**Enlarged Clickarea** Kleine Schaltflächen mit ebenfalls sensitiven, anklickbaren Bereichen umgeben, zur leichteren Bedienung.	[vanWelie05] Pattern
Phase	**Implementierung**	
Aufgabe	**I.Technische Funktionen**	
Aufgabe	**I.T.Datenmapping**	
Aufgabe	I.T.Datenmapping.Hierarchie	
Aufgabe	I.T.Datenmapping.Hierarchie.Struktur	
Muster / Quelle **Status**	**Composite** Hierarchiebildung durch Verweise von den Kindern zum Vaterknoten.	[Gamma 96] Muster
Regel / Status	**Stückliste** „Flache“ Speicherung von Hierarchien durch Angabe der laufenden Nummer und Stufe eines Knotens	Regel
Aufgabe	I.T.Datenmapping.Hierarchie.Effizientes Auslesen	
Regel / Status	**Wurzelverweis** Jeder Knoten zeigt auf die Wurzel um alle Baumelemente in einer Datenbankanfrage auslesen zu können	Regel

Tabelle 4-5: Entwurfsmusterkatalog mit Gliederung, Musterstatus und Regellösungen

4.3 Kooperatives Entwurfsmustersystem

Im Musterkatalog ist eine systematische Lokalisierung von Mustern möglich, die anhand von Entwurfsaufgaben thematisch gruppiert sind, eine gemeinsame Entwurfsfrage adressieren und alternative Lösungen hierfür bieten. Es fehlt die Darstellung der Komposition der Muster zur gemeinsamen Lösung weitergehender Entwurfsfragen. Im folgenden Abschnitt wird der strukturierte Entwurfsmusterkatalog zu einem kooperativen Entwurfsmustersystem erweitert, durch die explizite Angabe von Musterbeziehungen.

4.3.1 Beziehungsarten

Beziehungen sind in Entwurfsmustersystemen definiert durch die Nennung der kooperierenden Muster, vgl. Kapitel 3.1.2.2. Die neue Entwurfsmustersprache kombiniert Muster aus unterschiedlichen Domänen. Zum Teil sind die Beziehungen in der kombinierten Entwurfsmustersprache schwächer ausgeprägt, als in den stärker fokussierten, originären Mustersprachen. Nicht immer identifizieren sie eindeutig weiterführende Muster. Öfter referenzieren sie neue Entwurfsaufgaben, die durch ein oder mehrere Muster zu lösen sind. Kooperationsbeziehungen bestehen in der kombinierten Entwurfsmustersprache somit auch zwischen Mustern und Aufgaben. Für die neue Entwurfsmustersprache werden insgesamt fünf Beziehungsarten eingeführt, die Muster mit Muster oder Muster mit Aufgaben untereinander verbinden. Tabelle 4-6 gibt eine Übersicht der Musterbeziehungen sowie der in Beziehungsdiagrammen verwendeten Symbole. Rautenförmige Symbole kennzeichnen Beziehungen zwischen Mustern einer Aufgabe, kreisförmige Symbole markieren Beziehungen zwischen Mustern verschiedener Aufgaben.

Symbol	Name	Beziehungsart
◇	**OR**	**Substitution** zwischen Mustern einer Aufgabe: Alternative Lösungen, Koexistenz zulässig
◇X	**XOR**	**Exklusion** zwischen Mustern einer Aufgabe: Alternative, sich gegenseitig ausschließende Lösungen
◇+	**AND**	**Supplement** von Mustern einer Aufgabe Ergänzende Lösungen in Verbindung mit einem Ausgangsmuster
⊃	**USE**	**Implikation** von Mustern mehrerer Aufgaben Konkretisierende Entwurfsaufgaben zu einem Ausgangsmuster
⊕	**ADD**	**Komplement** von Mustern mehrerer Aufgaben Begleitende Entwurfsaufgaben zu dem Ausgangsmuster

Tabelle 4-6: Übersicht und Symbole der Musterbeziehungen

Substitutionen bestehen, wenn Entwurfsfragen mehrere gute Lösungen kennen, die konfligierende Zielstellungen unterschiedlich beantworten. Bei gleichen Zielprioritäten kristallisiert sich keine beste Lösung heraus. Es bleibt die Entscheidung des „entweder – oder". Substituierende Muster beschreiben bewährte, alternative Lösungen zu einer Aufgabenstellung. Ausdrücklich gestattet ist eine Koexistenz der Lösungen. Für eine Aufgabe können eines oder mehrere der substituierenden Muster eingesetzt und realisiert werden. Im Fallbeispiel besteht zwischen den Mustern THREADED DISCUSSION und FAQ eine Substitutionsmöglichkeit. Jedes der Muster ist eine mögliche Lösung für die Aufgabe „Asynchrone Benachrichtigung", wobei die Koexistenz beider Lösungen möglich ist.

Exklusionen schließen, im Unterschied dazu, die Koexistenz der Lösungen aus. So lässt sich zum Beispiel in einer relationalen Datenbank eine Hierarchie mittels COMPOSITES und Verweisen auf das Vaterelement oder durch STÜCKLISTEN mit Angabe der laufenden Nummer und der Hierarchiestufe eines Datensatzes aufbauen. Beide Lösungen erfüllen unterschiedliche Zielsetzungen. Vaterverweise ermöglichen die einfache Realisierung von Pflegeoperationen auf der Hierarchie. Das Erweitern und Umhängen von ganzen Teilhierarchien ist problemlos durch Änderung eines Verweises möglich. Bei einer flachen Speicherung in Stücklisten verlangt das nachträgliche Einfügen von Hierarchieelementen eine Neuvergabe der laufenden Nummern und Stufen aller Elemente. Eine laufzeitintensive Datenbanktransaktion. Dafür lassen sich mit dieser Darstellungsform Hierarchien in einer einzigen Selektionsoperation vollständig und in der richtigen Reihenfolge auslesen. Für umfangreiche, häufig benötigte und selten geänderte Hierarchien äußerst vorteilhaft. Beide Muster anzuwenden, ergibt keinen Sinn. Die Hierarchien müssten auf beide Arten in der Datenbank angelegt und gepflegt werden. Die Lösung würde die Nachteile beider Muster vereinen.

Supplemente sind Beziehungen zu ergänzenden Mustern, die in Verbindung mit einem anderen Muster angewandt werden. Sie lösen die ursprüngliche Aufgabenstellung nicht und können das Ausgangsmuster somit nicht ersetzen. Supplemente verfeinern und ergänzen das Ursprungsmuster um Lösungen, die nicht zwingend notwendig für die Aufgabenstellung sind. Im Fallbeispiel der asynchronen Benachrichtigung sind THREADED DISCUSSION und SHARED ANNOTATION supplementäre Muster. SHARED ANNOTATION sieht eine Gliederung von THREADED DISCUSSION nach Anwendungsobjekten vor. Die Kombination beider Muster führt somit zu einem Forum je Anwendungsobjekt. SHARED ANNOTATION alleine bietet keine Lösung der Benachrichtigungsaufgabe, da die Form der Annotationen vage ist. Es kann THREADED DISCUSSION somit weder ersetzen noch wird es zur weiteren Realisierung eines Forums zwingend benötigt. Es erweitert das Muster THREADED DISCUSSION durch eine zusätzliche Gliederungsfunktion.

Implikationen führen zu abhängigen Entwurfsfragen, die sich in Konsequenz der Anwendung eines Musters ergeben und im Zuge der weiteren Realisierung beantwortet werden müssen. Sie verweisen auf anschließende Entwurfsaufgaben oder direkt auf weitere anzuwendende Muster. Beispielsweise zeigt das Muster THREADED DISCUSSION eine Form der asynchronen Benachrichtigung mit Antwortmöglichkeit. Das Muster beantwortet nicht die Frage der datentechnischen Abbildung und Speicherung von hierarchischen Diskussionsforen, sondern verweist auf eine diesbezügliche Aufgabenstellung. Für die Persistierung von Hierarchien in einer Datenbank sind mehrere Implementierungsmuster bekannt, deren Auswahl nach Aspekten der zu erwartenden Systemlast und Änderungshäufigkeit erfolgt. Eine direkte Beziehung zwischen dem Analysemuster und genau einem Implementierungsmuster lässt sich nicht zwingend argumentieren. Sehr wohl besteht aber eine Beziehung zwischen dem Analysemuster und der Implementierungsaufgabe zur Speicherung von Hierarchien.

Komplemente weisen auf ergänzende, komplementäre Entwurfsaufgaben, die häufig gemeinsam mit dem Ausgangsmuster auftreten. Komplemente konkretisieren das Ausgangsmuster nicht und tragen nicht zur weiteren Realisierung bei. Sie sind, im Gegensatz zu Implikationen, nicht notwendig für das Ausgangsmuster und unabhängig voneinander einsetzbar. Komplemente lösen zusätzliche Aufgabenstellungen, ergänzen sich und verstärken ihren Nutzen, Sie können auf Entwurfsaufgaben hinweisen oder direkt zu Mustern anderer Entwurfsaufgaben führen. Komplementäre Muster im Fallbeispiel sind SHA-

RED ANNOTATION und POP-UP WINDOWS. Ersteres bewirkt eine Gliederung der asynchronen Nachrichten nach den Anwendungsobjekten. POP-UP WINDOWS zeigt ergänzende Informationen in einem Zusatzfenster an. Nachrichten und Annotationen konkurrieren somit nicht ständig mit den Anwendungsobjekten um die knappe Darstellungsfläche auf dem Bildschirm und die Aufmerksamkeit des Benutzers.

Die Kooperationsbeziehungen der Mustersprache sind an den Mustern des Fallbeispiels in einem Beziehungsdiagramm dargestellt, s. Abbildung 4-5. Das Beziehungsdiagramm veranschaulicht, dass Implikationen und Komplemente nicht auf die Entwurfsaufgaben der gleichen Phase beschränkt sind. Sie können sowohl zwischen Aufgaben der gleichen Aufgabengruppe als auch zu Entwurfsaufgaben anderer Phasen bestehen. Und sie können Muster mit Mustern oder Muster und Entwurfsaufgaben verbinden.

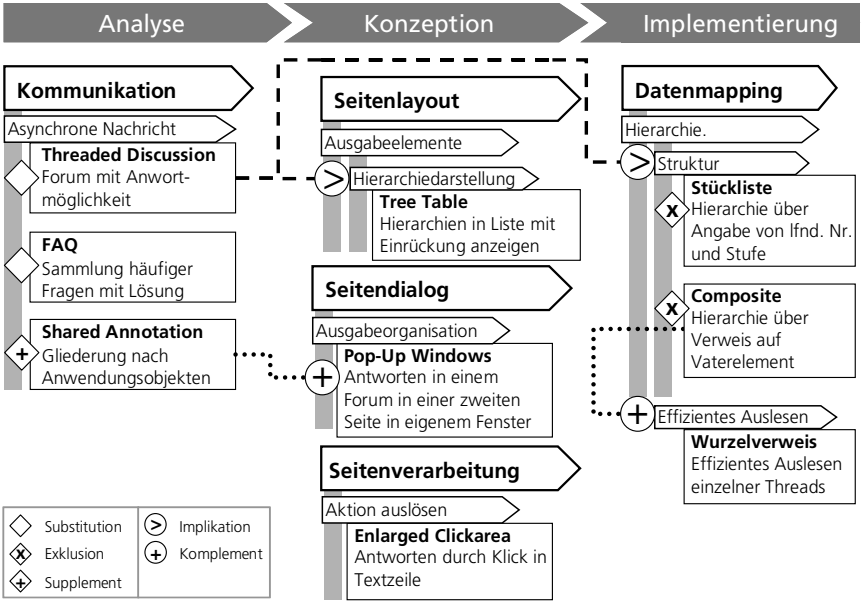


Abbildung 4-5: Kooperationsbeziehungen der Mustersprache

4.3.2 Notation von Beziehungen

Wie in Entwurfsmustersystemen üblich, wird eine semiformale, pragmatische Definition und Notation von Kooperationsbeziehungen verwendet [Buschmann07, S. 283f]. Im Entwurfsmusterkatalog sind Entwurfsmusterbeziehungen zwischen Mustern einer Aufgabe in einer separaten Spalte vermerkt. Beziehungen zwischen mehreren Aufgaben sind für jedes Muster in einem weiteren Abschnitt aufgeführt mit Angabe der Beziehungsart und des referenzierten Musters oder der Aufgabe. Diese Abschnitte werden im Weiteren auch als „Konsequenzen“ tituliert, in Anlehnung an eine in vielen Mustersprachen verwendete Terminologie, siehe z. B. [Porter05, S. 240; Zdun06, S. 984]. Tabelle 4-7 zeigt die Notation von Kooperationsbeziehungen an dem bekannten Ausschnitt der Entwurfssammlung,

mit den Mustern aus dem Fallbeispiel. Wiederum sind in der Erläuterungsspalte links die neuen Elemente betont dargestellt.

Die Semantik und Notation der Kooperationsbeziehungen ist semiformal gehalten, um Portalentwicklern eine möglichst operationale Mustersprache anzubieten, die einfach mittels weiterer Muster an heterogene Systemplattformen in den jeweiligen Unternehmen adaptierbar ist. Für den Umfang der hier aufgestellten Entwurfsmustersprache ist eine semiformale Notation völlig ausreichend und erleichtert ihren praktischen Einsatz. Daher wurde der Nutzbarkeit und Verständlichkeit der Vorzug eingeräumt, gegenüber einer formalen Präzision von Musterbeziehungen.

Phase		**Analyse**	
Aufgabe		**A.Kommunikation**	
Aufgabe		**A.K.Asynchrone Benachrichtigung**	
Substitution **Konsequenz**	**OR**	**Threaded Discussion / Forum / Message Board** Schwarzes Brett, offen für alle Benutzer. In der "threaded" Version mit Antwortmöglichkeit als Diskussionforum. **USE** K.I.Seitenlayout.Ausgabelemente.Hierarchien darstellen **USE** I.T.Datenmapping.Hierarchie.Struktur	[van Duyne 02] [Schümmer 07] [van Weliie 05] Muster
Substitution	**OR**	**FAQ / Frequently Asked Questions** Sammlung häufig wiederkehrender Fragen mit Lösung	[Schümmer 07] etc. Muster
Supplement **Konsequenz**	**AND**	**Shared Annotation** Verwendung von Anwendungsobjekten als virtuelles Register für Dokumente, Notizen etc. **ADD** K.V.Portalnavigation.Navigationsstrategie.UnifiedBrows. **USE** I.T.Datenmapping.Objektergänzung.UniversalDatalink	[Schümmer 07] Muster
Phase		**Konzeption**	
Aufgabe		**K.Information**	
Aufgabe		**K.I.Seitendialog**	
Aufgabe		K.I.Seitendialog.Ausgabeorganisation	
		Pop-Up Windows / Pile of Working Surfaces Verteilung des Informationsangebots auf möglichst wenige Zusatzfenster für ergänzende Informationen.	[van Duyne 02] [Tidwell99] Muster
Aufgabe		**K.I.Seitenlayout**	
Aufgabe		K.I.Seitenlayout.Ausgabeelemente	
Aufgabe		K.I.Seitenlayout.Ausgabeelemente.Hierarchien darstellen	
		Tree Table Hierarchiedarstellung in einer geordneten Tabelle mit Angabe der Stufe und Einrückung der Unterelemente.	[Tidwell06] Muster
Aufgabe		**K.I.Seitenverarbeitung**	
Aufgabe		K.I.Seitenverarbeitung.Aktion auslösen	
		Enlarged Clickarea Kleine Schaltflächen mit ebenfalls sensitiven, anklickbaren Bereichen umgeben, zur leichteren Bedienung.	[vanWelie05] Pattern
Phase		**Implementierung**	
Aufgabe		**I.Technische Funktionen**	
Aufgabe		**I.T.Datenmapping**	
Aufgabe		I.T.Datenmapping.Hierarchie	
Aufgabe		I.T.Datenmapping.Hierarchie.Struktur	
Exklusion **Konsequenz**	**XOR**	**Composite** Hierarchiebildung durch Verweise von den Kindern zum Vaterknoten. **ADD** I.T.Datenmapping.Hierarchie.Effizientes Auslesen	[Gamma 96] Muster
Exklusion	**XOR**	**Stückliste** „Flache" Speicherung von Hierarchien durch Angabe der laufenden Nummer und Stufe eines Knotens	Regel
Aufgabe		I.T.Datenmapping.Hierarchie.Effizientes Auslesen	
Muster / Quelle		**Wurzelverweis** Jeder Knoten zeigt auf die Wurzel um alle Baumelemente in einer Datenbankanfrage auslesen zu können	Regel

Tabelle 4-7: Mustersammlung mit Kategorisierung und Kooperationsbeziehungen

4.4 Kohärente und progressive Entwurfsmustersprache

Mustersprachen sollen vollständig, kohärent und progressiv sein, vgl. Kapitel 3.1.2.3. Sie dürfen nicht nur punktuelle Lösungen für einzelne Probleme bieten. Vielmehr müssen sie den gesamten Problemraum erschließen, partitionieren und gangbare Lösungswege vermitteln. Sie sollen zeigen, was für Entwurfsfragen zuerst zu klären sind, welche Entwurfsmuster bestehen und wohin gehend sich weitere Entwurfsfragen ergeben. Kooperationsbeziehungen zeigen zwar weitergehende Entwurfsaufgaben und implizieren eine Anwendungsreihenfolge der in Bezug stehenden Muster. Dies ist jedoch nicht hinreichend, die Entwurfsmustersammlung und den Problemraum vollständig zu erfassen. Kooperationsbeziehungen müssen nicht zwischen allen Mustern bestehen. Es können nur partielle Beziehungsgeflechte existieren, die keine Anwendungsreihenfolgen und Lösungswege erkennen lassen, die alle Muster beteiligen. Insbesondere bei Verwendung von Mustern aus mehreren originären Mustersprachen sind eher partielle Beziehungsgeflechte zwischen den Mustern zu erwarten. Das Beziehungsdiagramm für die Entwurfsmuster des Fallbeispiels in Abbildung 4-5 am Ende des letzten Abschnitts zeigt ein Beispiel hierfür. Aus den Kooperationsbeziehungen von THREADED DISCUSSION und SHARED ANNOTATION zur Aufgabe der Hierarchiedarstellung bzw. zum Muster POP-UP WINDOWS folgt, dass diese Entwurfsfragen später zu betrachten sind. Eine Reihenfolge zwischen den übergeordneten Aufgabengruppen Seitenlayout, Seitendialog und Seitenverarbeitung ist jedoch nicht definiert. Auch ist nicht klar, ob die isolierte Aufgabe Seitenverarbeitung überhaupt zu betrachten ist.

Es bedarf daher eines weiteren Elements in der neuen Entwurfsmustersprache, das die originären Entwurfsmuster vollständig in einem Lösungsweg erfasst. Als ein Schlüssel hierfür werden mittlerweile Mustersequenzen angesehen. Sie sind in der Lage Muster aus verschiedenen Quellen für neue Anwendungen zu vereinen und in einen gemeinsamen Lösungsweg zu integrieren. Voraussetzung hierfür ist, dass ein gemeinsamer Kontext für die originären Mustersprachen gegeben ist [Porter05, S. 240 & S. 247]. Für die Entwurfsmustersprache werden die Kooperationsbeziehungen daher um Sequenzen ergänzt. Analog zu den Kooperationsbeziehungen zwischen Mustern und Aufgaben, werden ebenfalls Aufgaben und Aufgabengruppen verwendet um Sequenzen zu bilden.

4.4.1 Aufgabensequenzen

Eine **Aufgabensequenz** wird definiert als eine Folge von Entwurfsaufgaben oder Aufgabengruppen aus dem Entwurfsmusterkatalog mit einer Start- und einer Endaufgabe. Sie umfasst alle zu erwartenden Entwurfsaufgaben bei der Realisierung eines abgeschlossenen Softwareartefakts. Bei der Startaufgabe beginnend werden alle Entwurfsaufgaben der Aufgabensequenz betrachtet. Für jede zutreffende Entwurfsaufgabe werden die enthaltenen Entwurfsmuster auf ihre Anwendbarkeit geprüft und ein oder mehrere Muster in die Konzeption des Softwareartefakts übernommen. Aufgabengruppen sind stellvertretend für ihre Unteraufgaben Teil der Aufgabensequenz. Alle Unteraufgaben sind bei der Konzeption zu betrachten, wobei die Reihenfolge der Betrachtung unerheblich ist. In Abhängigkeit eines zu realisierenden Informationsangebots können manche Entwurfsaufgaben einer Aufgabensequenz nicht zutreffend sein. Diese Entwurfsaufgaben werden übergangen, die zugeordneten Muster bleiben unberücksichtigt.

Beispielhaft ist in Abbildung 4-6 das Beziehungsdiagramm aus Abbildung 4-5 zusätzlich in einer Aufgabensequenz angeordnet. Die übergeordnete Aufgabensequenz wird für die

Erstellung von Portalseiten verwendet. Sie enthält als Teilsequenz die dargestellte Folge der Aufgabengruppen „Kommunikation“, „Seitendialog“, „Seitenlayout“ und „Seitenverarbeitung“. Über die Aufgabensequenz ist nun die Reihenfolge der Aufgabengruppen eindeutig festgelegt und die Entwurfsaufgabe „Seitenverarbeitung“ in den Lösungsweg integriert. Hinsichtlich des Fallbeispiels der „Asynchronen Benachrichtigung“ ist eine Reihe von Unteraufgaben nicht zutreffend und grau hinterlegt. So enthalten die Masken des Fallbeispiels keine Eingabefunktionen und Eingabeelemente, s. Kapitel 4.1.2. Die Unteraufgaben „Eingabeorganisation“, „Eingabeelemente“ und „Eingabewerte prüfen“ sind daher außer Acht zu lassen.

Neben der Aufgabensequenz sind bei der Musterauswahl die Kooperationsbeziehungen zu berücksichtigen. Sich gegenseitig ausschließende, exklusive Muster sind zu vermeiden, implizite Muster in jedem Fall aufzunehmen, supplementäre und komplementäre Muster bevorzugt anzuwenden. Im Beispiel der Aufgabensequenz aus Abbildung 4-6 impliziert THREADED DISCUSSION die Entwurfsaufgabe „Hierarchiedarstellung“ aus der Aufgabengruppe „Seitenlayout“. SHARED ANNOTATION legt die Verwendung von POP-UP WINDOWS nahe.

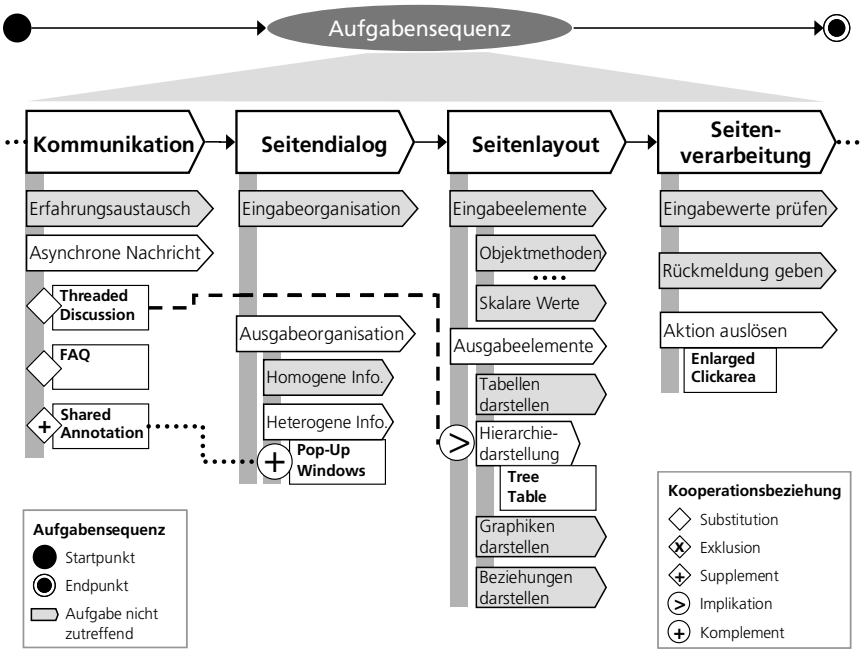


Abbildung 4-6: Beispielhafter Auszug einer Aufgabensequenz

Nach Abarbeitung einer Aufgabensequenz liegt ein abgeschlossenes Softwareartefakt als Resultat vor. Darunter sind nicht notwendig vollständige Portalanwendungen zu verstehen. Eigenständig realisierbare Softwarebausteine von Portalanwendungen genügen ebenfalls der Definition, z. B. einzelne Module oder Portalseiten. Tendenziell bestehen Portale aus Sammlungen von möglichst eigenständigen Informationsangeboten, die auf

jeweils wenige Portalseiten verteilt, über eindeutige Einstiegsseiten erreichbar und mit einer übersichtlichen Orientierungsstruktur verknüpft sind. Insbesondere produktionsnahe Informationsportale realisieren abgeschlossene Informationsangebote mit arbeitsplatzbezogenen und mitarbeiterspezifischen Filter- und Präsentationsfunktionen für vorhandene Informationen. Zwischen den Portalseiten bestehen nur geringe softwaretechnische Abhängigkeiten. Entsprechend der losen Kopplung von Portalseiten werden diese typischerweise unabhängig voneinander und mit inkrementellen Vorgehensweisen entwickelt [Dyson04, S. 287]. Auch eine inkrementelle Entwicklung hat sich an einer übergreifenden Portalarchitektur auszurichten, die in der Portal-Engine umzusetzen, zu konfigurieren und vorzuhalten ist.

Portalanwendungen entstehen somit in zwei typischen Aufgabensequenzen: dem einmaligen Erstellen einer Portal-Engine sowie dem mehrfachen und häufig wiederkehrenden Ergänzen von Portalseiten, s. Abbildung 4-7. Beide Aufgabensequenzen werden im Folgenden beschrieben.

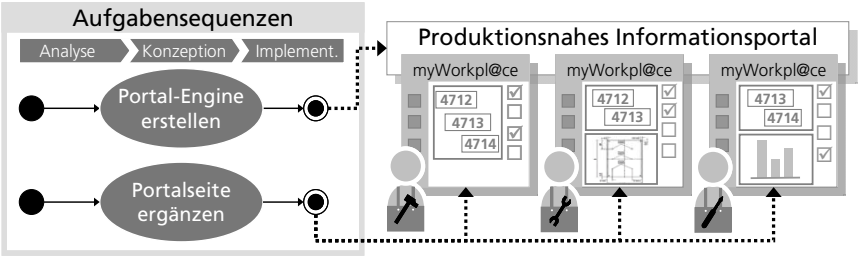


Abbildung 4-7: Aufgabensequenzen in der Entwurfsmustersprache

4.4.2 Aufgabensequenz „Portal-Engine erstellen“

Portal-Engines sind Bestandteile von Portalwerkzeugen mit zentralen Basisdiensten und technischen Funktionen zur Realisierung von Portalseiten für spezielle Anwendungen, s. Kapitel 2.4.2. Die Aufgabensequenz zum Erstellen einer Portal-Engine aus Tabelle 4-8 führt zu Mustern, die grundlegende Architekturfragen adressieren, das Portal im Ganzen und alle Portalseiten betreffen. Für jede Portalanwendung wird die Sequenz einmal abgearbeitet. Danach wird sie nur noch bei umfangreichen Änderungen („Relaunch“) benötigt, die eine Vielzahl von Portalseiten betreffen.

Aufgaben-sequenz	Erläuterung „Portal-Engine erstellen“
Analyse	
Geschäftsprozesse	Die Geschäftsprozesse liefern Hinweise für die für alle Portalseiten gemeinsame Portalnavigation. Dieser Schritt dient der Abgrenzung des Einsatzbereichs, ist Ausgangspunkt der gesamten Portalentwicklung und muss daher zu Beginn erfolgen.
Stammdaten	Eine Portalnavigation orientiert sich häufig an Begriffen, die typischerweise Stammdaten entsprechen: „Produkte, Material, Stationen“ etc. Die benötigten Stammdaten sollten daher im Zuge des Aufbaus der Portal-Engine identifiziert und im Gerüst der Portalnavigation berücksichtigt werden. Die Aufgabe schließt sich der Erhebung der Geschäftsprozesse an.
Personalisierung	Eine Personalisierung von Informationsinhalten sollte im gesamten Portal nach einem einheitlichen Schema erfolgen, und ist mit der Portal-Engine vorzugeben. Die Ermittlung des Personalisierungsschemas basiert auf den Stammdaten und erfolgt daher nach deren Erhebung.
Konzeption	
Portalnavigation	Die Portalnavigation umfasst alle Informationsinhalte, ist zentral festzulegen und der erste Schritt der Konzeption.
Navigations-präsentation	Die Präsentation der Portalnavigation muss für einen Benutzer stets einheitlich und somit erwartungskonform ausgeführt sein. Sie ist ebenfalls zentral vorzugeben. Die Navigationsstruktur muss für diesen Schritt bekannt sein.
Portallayout	Analog gilt für das Portallayout, dass ein identisches Schema für die Anordnung und Gestaltung vergleichbarer Elemente eingehalten und daher zentral vorgegeben wird. Das Portallayout berücksichtigt die Darstellungselemente für die Navigationspräsentation und ist daher im Anschluss sowie in enger Abstimmung festzulegen.
Implementierung	
Portalarchitektur	Eine Portalarchitektur legt die grundlegenden Programmstrukturen und Basisdienste fest, die von allen Portalseiten einzuhalten und zu verwenden sind. In ihr liegt der Schlüssel für die funktionalen Eigenschaften Wartbarkeit, Zuverlässigkeit und Erweiterbarkeit eines Portals. Sie kann nur zentral vorgegeben werden.
Unterstützende Funktionen	Unterstützende Funktionen sichern den Betrieb eines Portals. Sie sind transparent für einzelne Portalseiten, und somit in dieser Sequenz festzulegen. Der einzusetzende Umfang an unterstützenden Funktionen folgt der Portalarchitektur, bestimmt zum Teil den Bedarf an Datenhandling und ist somit zwischen diesen beiden Aufgaben wahrzunehmen.
Datenhandling	Das Datenhandling ist Teil der Basisdienste eines Portals. Es steht allen Portalseiten zur Verfügung und ist zentral als Teil der Portal-Engine zu realisieren.
Interaktions-leistung	Wie die unterstützenden Funktionen sind Fragen der Interaktionsleistung transparent für die Portalseiten und durch die Portal-Engine zentral zu beantworten. Maßgeblich für die Interaktionsleistung ist u. a. das Datenhandling, das daher zuvor geklärt sein muss.

Tabelle 4-8: Aufgabensequenz „Portal-Engine erstellen“

4.4.3 Aufgabensequenz „Portalseiten ergänzen“

Die Aufgabensequenz zum Erstellen, Ändern und Ergänzen von Portalseiten wird für jedes Informationsangebot ausgeführt. Die Entwurfsentscheidungen wirken nur auf eine

oder auf wenige Portalseiten, soweit diese zur Realisierung des Informationsangebots benötigt werden.

Aufgabensequenz	Erläuterung „Portalseiten ergänzen“
Analyse	
Bewegungsdaten	Das bestehende Datenmodell eines Portals mit den grundlegenden Stammdaten ist zur Realisierung einer Portalseite zu detaillieren. Die benötigten Informationsinhalte und Datenumfänge sind nur in Verbindung mit dem Informationsangebot einer Portalseite exakt zu ermitteln. Da produktionsnahe Informationsportale in der Abwicklung von Geschäftsprozessen eingesetzt werden, sind insbesondere die benötigten Bewegungsdaten zu analysieren und ggf. die Stammdaten zu erweitern. Dieser Schritt ist Ausgangspunkt der Sequenz.
Temporale Daten	Gleiches gilt für temporale Daten, sofern diese auftreten. Die Analyse der temporalen Daten kann mit oder im Anschluss der Analyse der Bewegungsdaten erfolgen.
Kommunikation	Kommunikationsfunktionen sind nach Bedarf gezielt in einzelne Informationsangebote zu integrieren. Die Analyse der Kommunikationsfunktionen kann ebenfalls mit oder im Anschluss der Datenanalyse erfolgen.
Konzeption	
Objektnavigation	Eine Objektnavigation schließt sich an die übergeordnete Portalnavigation an. Sie ist auf das Informationsangebot einer Portalseite abzustimmen und somit Teil der Portalseitensequenz. Insbesondere kontextorientierte Auswahlstrategien sind nur für einzelne Portalseiten definierbar. Die Auswahl der Objektnavigation ist die erste konzeptionelle Entwurfsaufgabe.
Seitendialog	Die „Mikronavigation“ innerhalb eines abgeschlossenen Informationsangebots kann nur für jede Portalseite individuell festgelegt werden. Die diesbezüglichen Entwurfsfragen sind für alle Portalseiten mit mehrseitigen Informationsinhalten zu lösen. Ein Seitendialog schließt sich an die Objektnavigation an, dem entsprechend ist die Aufgabe nach der Wahl der Objektnavigation wahrzunehmen.
Seitenlayout	Das Seitenlayout ist stets für jede Portalseite spezifisch zu gestalten. Die Vorgaben des Portallayouts sind zu beachten. Bei der Festlegung des Seitenlayouts gilt es, die Darstellungselemente für den Seitendialog zu berücksichtigen. Es ist daher im Anschluss und enger Abstimmung festzulegen.
Seitenverarbeitung	Die Seitenverarbeitung ist für Portalseiten mit Eingabefunktionen zu klären. Die hierfür notwendige Interaktion mit dem Benutzer kann nur individuell für jede Portalseite festegelegt werden. Um die Seitenverarbeitung zu bestimmen, müssen die im Seitenlayout festgelegten Ein- und Ausgabeelemente bekannt sein, da diese die Syntax der Eingabewerte und Rückmeldungsformate festlegen.
Implementierung	
Spezielle Anwendungen	Spezielle Anwendungen werden in einem Portal mittels einzelner Seiten realisiert, die diese Funktionen anbieten. Ein Ergebnis dieser Implementierungsaufgabe sind die verwendeten Datenstrukturen. Sie ist daher vor dem anschießenden Datenmapping wahrzunehmen.
Datenmapping	Im letzten Schritt der Aufgabensequenz für Portalseiten sind die erhobenen Datenstrukturen auf die relationalen Datenquellen abzubilden.

Tabelle 4-9: Aufgabensequenz „Portalseiten ergänzen“

4.4.4 Fallbeispiel mit Aufgabensequenz

Mit der Aufgabensequenz zum Ergänzen von Portalseiten lassen sich nun alle im Fallbeispiel verwendeten Entwurfsmuster systematisch herleiten, wie in Abbildung 4-8 dargestellt. Das Beziehungsdiagramm zeigt zudem ein Beispiel für eine Kooperationsbeziehung zwischen Aufgabensequenzen. SHARED ANNOTATION verweist auf das komplementäre Muster UNIFIED BROWSING HIERARCHY, das Teil der Aufgabensequenz „Portal Engine erstellen" ist. Der Einsatz einer UNIFIED BROWSING HIERARCHY innerhalb der Aufgabensequenz für eine Portalseite, ist somit nur möglich, falls dieses Muster bereits in der Konzeption der Portal-Engine vorgesehen ist.

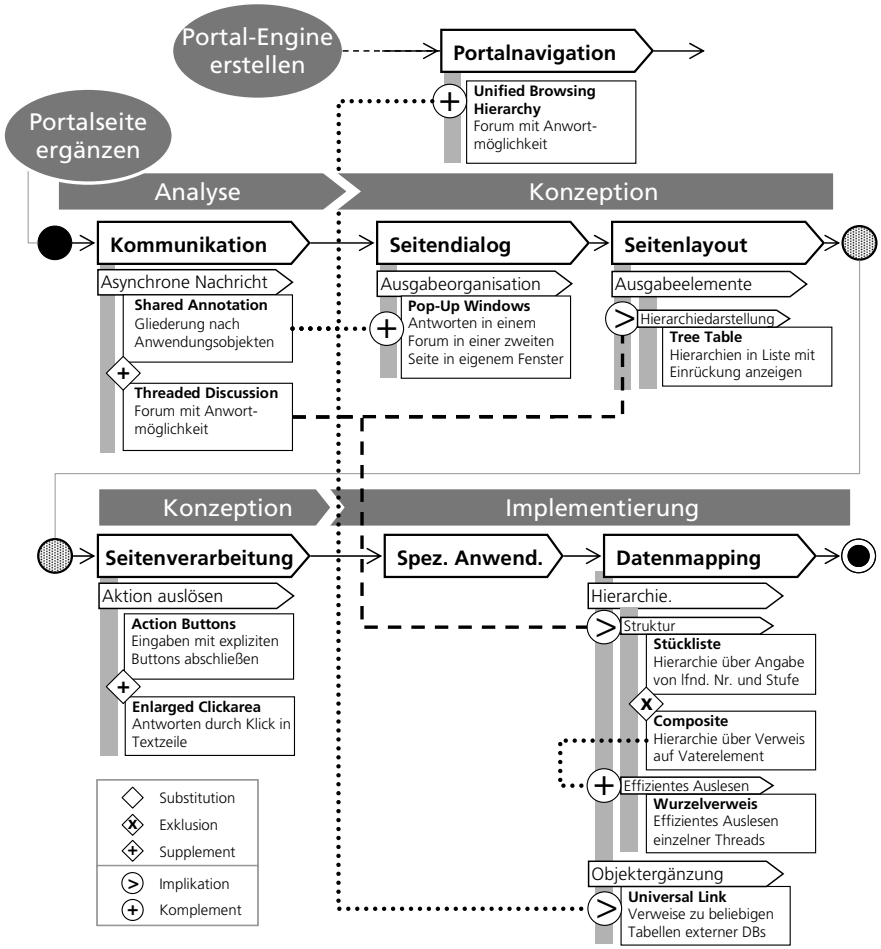


Abbildung 4-8: Aufgabensequenz des Anwendungsbeispiels „Asynchrone Benachrichtigung"

4.5 Werkzeug zur Pflege der Entwurfsmustersprache

4.5.1 Adaption der Entwurfsmustersprache

Entwurfsmustersprachen sind keine statischen, sondern lebendige Konstrukte, die neu gefundene Muster integrieren, veraltete Muster ersetzen und so den Wechsel der Anforderungen und den Fortschritt der Systemtechnologien begleiten [Buschmann98, S. 359ff; Zdun06, S. 991]. Zugleich bieten sie ein Vokabular für Entwickler, das den Austausch von Designideen und Regellösungen erleichtert und stimuliert. Regellösungen zeigen „Best Practices“ für die Einsatzfälle, dokumentieren die Erfahrung von Entwicklungsteams und dienen dem Erhalt von Expertise in den Unternehmen [Pressmann09, S. 332]. Somit garantiert die permanente Adaption einer Entwurfsmustersprache mit neuen Entwurfsmustern, Protopattern und Regellösungen einerseits den aktuellen Stand der Portaltechnik, andererseits ist sie eine Maßnahme für den Transfer von Entwurfswissen im Unternehmen. Einen weiteren Anlass zur Adaption der Entwurfsmustersprache bietet die Ergänzung von Implementierungsmustern passend zur Systemplattform des Einsatzfalls. Technische Fragen der Implementierung und korrespondierende Aufgaben sind zwar berücksichtigt in der Entwurfsmustersprache, jedoch nur mit grundlegenden und plattformneutralen Mustern hinterlegt, um Abhängigkeiten von Systemplattformen zu vermeiden.

4.5.2 Musterbank

Sowohl die permanente Adaption der Entwurfsmustersprache als auch die Auswahl von Entwurfsmustern lässt sich durch ein Werkzeug deutlich komfortabler gestalten. Als Grundlage für die Realisierung von Werkzeugen dient das in Abbildung 4-9 dargestellte UML-Klassendiagramm der Entwurfsmustersprache, das direkt in eine Musterbank umgesetzt werden kann.

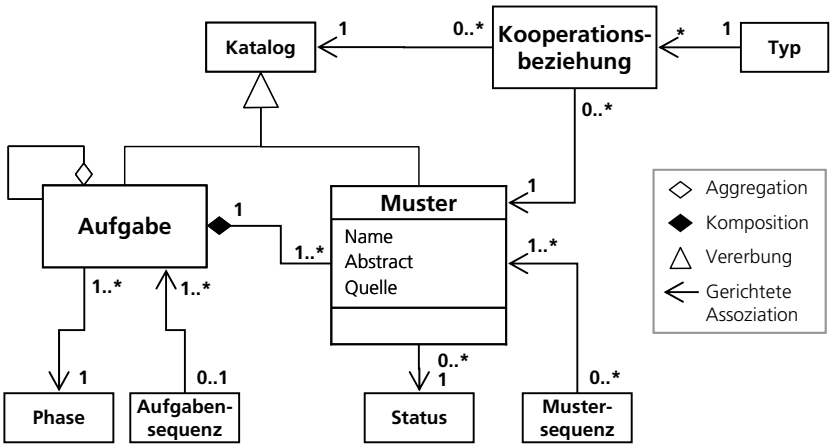


Abbildung 4-9: Struktur der Musterbank in UML-Notation

Den Kern des Klassendiagramms bilden die Klassen Aufgabe und Muster. Die Zuordnung von Mustern zu Aufgaben ist als Klassenkomposition ausgeführt. Ein Muster muss dabei immer Teil einer Aufgabe sein, es darf nicht „frei“ in der Musterbank stehen. Zudem

drückt eine Komposition in UML eine Existenzabhängigkeit aus. Mit einer Aufgabe werden alle zugeordneten Muster aus der Musterbank entfernt, da es keiner Muster für nicht mehr auszuführende Aufgaben bedarf. Zur weiteren Abbildung der Gliederungsstruktur sind die Aufgaben Phasen zugeordnet und über eine Aggregation hierarchisch strukturiert. Die Modellierung von Kooperationsbeziehungen sieht für Muster und Aufgaben eine gemeinsame Superklasse Katalog vor. Dadurch können Kooperationsbeziehungen sowohl zwischen Mustern als auch zwischen Mustern und Aufgaben definiert werden. Sequenzen lassen sich jeweils für Aufgaben und Muster definieren.

In der Praxis sind einfache Werkzeuge mit einer leichten Bedienung, komplexen und hochgradig spezialisierten Werkzeugen vorzuziehen. So genügt zum Beispiel ein Tabellenschema mit den wichtigsten Beschreibungsfeldern für die teaminterne Dokumentation von Entwurfsmustern [Pressmann09, S. 333].

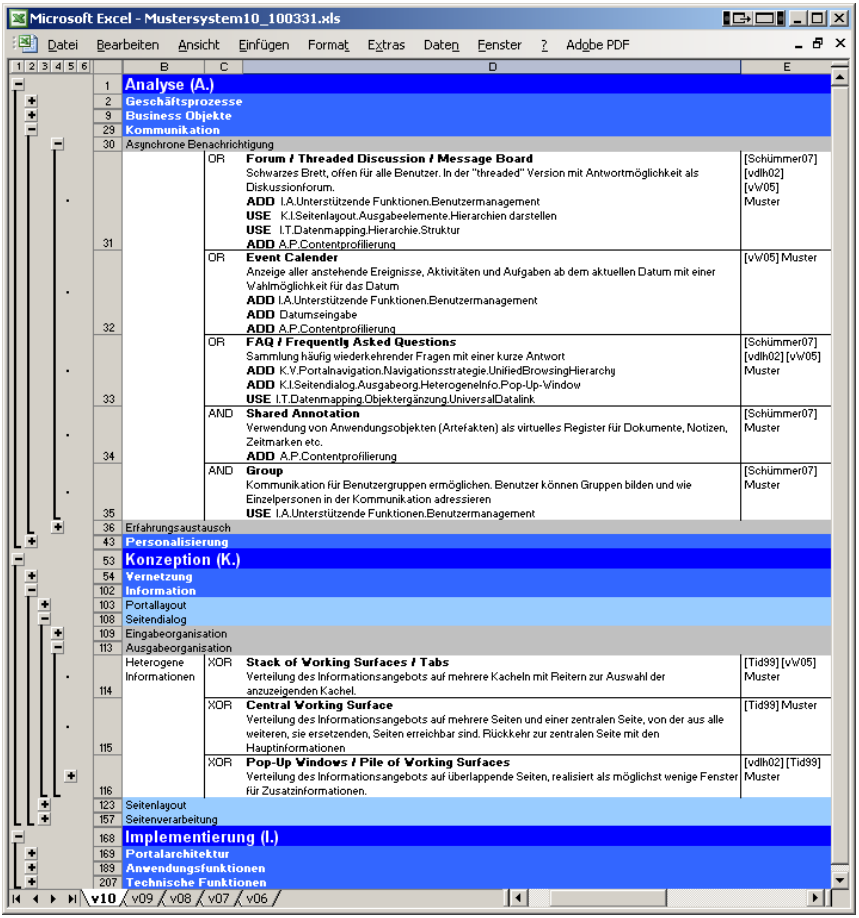


Abbildung 4-10: Praxisorientierte Umsetzung der Musterbank in einer Tabellenkalkulation

Für den Umfang der Entwurfsmustersprache ist eine tabellarische Aufbereitung zweckmäßig und noch ausreichend. Der in Anhang 9.2 gezeigte Entwurfsmusterkatalog wurde für den praktischen Einsatz in das Tabellenkalkulationsprogramm Microsoft Excel übertragen und mit einer Gliederungsfunktion versehen. So lassen sich gezielt Entwurfsmuster einblenden, entsprechend den anstehenden Entwurfsaufgaben, s. Abbildung 4-10.

Mit der Entwurfsmustersprache steht nun eine unterstützende Methode zur Verfügung, entsprechend den Anforderungen aus Kapitel 2.5.1. Sie bietet lösungs- und problemorientierte Konzepte für den kompletten Informationsbedarf produktionsnaher Informationsportale hinsichtlich der konstruktiven Sichten der Analyse, des Entwurfs und der Implementierung. Vor allem ermöglicht sie den Aufbau von Informationsportalen, die den hohen Gestaltungsansprüchen an ein KIP-Infoset genügen. Den Aufbau der Entwurfsmustersprache und das Vorgehen zu ihrer Anwendung illustriert das Fallbeispiel der asynchronen Benachrichtigung. Im nächsten Kapitel wird der Unterschied von KIP-Infosets zu standardisierten Informationsangeboten veranschaulicht. Gezeigt wird die arbeitsplatzspezifische und mitarbeiterspezifische Ausgestaltung eines zentralen Informationsangebots bei KIP, das auf viele, ähnlich gelagerte Einsatzfälle übertragbar ist. Das Informationsangebot wird unter Einsatz der neuen Entwurfsmustersprache konzipiert, und führt zu zwei sogenannten lösungsorientierten Mustersequenzen, die zur Vereinheitlichung und Konformität von Informationsportalen beitragen.

5 Mustersequenzen für kundenindividuelle Auftragsinformationen

Die neue Entwurfsmustersprache vermittelt Lösungskonzepte für den Aufbau produktionsnaher Informationsportale. Sie weist zulässige Kombinationen der Konzepte aus, zeigt anhand der Aufgabensequenzen eine Reihenfolge bei der Anwendung der Lösungsansätze und leitet zu effektiven und erwartungskonformen Informationsportalen. Grundsätzlich generieren Mustersprachen nicht nur ein Informationsangebot zu einer Aufgabenstellung. Vielmehr zeigen sie viele mögliche Lösungskombinationen, decken ihre Vor- und Nachteile auf und weisen auf die zu erwartenden Entwurfsfragen hin. Die bestehenden Freiheitsgrade bei der Lösungskombination sorgen für eine breite Anwendbarkeit von Mustersprachen bei vielen, ähnlich gelagerten Aufgabenstellungen.

Lösungsorientierte Mustersequenzen sind im Gegensatz dazu sehr viel enger gefasst. Sie beschreiben den Lösungsweg für ein realisiertes Informationsangebot, nennen die hierfür eingesetzten Muster und deren Anwendungsreihenfolge. Eine Mustersequenz ist auf den Einsatzfall und seine Randbedingungen abgestimmt, die zum Beispiel bestehen hinsichtlich des Individualisierungsgrad des Produktes, der Form der individualisierten Spezifikationen, der Art der Prozessorganisation, den Produktionstechnologien, der Informationsarchitektur, der Portalplattform etc. Betrachtet man nun eine kundenindividuelle Prozesskette, enthalten die einzelnen Produktionsschritte häufig ähnlich gelagerte Informationsbedarfe. Dies führt zu wiederkehrenden Bestandteilen bei den arbeitsplatzspezifischen Informationsangeboten, die nach einem gleichartigen Prinzip aufgebaut sind. Innerhalb einer Portalanwendung sind daher häufig ähnliche Lösungen realisiert, deren Mustersequenzen sich an nur wenigen Stellen unterscheiden. Eine einmal realisierte und bewährte Mustersequenz ist somit eine Vorlage für weitere Informationsangebote. Die Wiederverwendung lösungsorientierter Mustersequenzen reduziert den Aufwand zur Lösungskonzeption. Vor allem aber sorgt sie für eine Standardisierung und Erwartungskonformität des Informationsangebots in einem Portal. Bei einer Entwicklung eines produktionsnahen Informationsportals sind zunächst mit Unterstützung der Entwurfsmustersprache lösungsorientierte Mustersequenzen zu bestimmen, die grundlegende Konzeptionen für die Informationsangebote vorgeben. Diese Mustersequenzen werden im Anschluss für alle Portalseiten wiederverwendet und modifiziert.

Im Weiteren werden zwei lösungsorientierte Mustersequenzen dargelegt. Am Beispiel der Produktion von elektrischen Schaltschränken werden gezeigt eine Mustersequenz für die arbeitsplatzspezifische Darstellung der Montagereihenfolge mit Auftragsoptionen sowie eine Mustersequenz für die Personalisierung des vorgenannten Informationsangebots. Die Informationsanalyse ergab, dass bei KIP eine effiziente Informationsversorgung nicht generalisierbar ist und Informationsangebote für jeden Einsatzfall aufzubauen sind. Demzufolge ist es nicht möglich, für alle Einsatzfälle und Informationsbedarfe bei KIP Mustersequenzen vorzudefinieren. Die beispielhaften Mustersequenzen wurden jedoch so gewählt, dass sie den Kern der Informationsbedarfe treffen für viele Einsatzfälle im Gegenstandsbereich der Arbeit. Gleiches gilt für die Mustersequenz „Asynchrone Benachrichtigung“ die im letzten Kapitel als Beispiel diente. Alle drei in der Arbeit beschriebenen Mustersequenzen zeigen die Umsetzung der elementaren Anforderungen an ein Informationsangebot und sind nicht auf alle, aber doch auf viele Einsatzfälle übertragbar. Insbesondere zeigen sie deutlich den wesentlichen Unterschied zwischen einem KIP-Infoset und einem generalisierten Informationsangebot.

5.1 Anwendungsbeispiel „Elektrische Schaltschränke“

5.1.1 Kundenindividuelle Produktausführungen

Schaltschränke sind robuste, langlebige und preiswerte Investitionsgüter. Sie dienen der Umhausung, Strukturierung und Sicherung von elektrischen und elektronischen Komponenten in vielfältiger Hinsicht und mannigfaltigen Typen, Maßen, Ausstattungen und Einsatzmöglichkeiten. Als Teil von Werkzeugmaschinen beherbergen sie deren Steuerungskomponenten, in Rechenzentren finden sich 19 Zoll breite Serverstandgehäuse, bei der Netzwerkverteilung kommen Stand-, Wand- und Deckengehäuse zum Einsatz. Für alle Typen werden reichhaltige Zubehöre und variable Innenaufbauten angeboten, z. B. Tastaturkonsolen, Bodenbleche mit Kabelschächten, Kabelverschraubungen und Kabeldurchführungen, s. Abbildung 5-1.

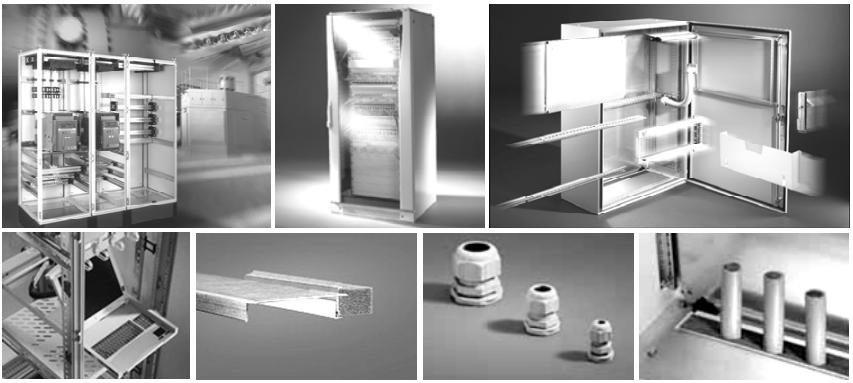


Abbildung 5-1: Beispiele von elektrischen Schaltschränken und Zubehören. Bild: [Rittal GmbH & Co. KG]

Schaltschränke sind auf ihren Einsatzzweck auszurichten und lassen sich aus modularen Baukastensystemen individuell konfigurieren. Beispielsweise besteht ihr Innenaufbau aus Trägerschienen und Befestigungskomponenten, die nach individuellen Layouts in einem Montageraster angeordnet werden können, wie in Abbildung 5-2 gezeigt. Die Varianz des Endprodukts beruht dabei auf der individualisierten Positionierung von identischen Teilen.

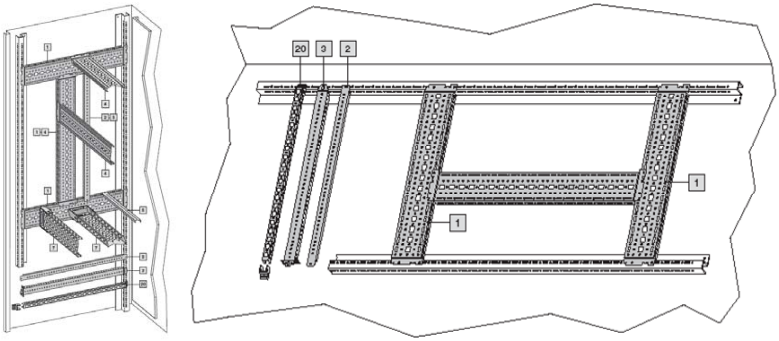


Abbildung 5-2: Rasterprofile für den individuellen Schaltschrankausbau. Bild: [Rittal GmbH & Co. KG]

5.1.2 Produktionsprozess

Als Teil einer umfassenderen Anwendung müssen sich Schaltschränke anspruchslos und unauffällig in ihre Umgebung integrieren. Daraus resultiert ein ständig wachsender Individualisierungsdruck, den die vorhandenen Baukastenlösungen nicht mehr vollständig befriedigen können. Zunehmend sehen sich die Hersteller von Schaltschränken mit Anforderungen konfrontiert, die nur noch durch eine Individualisierung ganzer Prozessketten leistbar ist. Die individuelle Ausgestaltung des Innenaufbaus ist noch vergleichsweise einfach zu lösen. Wie zuvor gezeigt, ermöglicht ein Profilbaukasten mit Rasterabständen die Produktindividualisierung in der Endmontage. Deutlich schwerer zu realisieren sind individuelle geometrische Spezifikationen. Einzelne Aussparungen und Öffnungen im Gehäuse lassen sich durch eine mechanische Nachbearbeitung am fertigen Rohbaugehäuse darstellen. Die Bestrebungen der Hersteller gehen aber bereits dahin, individuelle Gehäuseabmessungen zuzulassen. Die Individualisierung des Produkts beginnt dann bereits mit dem Blechzuschnitt im ersten Prozessschritt.

Abbildung 5-3 zeigt den schematisierten Produktionsprozess von Schaltschränken. Die zur Blechbearbeitung typische Prozesskette beginnt mit dem Trennen der Kontourbleche, die anschließend gebogen, geschweißt und lackiert werden. Parallel sind für den Innenausbau Profilschienen auf die benötigten Maße abzulängen. Alle Komponenten fließen in der mechanischen Montage zusammen, der eine mechanische und elektrische Prüfung und Abnahme folgt, bevor der fertige Schaltschrank versendet wird. Entsprechend der Individualisierungstiefe des Produkts werden die Kundenaufträge in den jeweiligen Prozessschritt der Schaltschrankproduktion eingesteuert. Bei Schaltschränken mit kundenspezifischen Abmessungen durchläuft ein Kundenauftrag den gesamten Produktionsprozess von Beginn an. Für den individuellen Innenausbau nach Kundenangaben ist eine Einsteuerung im Bereich der Montage möglich, da nur Standardteile individuell kombiniert werden. Einige Kunden begnügen sich gar mit reinen Katalogprodukten, die dafür direkt ab Lager sofort erhältlich sind. In der Praxis muss eine Produktionsanlage alle Auftragsarten bedienen können, d. h. sowohl vollständig individualisierte Schaltschränke in Stückzahl 1 bis zur kundenanonymen Produktion von Lagerware in unterschiedlichen Losgrößen.

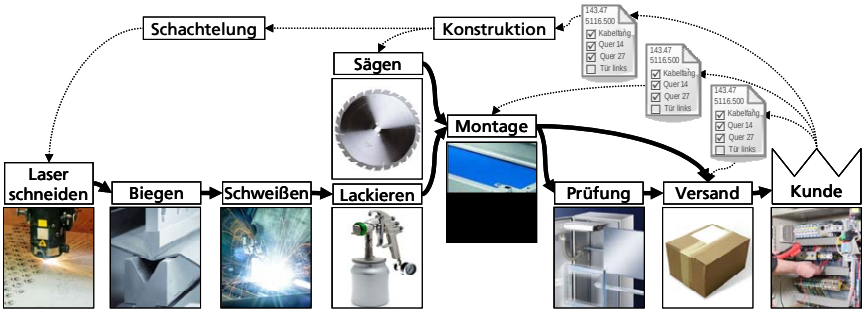


Abbildung 5-3: Schematisierter Produktionsprozess für individualisierte Schaltschränke.

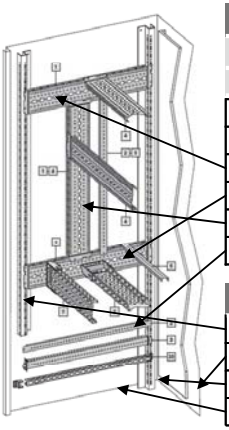
5.1.3 Bedarf an Auftragsinformationen

Eine Fertigung von Schaltschränken mit individuellen Abmessungen und Innenausbau erfordert eine lückenlose Informationskette in der Produktion. Bereits vor dem ersten Produktionsschritt sind konstruktive CAD/CAM-Daten aufzubereiten und NC-Programme zu erstellen. Ein zweiter, vorbereitender Schritt ist die Minimierung des Verschnitts. Dazu werden Teile aus mehreren Aufträgen auf Blechplatinen verschachtelt, getrennt und wieder auftragsspezifisch zurücksortiert. Die Kommissionierung der auftragsspezifischen Teile ist im weiteren Prozess bis in die Endmontage aufrecht zu erhalten. Je nach Flexibilität der eingesetzten Fertigungsanlagen erfolgt beim Biegen, Schweißen und Sägen in der Feinplanung ebenfalls eine Losbildung über einen Auftragsvorrat. Gleiches gilt für die Lackierung, bei der Teile mit gleichen Farben zu einer Lackfolge entsprechend der Farbkampagne zusammengefasst werden. Für den kundenindividuellen Innenausbau werden in der Montage vor allem Angaben über die Anzahl, Art und Anordnung der Einbauten benötigt, insbesondere der einzusetzenden Profile und deren Position. Des Weiteren wünschen Kunden unterschiedliche Prüfzertifikate und bestimmen damit den Prüfumfang. Die Prüfungen erfolgen dabei nicht nur unter dem Aspekt der internen Qualitätssicherung, sondern sind vom Kunden in wechselnder Kombination explizit verlangte Produkteigenschaften und daher optionale Bestandteile der Auftragsstückliste. In Tabelle 5-1 sind die in den einzelnen Prozessschritten benötigten Kerninformationen aufgestellt.

Informations-cluster	Prozessschritt					
	Laser-schneiden	Biegen, Schweissen	Lack	Sägen	Montage	Prüfen
Auftragsabwicklung						
Auftragsoptionen	Maßangaben		Farbe	Maßangaben	**Montageposition**	
Produktunterlagen: - Auftragsstückliste	-Korpus			-Profile	**- Profile** **- Zubehör**	-Prüfposition
- Zeichnung	-Teile				**- Zusammenbau**	
Arbeitsplan	NC-Programm	NC-Programm			**Montagefolge**	Prüfschritte
PPS						
Fertigungsaufträge	Auftragsvorrat	Auftragsvorrat	Auftragsvorrat	Auftragsvorrat		
Feinplanung	Schachtelung	Losbildung	Lackfolge	Losbildung		
Ressourcenplan			Farbkampagne			

Tabelle 5-1: Auftragsinformationsbedarf in der kundenindividuellen Schaltschrankproduktion

In einer abstrahierenden Sicht ist der Informationsbedarf in allen Prozessschritten nahezu gleich. Benötigt werden die Auftragsspezifikation, Produktunterlagen wie Stücklisten und Zeichnung, sowie der Arbeitsplan und die anstehenden Fertigungsaufträge. Tatsächlich benötigt wird jedoch immer nur ein wechselnder, arbeitsplatzspezifischer Ausschnitt des gesamten Informationsinhalts. Ausgehend von diesem Informationsbedarf werden im Folgenden ein arbeitsplatzspezifisches und ein mitarbeiterspezifisches Informationsangebot für die Montage entwickelt. Zur Veranschaulichung ist den Darstellungen des Informationsangebots ein fiktiver Auftrag hinterlegt, dessen kundenindividuelle Auftragsoptionen Tabelle 5-2 auflistet. Hervorgehoben sind die Kerninformationen für die Montage.



Auftragsspezifikation			
Produkt CM 5116.500			
Individuelle Ausstattung	**Merkmal**	**Wert**	**Einheit**
Asym. Kabelverschraubung	**Position**	**Vorne**	
Kabelabfangschiene	**Schienensitz**	**Diagonal**	
TS Quersteg	**Montageposition**	**14**	Raster
TS Quersteg	**Montageposition**	**27**	Raster
TS Vertikalsteg	**Montageposition**	**9**	Raster
Montagesteg	**Montageposition**	**13**	Raster

Standardteile
Grundschiene
Rückwand
Eckwinkel
Seitenwand

Tabelle 5-2: Beispielhafte kundenindividuelle Auftragsoptionen eines Schaltschranks.

5.2 Arbeitsplatzspezifische Informationsbereitstellung

Das Vorgehen zur Konzeption eines arbeitsplatzspezifischen Informationsangebots folgt der Aufgabensequenz zum Ergänzen von Portalseiten und wird in dieser Reihenfolge beschrieben, s. Abbildung 5-4.

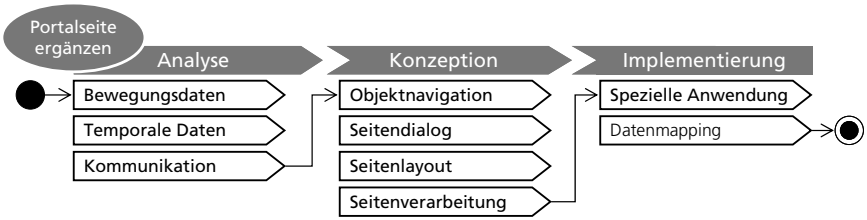


Abbildung 5-4: Mustersequenz „Portalseiten ergänzen“

5.2.1 Analyse des Informationsbedarfs

Bewegungsdaten

Ausgangspunkt ist das Analysemuster PRODUKTIONSAUFTRAG, s. Abbildung 5-5. Es zeigt die als Informationsbedarf erkannten Elemente und deren Beziehungen in einer für die Serienproduktion typischen Konstellation, die auch von gängigen ERP-Systemen benutzt wird. Alle Exemplare eines Materials oder Artikels werden als identisch betrachtet, und als zählbare oder volumenmäßig erfassbare Einheiten dargestellt. Die identischen, technischen, kaufmännischen und dispositiven Eigenschaften werden in einem Artikelstammdatensatz zusammengefasst und zentral abgelegt. Eine Stückliste beschreibt den Aufbau eines Artikels aus Teilen oder Materialien. An den Artikelstammdaten hängen ihre Bewegungsdaten. Arbeitspläne geben Vorlagen für die Produktionsprozesse und fassen die Arbeitsgänge zusammen, die zur Fertigung eines Artikels auszuführen sind. Terminierte Produktionsvorgänge an Stationen werden über Fertigungsaufträge ausgelöst.

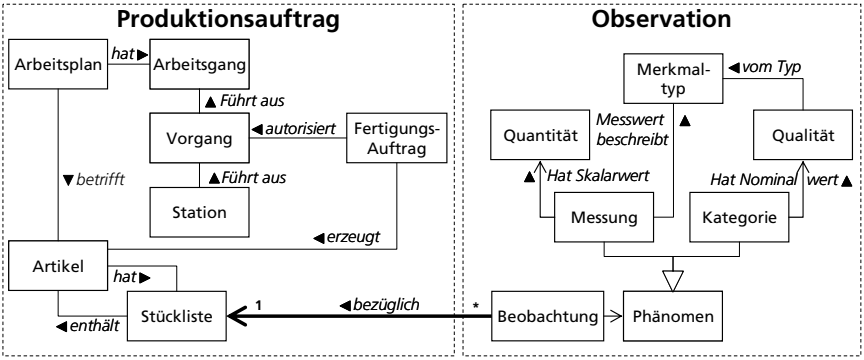
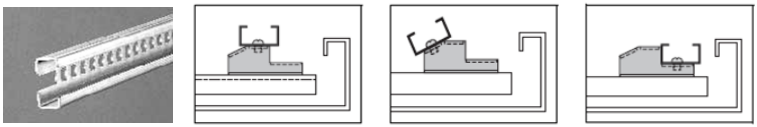


Abbildung 5-5: Analysemuster Produktionsauftrag und Observation. Nach: [Hay96; Fowler97]

Kundenindividuelle Produktionsprozesse zeigen Eigenschaften, die durch das an der Serienproduktion orientierte Muster nicht vollständig abgedeckt sind. So müssen in dem Muster Teile mit abweichenden Merkmalen immer als eigene Artikelstämme angelegt werden. Ein aufwendiger Vorgang, da hierfür dispositive Angaben, Bezugsquellen, Lagerorte etc. festzulegen sind. Bei KIP werden dagegen Teile in Einzellosen gefertigt, die eine große Ähnlichkeit aufweisen, aber in Details individuell ausgeprägt sein können. Die Unterschiede sind für die Disposition und den Bezug der Teile nicht relevant, sondern nur für einzelne, genau festgelegte, kundenspezifische Bearbeitungsschritte. Das Anlegen und Erfassen von Artikelstammdaten für Artikelvarianten führt zu einem erheblichen Mehraufwand. Daher sind kundenindividuelle Auftragsoptionen als Bewegungsdaten anzusehen und als Bestandteil der Artikelstammdaten zu vermeiden. Zur Beschreibung der auftragsspezifischen Ausprägung der Teile wird PRODUKTIONSAUFTRAG mit dem Muster OBSERVATION kombiniert. OBSERVATION zeigt ein Schema für die Dokumentation von Merkmalswerten unabhängig vom beobachteten Objekt, s. Kapitel 3.2. Übertragen auf KIP lassen sich zu einem Artikelstamm die kundenindividuell ausgeprägten Auftragsoptionen „beobachten". Bei elektrischen Schaltschränken beispielsweise die Option „Schienensitz" von Kabelabfangschienen, s. Abbildung 5-6. Möglich Werte sind „Oben", „Diagonal" oder „Seitlich". Weitere Optionen können für denselben Gegenstand vorliegen, zum Beispiel eine „Montageposition" mit skalaren Werten. Auch mehrfache Beobachtungen der Optionen sind möglich, zum Beispiel im Falle von Auftragsänderungen.



Observationen für den Artikel Kabelabfangschiene					
Beobachtung	Phänomen				
	Messung	Quant.	Kategorie	Qualität	Merkmaltyp
Auftragseingang			Schienensitz	Diagonal	
Auftragseingang	Montageposition	19			Raster
Änderung	**Montageposition**	**14**			Raster

Abbildung 5-6: Beispielhafte Observation von kundenindividuellen Auftragsoptionen

Neben den fehlenden Möglichkeiten zur Abbildung von Auftragsoptionen weist das Muster PRODUKTIONSAUFTRAG einige Strukturen auf, die die speziellen Anforderungen von kundenindividueller Produktion nicht optimal unterstützen und Anpassungen sowie den Einsatz weiterer Muster erfordern.

Artikel können sich in einer KIP nicht nur durch einzelne Teilemerkmale unterscheiden. Bei Verwendung eines modularen Baukastensystems ergeben sich auch individuelle Aufbauten der Produktexemplare. Das Analysemodell wird daher erweitert um Auftragsstücklisten für Produktexemplare. Zur Unterscheidung werden die bestehenden Stücklisten für Standardartikel nun als Baukasten-Stücklisten bezeichnet. Die einstufigen Baukasten-Stücklisten zeigen alle Kombinationsmöglichkeiten eines Produkts. Im Gegensatz dazu beschreiben Auftragsstücklisten den Aufbau von einzelnen Produktexemplaren über alle Stufen. Zur Abbildung wird das Muster COMPOSITE verwendet, das den Aufbau hierarchischer Baumstrukturen ermöglicht, s. Abbildung 5-7.

Eine kundenindividuelle Produktion erfolgt naturgemäß nicht mehr in Losen. Somit referenziert ein Fertigungsauftrag genau ein kundenspezifisches Produktexemplar. Die bestehende Beziehung von Fertigungsaufträgen zu Artikeln wird daher in eine Beziehung zu Auftragsstücklisten geändert.

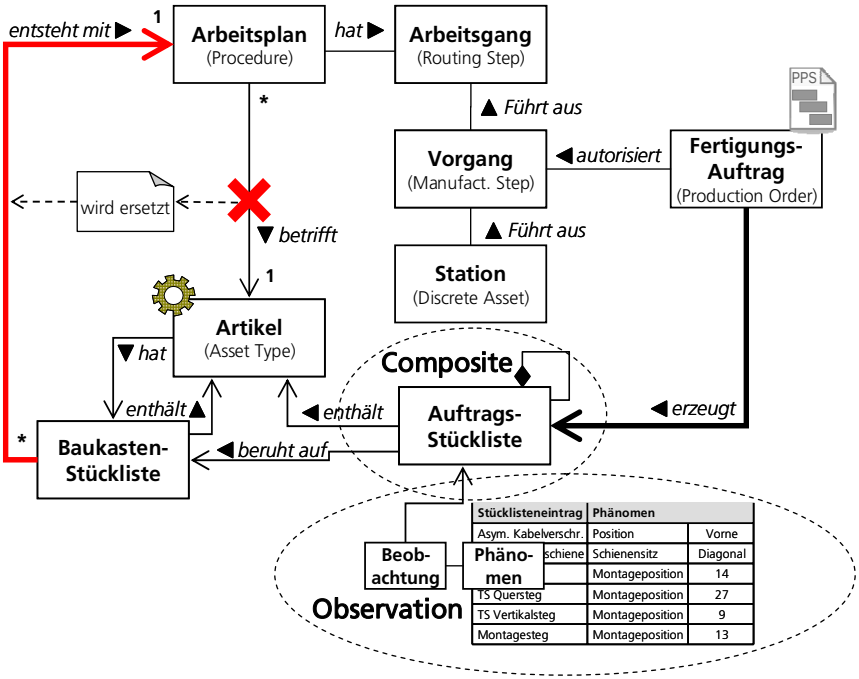


Abbildung 5-7: Modellierung kundenindividueller Produktionsaufträge.

Als letzte Änderung werden für kundenindividuelle Produkte eindeutig definiert Produktionsabläufe in dem Modell fixiert. Das ursprüngliche Muster gestattet mehrere alternative

Arbeitspläne je Stückliste und somit variable Produktionsabläufe. Bei einer Serienfertigung in Losen rechtfertigt das Optimierungspotenzial bei der Auslastung der Fertigungskapazitäten den damit verbundenen höheren Planungsaufwand. Bei einer kundenindividuellen Produktion wäre jedoch im Auftragsfall für jedes kundenindividuelle Teil einer der alternativen Arbeitspläne auszuwählen. Eine mühsame Prozedur ohne wirkliches Optimierungspotenzial, da nur stückweise gefertigt wird. Eine kundenindividuelle Produktion verlangt daher flexible und eindeutig definierte Produktionsprozesse, die in beliebigen Stückzahlen ausführbar sind. Zur Fixierung des Produktionsablaufs wird eine neue Zuordnung aufgebaut zwischen Baukasten-Stücklisten und Arbeitsplänen. Jeder Baukasten-Stückliste und somit jedem Artikel ist ein Standardarbeitsplan zugewiesen und der Produktionsablauf eindeutig festgelegt. Alternative Arbeitspläne werden nicht mehr berücksichtigt, die Beziehung von den Arbeitsplänen zu den Artikeln entfällt.

Temporale Daten und Kommunikation
Die Aufgabensequenz für Portalseiten sieht noch vor, den Informationsbedarf an temporalen Daten und Kommunikationsfunktionen zu analysieren. Beide Informationsarten treten auf bei Änderungen der Kundenaufträge und der Auftragsoptionen. Die Montagezeit von Schaltschränken ist jedoch relativ kurz, je nach Ausstattungsumfang im Bereich von einigen Minuten, selten mehr als eine Stunde. Auftragsänderungen finden somit immer vor Beginn der Montage statt, Änderungen während der Montage sind unwahrscheinlich. Beide Informationsarten sind somit nicht Teil der Kerninformationen für die Montage von Schaltschränken. Die Aufgaben und die zugeordneten Muster sind nicht zutreffend, die Schritte der Aufgabensequenz werden ausgelassen.

5.2.2 Konzeption des Informationsangebots

Im nächsten Schritt der Aufgabensequenz ist ein Informationsangebot zu konzipieren. Aus der Analysephase ergaben sich als Kerninformationen für die Montage die Auftragsoptionen mit den Innenausbauteilen, deren Montageposition und ihre im Arbeitsplan festgelegte Montagereihenfolge. Hierfür ist eine bedarfsgerechte Navigation zu gestalten.

Objektnavigation
Eine Objektnavigation findet statt, zur Bestimmung der anzuzeigenden Entitäten aus einer Datenbank. Eine mögliche Navigationsstrategie ist die Eingrenzung passender Entitäten auf den Nutzungskontext, im Anwendungsbeispiel auf die Arbeitsstation und die Planungsperiode. Dazu passende Entitäten sind alle Fertigungsaufträge, die an der Arbeitsstation anstehen, wobei die Periode der aktuelle Tag, eine Schicht oder bei entsprechend kurzzyklischen Tätigkeiten ggf. wenige Stunden sein können. Der Kontext ergibt sich somit aus der Rolle „ausführende Arbeitsstation" für den Fertigungsauftrag, das Muster wird als ROLLENKONTEXT bezeichnet, s. Abbildung 5-8.

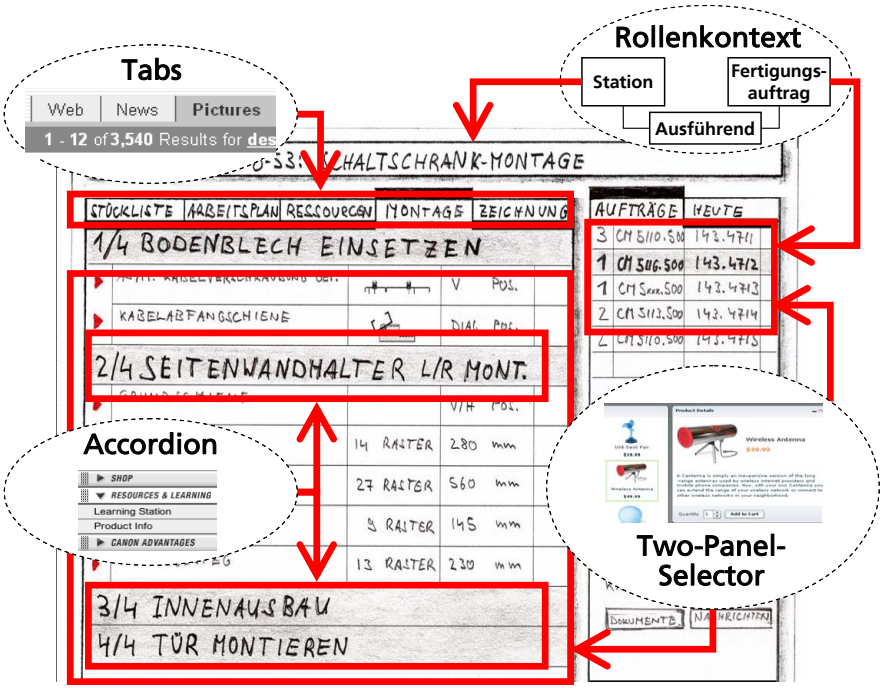


Abbildung 5-8: Gliederung des Informationsangebots und Navigationselemente

Seitendialog

Der Umfang der Kerninformationen verhindert, sie vollständig auf einer Bildschirmseite darzustellen. Eine Seitennavigation ist vorzusehen, die möglichst wenige Interaktionen vom Nutzer verlangt. Zur Seitennavigation bietet sich das Muster TWO-PANEL-SELECTOR an, das eine Übersichtsleiste für Objekte mit einem Anzeigebereich für Detailinformationen kombiniert. In der Übersichtleiste sind die Fertigungsaufträge gemäß dem Rollenkontext aufgelistet. Im Detailbereich werden zu jedem Fertigungsauftrag die zugehörigen Auftragsoptionen dargestellt und unterteilt in klappbare Abschnitte, die bedarfsweise ein- oder ausgeblendet sind. Bei Verwendung von Touchscreens und Ausführung der Gruppierungszeilen als ENLARGED CLICK-AREAS können die Mitarbeiter durch Antippen einzelne Abschnitte rasch ein- respektive ausklappen. Das Muster wird entsprechend dem Verhalten plakativ als ACCORDION bezeichnet.

Die Bereiche und Navigationselemente der Muster TWO-PANEL-SELECTOR und ACCORDION sind permanent angezeigt. Mit nur einem Klick können die Benutzer Fertigungsaufträge auswählen und deren Kerninformationen einblenden. Das erreichbare Informationsangebot deckt dabei den Arbeitsvorrat eines längeren Zeitraums ab. Nur selten muss die Maske für eine Objektnavigation verlassen werden. Dies vermeidet unnötige Navigation und führt in Verbindung mit der übersichtlichen Präsentation zu einem geringen Aufwand für die Informationsbeschaffung.

Den Kerninformationen sind flankierend Komplementärinformationen beizustellen. Die Reiterleiste entsprechend dem Muster TABS führt direkt zu den häufigsten Komplementärinformationen, ohne die Grundmaske verlassen zu müssen. Im Beispiel sind dies die vollständige Auftragsstückliste, der komplette Arbeitsplan mit allen Standardarbeitsgängen, Informationen zu den Ressourcen und detaillierte Zeichnungen.

Seitenlayout und Seitenverarbeitung

Die Navigationsstruktur ist damit vollständig. Als letzter Schritt der Konzeption sind das Seitenlayout und die Verarbeitungsfunktionen festzulegen. Der Seitendialog mit den TABS, dem TWO-PANEL-SELECTOR und dem ACCORDION liefert bereits Vorgaben für die Gestaltung der Seite. Es bleibt noch den Anzeigebereich mit den Detailinformationen über die kundenindividuellen Auftragsoptionen zu gestalten. Entsprechend den Analysemustern sind die Auftragsoptionen der Auftragsstückliste und den zugehörigen OBSERVATIONEN zu entnehmen. Zu jedem Fertigungsauftrag wird die hinterlegte Auftragsstückliste tabellarisch dargestellt. In die Tabelle werden die observierten Phänomene, wie die Montagepositionen etc. aufgenommen. Die Gruppierungszeilen für das ACCORDION ergeben sich aus den Arbeitsgängen des Arbeitsplans und entsprechen somit der Montagereihenfolge, s. Abbildung 5-9.

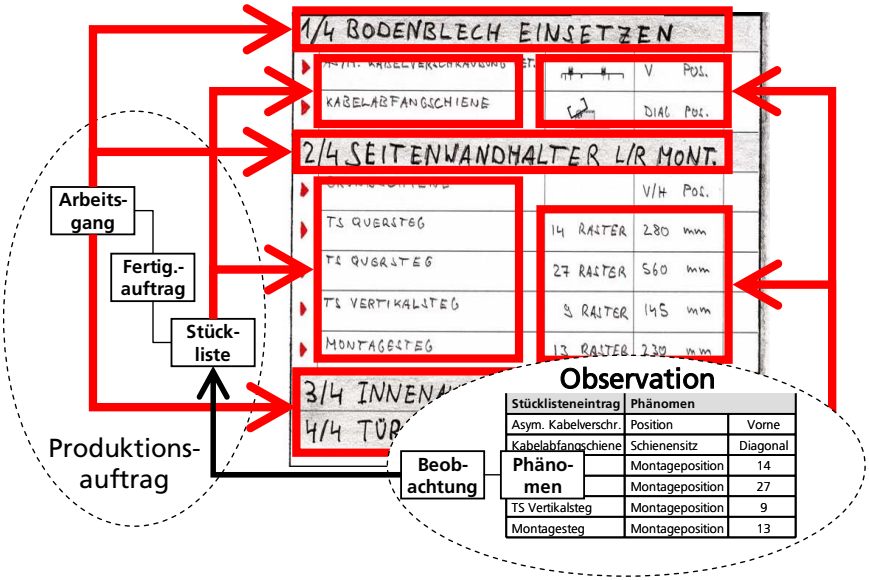


Abbildung 5-9: Darstellung der Auftragsoptionen

Zur weiteren Ausgestaltung des Seitenlayouts sieht die Mustersprache passende Entwurfsmuster für die tabellarische Informationsaufbereitung vor, z. B. ALTERNATING ROW COLOURS. Im Anwendungsbeispiel kommt ihnen keine Schlüsselfunktion zu, daher wird hier auf ihre Erläuterung verzichtet. Ebenfalls übergangen werden Entwurfsaufgaben und Entwurfsmuster für Seitenverarbeitungsfunktionen und Eingabeelemente, die nicht vorgesehen sind für das Informationsangebot.

5.2.3 Implementierung einer Portalseite

Die Aufgabensequenz endet mit den Entwurfsaufgaben zur Implementierung spezieller Anwendungsfunktionen und des Datenmappings.

Spezielle Anwendungen
Das Informationsangebot für die Schaltschrankmontage beschränkt sich auf die effiziente Bereitstellung von Auftragsoptionen. Eingabe- und Verarbeitungsfunktionen sind nicht vorgesehen. Es bedarf daher keine Entwurfsmuster für die Implementierung spezieller Anwendungsfunktionen.

Datenmapping
Im weiteren Verlauf der Implementierung ist ein realisierungsfähiges Datenmodell zu entwerfen. Grundlage sind die Analysemuster aus dem ersten Schritt, auf die weitere Implementierungsmuster angewandt werden, wie in Abbildung 5-10 dargestellt. Eine erste Implementierungsaufgabe besteht in der Verbindung von OBSERVATIONEN mit Stücklisten. Hierfür wird das Entwurfsmuster DECORATOR eingesetzt, um bestehende Klassen mit zusätzlichen Elementen zu „dekorieren".

Weiter bestehen vielen Gemeinsamkeiten zwischen den Datenstrukturen der Auftrags- und der Baukasten-Stückliste. Eine objektorientierte Programmierung sieht für solche Fälle die Bildung einer Oberklasse vor, mit Vererbung der gemeinsamen Eigenschaften auf die spezialisierten Subklassen. Das Muster SINGLE TABLE INHERITANCE überträgt das Prinzip auf relationale Datenmodelle. In der Datenbank wird eine Tabelle mit allen Attributen beider Subklassen angelegt und einem zusätzlichen Attribut, das die jeweilige Zuordnung eines Datensatzes zu einer Subklasse anzeigt. Zur weiteren Vereinheitlichung wird die Baukasten-Stückliste ebenfalls als COMPOSITE ausgeführt, dessen Elemente einstufig angelegt sind. Damit können sowohl die Hierarchie als auch alle Beziehungen zu den Arbeitsplänen und Arbeitsgängen auf die Oberklasse übertragen werden. Die einheitliche Erfassung von Stücklisten in einer gemeinsamen Tabelle mit SINGLE TABLE INHERITANCE reduziert den Implementierungsaufwand erheblich. Aus dem gleichen Grund werden die Entitäten Arbeitsplan und Arbeitsgang in einem COMPOSITE vereinigt. Arbeitsgänge sind dann hierarchische Verfeinerungen eines Arbeitsplans. Neben der einheitlichen Speicherung in einer Datenbanktabelle besteht zusätzlich die Möglichkeit, weitere Hierarchiestufen einzuziehen, und Arbeitsgänge in Teilverrichtungen zu unterscheiden.

Für die bedarfsgerechte Präsentation der optionalen Stücklistenpositionen entsprechend der Montagereihenfolge bedarf es für jede Position der Auftragsstückliste der Kenntnis des Arbeitsgangs, in dem das Teil verbaut wird. Dazu wird eine weitere Verwendungsbeziehung aufgebaut zwischen Stücklisten und dem Composite „Arbeitsplan & Arbeitsgänge", sodass nun zwei Beziehungen bestehen mit abweichender Semantik. Das Wurzelelement einer Baukasten-Stückliste steht für das Erzeugnis und verweist auf den Arbeitsplan, mit dem es erzeugt wird. Seine Komponenten zeigen auf die sie verwendenden Arbeitsgänge. Aus den Baukasten-Stücklisten werden Auftragsstücklisten kopiert durch eine vollständige Stücklistenauflösung. Sie übernehmen sowohl deren Struktur, als auch deren Beziehungen. Ihre Komponenten verweisen auf die gleichen Arbeitsgänge und erlauben dadurch die Anordnung der Auftragsstücklisten in der Montagereihenfolge.

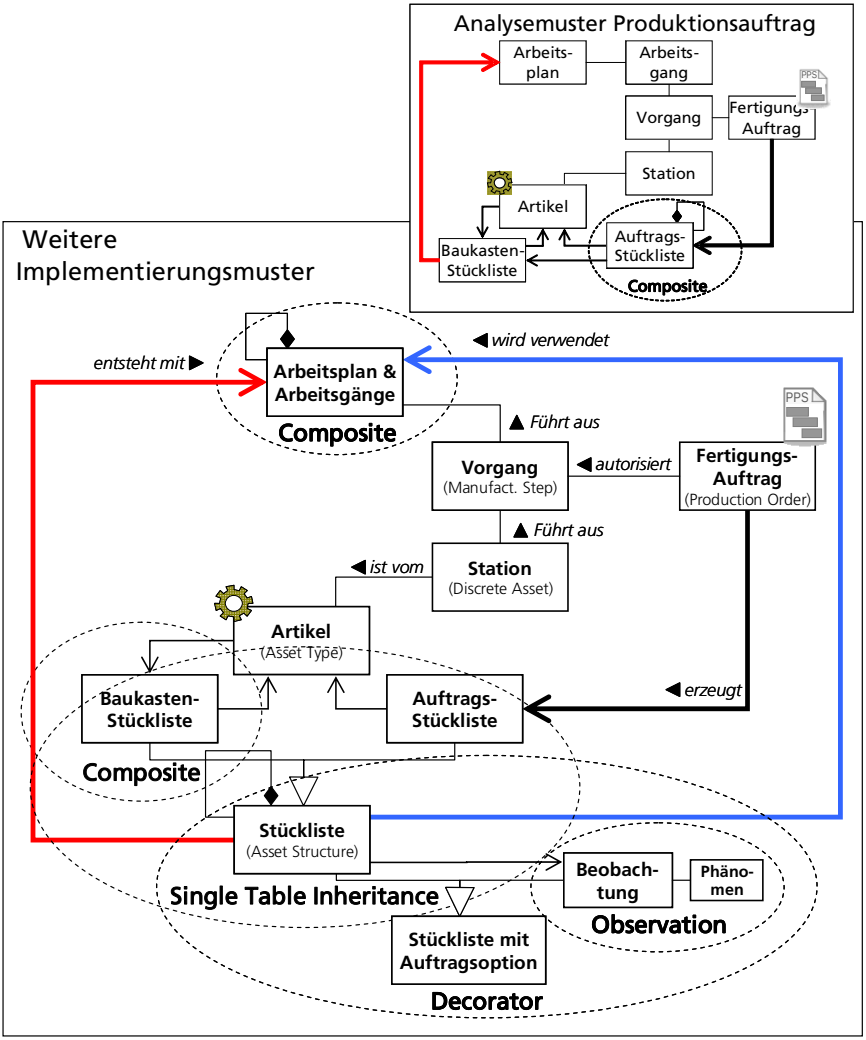


Abbildung 5-10: Implementierungsbedingter Einsatz von Entwurfsmustern zur Datenmodellierung

Das Objektdiagramm in Abbildung 5-11 verdeutlicht die neuen Beziehungen am Beispiel des fiktiven Kundenauftrags aus Tabelle 5-2. Gezeigt ist der Aufbau von Baukasten-Stückliste, Auftragsstückliste und Arbeitsplan für einen Schaltschrank vom Typ „CM 5116.500“, bestehend aus einem Korpus, einer Grundschiene sowie diversen Querstreben. Der Aufbau der Auftragsstückliste entspricht der Struktur der Baukasten-Stückliste, kann allerdings kundenindividuell abgewandelt sein. So enthält die Auftragsstückliste zwei seitliche Querstreben, die an Position 14 und 27 zu montieren sind. Diese wurden

aus dem gleichen Teil der Baukasten-Stückliste kopiert und übernehmen somit auch die Verwendungsbeziehung zum Arbeitsgang „Seitenwand montieren“.

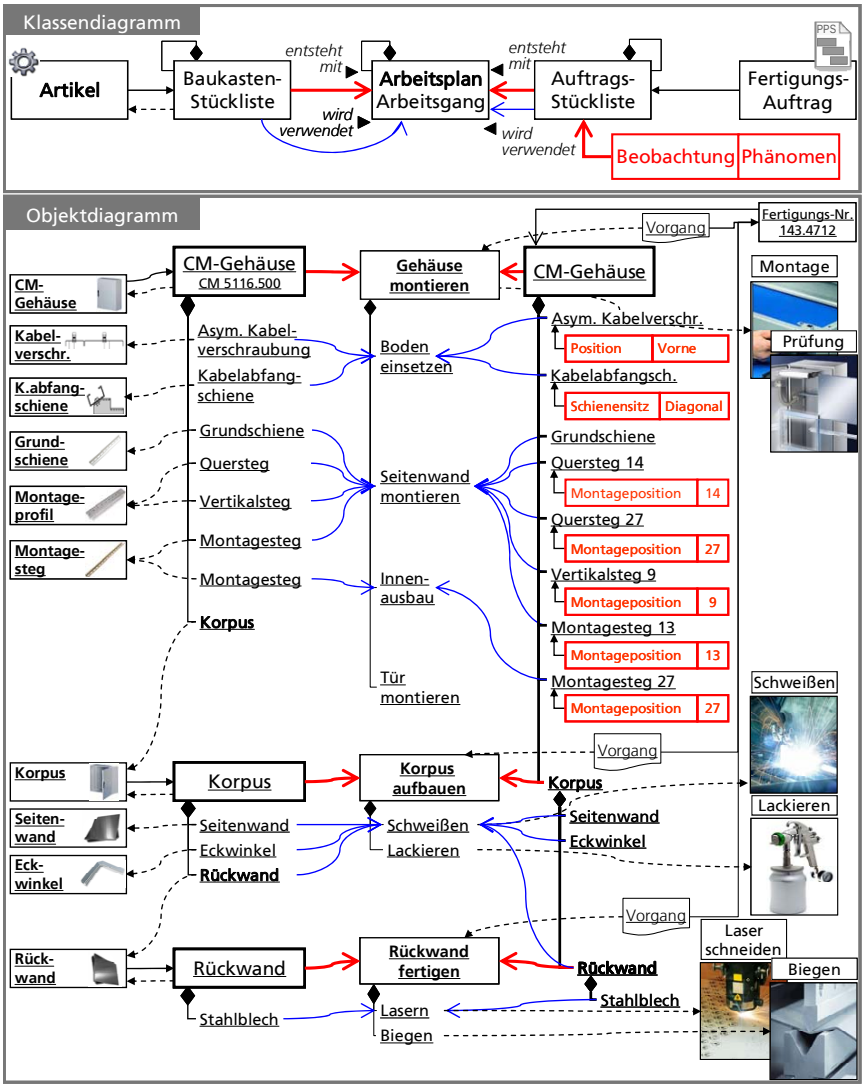


Abbildung 5-11: Beispielhaftes Objektdiagramm zum implementierten Datenmodell.

Das überarbeitete Datenmodell vereinfacht die Programmierung der Portalseiten. Baukasten- und Auftragsstücklisten können nun einheitlich implementiert und mit denselben Prozeduren und Portalseiten dargestellt werden. Abbildung 5-12 zeigt beispielhaft eine

Seite einer prototypischen Realisierung eines Informationsportals für das hier konzipierte Informationsangebot, s. Kapitel 6.2. Die Portalseite zeigt eine Übersicht der angelegten Stücklisten. Die linke Tabelle gibt eine Übersicht der Auftrags- und Baukastenstücklisten in einer gemeinsamen Struktur. Je nach Stücklistenart wird entweder der zugehörige Auftrag oder der zugeordnete Arbeitsplan mit angezeigt. In der rechten Tabelle ist die ausgewählte Stückliste stets gleichartig dargestellt, unabhängig von der Stücklistenart.

Abbildung 5-12: Einheitliche Darstellung und Verarbeitung von Baukasten- und Auftragsstücklisten

5.2.4 Lösungsorientierte Mustersequenz

Das gezeigte Vorgehen folgt der Aufgabensequenz zum Erstellen von Portalseiten. Insgesamt wurden zehn verschiedene Entwurfsmuster eingesetzt, die eine lösungsorientierte Mustersequenz für die arbeitsplatzspezifische Präsentation von Auftragsoptionen ergeben. Tabelle 5-3 zeigt die Entwurfsmuster der lösungsorientierten Mustersequenz, die das Informationsangebot, dessen Struktur, Navigation und Präsentation wesentlich prägen.

Aufgabe	Eingesetzte Entwurfsmuster
Analyse	
Bewegungsdaten	**PRODUKTIONSAUFTRAG** Modellierung von Produktionsablauf und Aufträgen. **OBSERVATION** Darstellung von Werten und Beobachtungen zu einem Objekt.
Temporale Daten & Kommunikation	Diesbezügliche Informationsbedarfe liegen nicht vor, die Schritte der Aufgabensequenz werden ausgelassen.
Konzeption	
Objektnavigation	**ROLLENKONTEXT** Einschränkung der Objekte auf einen Anwendungskontext.
Seitendialog	**TWO-PANEL-SELECT** Objektübersicht mit Detailsicht zu einem ausgewählten Objekt. **TABS** Verteilung des Informationsangebots auf mehrere Kacheln mit Reitern. **ACCORDION** Aufklappbare Abschnitte in einer Liste homogener Informationen.
Seitenlayout & Seitenverarbeitung	Im Anwendungsbeispiel kommt ihnen keine Schlüsselfunktion zu, daher wird auf ihre Erläuterung verzichtet.
Implementierung	
Spezielle Anwendungen	Keine speziellen Verarbeitungsfunktionen erforderlich, der Schritt der Aufgabensequenz wird ausgelassen.
Datenmapping	**COMPOSITE** Abbildung von Hierarchien als Verweis auf die eigene Klasse. **SINGLE TABLE INHERITANCE** Vereinigung von Unterklassen in einer gemeinsamen DB-Relation. **DECORATOR** Flexible Erweiterung von Klassen für die Zuordnung von Observationen mit Stücklisten.

Tabelle 5-3: Mustersequenz für die Portalseite mit Montageinformationen

Die Mustersequenz kann mit geringen Variationen auf das Informationsangebot für die Stationen Laserschneiden und Prüfung übertragen werden, s. Abbildung 5-13. Beide Portalseiten weisen viele Gemeinsamkeiten mit dem Informationsangebot für die Montage auf. Sie verwenden ebenso PRODUKTIONSAUFTRAG als zentrales Business Objekt, die gleiche Objektnavigation über den ROLLENKONTEXT und eine Seitennavigation mittels TWO-PANEL-SELECT und TABS. Das Informationsangebot für die Station Laserschneiden listet die geschachtelten Teile auf, die in einem Produktionsauftrag gemeinsam gefertigt und aus einer Blechplatine ausgeschnitten werden. Die Anzahl der Teile und deren Schattenriss unterstützten die Mitarbeiter beim auftragsspezifischen Kommissionieren der ausgetrennten Teile. Eine Unterteilung der Liste mit ACCORDION ist entbehrlich, da nur wenige Teile auf einer Platine verschachtelt sind, und die Tabelle mit einer Bildschirmseite auskommt. Für die Prüfung der Schaltschränke benötigen die Montagemitarbeiter die Angabe der kundenindividuell auszuführenden Prüfschritte, die eigene Positionen der Auftragsstückliste darstellen. Anhand von Prüfarbeitsplänen wird die Auflistung mit einem ACCORDION unterteilt, wie bei der Montagereihenfolge. Ergänzend dazu ist der Seite eine Eingabefunktion hinterlegt, über die Mitarbeiter die Durchführung der Prüfschritte quittieren, ohne das Informationsmedium verlassen zu müssen.

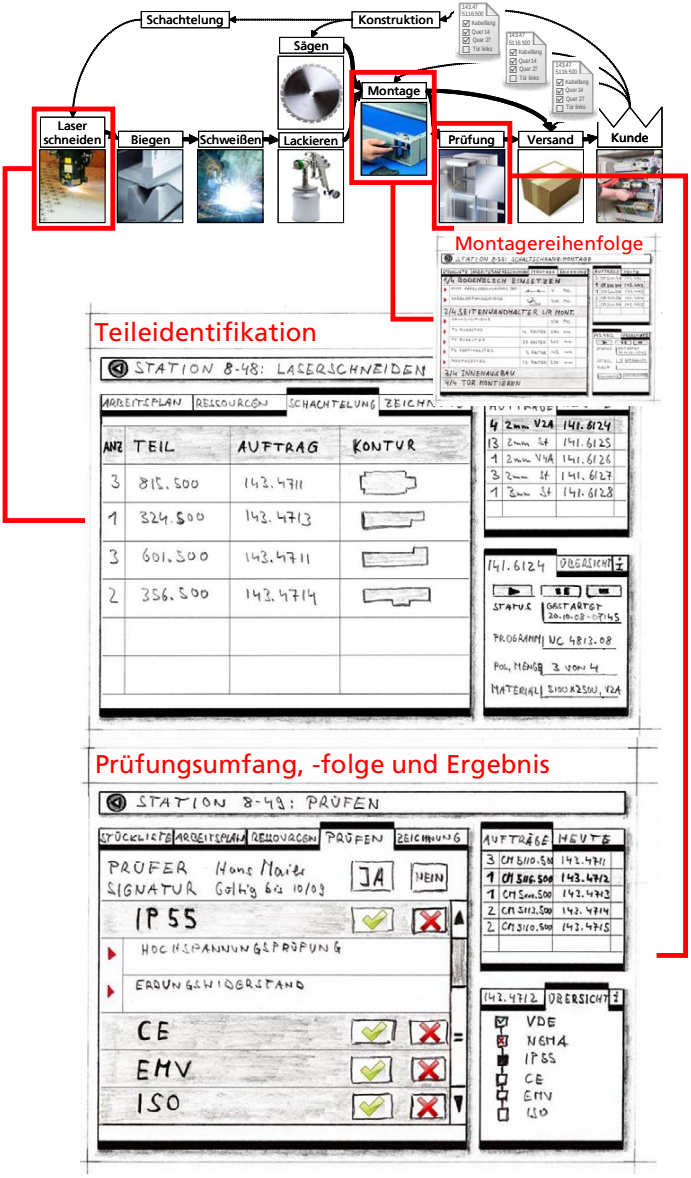


Abbildung 5-13: Wiederverwendung von Mustersequenzen.

5.3 Mitarbeiterspezifische Informationsangebote

Eine effiziente, produktionsnahe Informationsversorgung bei KIP ist arbeitsplatzspezifisch und mitarbeiterspezifisch auszurichten. Das im letzten Abschnitt entwickelte Informationsangebot zeigt die arbeitsplatzspezifische Bereitstellung von Kerninformationen für die Montage. Im folgenden Abschnitt wird es zusätzlich mitarbeiterspezifisch ausgeführt. Ein Schema zur Personalisierung von Informationsinhalten für die Produktion wird entwickelt und auf das arbeitsplatzspezifische Informationsangebot für die Montage von Schaltschränken angewandt. Das Personalisierungsschema muss auf den üblichen Konzepten aufsetzen, die Portale als Teil ihrer Basisdienste anbieten. Daher werden die bekannten Möglichkeiten und Formen der Personalisierung von Informationen zunächst kurz vorgestellt.

5.3.1 Grundlagen der Personalisierung von Informationsangeboten

5.3.1.1 Benutzermodelle und Profilierung

In der Informationstechnologie bedeutet Personalisierung die Anpassung der Benutzungsoberfläche und des Systemverhaltens an die individuellen Fähigkeiten und Präferenzen des Benutzers [Mertens04, S. 4; Bruse05, S. 79; Ziemer04, S. 58]. Dabei bedarf es eines Benutzermodells zur Erfassung, Formalisierung und Speicherung aller für das Dialogverhalten relevanten Annahmen und Informationen über den Benutzer [Essing01, S. 5]. Personalisierung und Benutzermodelle sind zusammengehörige Konzepte, die in vielen Facetten ausgeprägt sein können, s. Morphologie in Abbildung 5-14.

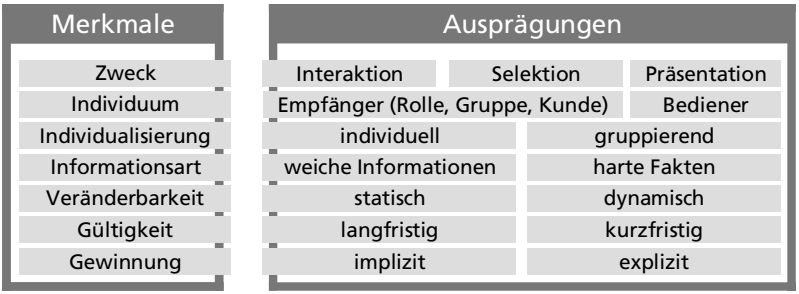


Abbildung 5-14: Formen der Personalisierung und Benutzermodellierung. Nach: [Mertens04, S. 5]

Ein Benutzermodell besteht zum *Zweck* der Anpassung des Interaktionsverhaltens, der Datenselektion und der Inhaltspräsentation. Das *Individuum* der Personalisierung ist überwiegend der Benutzer, der das System bedient und nach dessen Präferenzen das Informationsangebot angepasst wird. Die Personalisierung kann aber auch für die Empfänger der Informationen erfolgen, z. B. bei personalisierten Serienbriefen. Die *Individualisierung* gibt an, ob Benutzermodelle erstellt werden für jeden Benutzer oder für Benutzergruppen, die eine gleichartige Personalisierung erfahren. Die berücksichtigten *Informationsarten* der Präferenzen können auf harten Fakten beruhen, z. B. dem Alter des Benutzers, oder aus weichen Informationen bestehen, z. B. Annahmen über Ziele und Pläne des Benutzers. Die *Veränderbarkeit* und die *Gültigkeit* beziehen sich auf die Dauer einer Benutzersitzung. Dynamische Präferenzen werden im Laufe einer Benutzersitzung angepasst, statische Präferenzen bleiben unverändert. Kurzfristige Präferenzen bestehen nur

für eine Benutzersitzung, langfristige werden gespeichert und gelten für mehrere Sitzungen. Die *Gewinnung* des Benutzerprofils erfolgt entweder explizit durch Eingabe der Präferenzen oder sie erfolgt implizit durch Auswertung der Systeminteraktionen. Damit verbunden ist die Unterscheidung in *fremdbestimmte* oder *selbstbestimmte* Benutzermodelle. Selbstbestimmte Modelle sind explizit vom Benutzer erstellt. Fremdbestimmte Modelle sind entweder implizit gewonnen oder explizit durch Administratoren vorgegeben. Der Benutzer kann fremdbestimmte Profile häufig selbstbestimmt anpassen [Bach05, S. 41].

Entsprechend den vielfältigen Ausführungsformen der Personalisierung sind Benutzermodelle datenintensive und stark strukturierte Informationsobjekte. Typischerweise bestehen Benutzermodelle aus Benutzerdaten, Benutzungsdaten und Umgebungsdaten [Kobsa01, S. 115ff; Bach05, S. 158]. *Benutzerdaten* beschreiben an die Person gebundene, persönliche Eigenschaften des Benutzers. Hier finden sich Fakten über Alter, Beruf, Geschlecht, Fähigkeiten und Kenntnisse. *Benutzungsdaten* ergeben sich durch Beobachten und Protokollieren des im Rahmen der Datenschutzgesetze aufgezeichneten und ausgewerteten Benutzerverhaltens, z. B. von besuchten Seiten, abgerufenen Informationen, Navigationspfaden etc. [Großmann05, S. 94f; Schulz04, S. 31; Bruse05, S. 82]. Beobachtbare Benutzungsdaten können direkt in die Personalisierung eingehen. Weiterhin lassen sie auf Benutzungsgewohnheiten schließen, z. B. Aktionsfolgen und bevorzugte Aktionen in einer gegebenen Situation. *Umgebungsdaten* liefern Informationen zu der verwendeten Hardware- und Softwareplattform sowie zu den äußeren Bedingungen der lokalen Umgebung, in der die Systemnutzung erfolgt. Unterschiede bestehen beispielsweise zwischen dem Einsatz von mobilen oder stationären Endgeräten.

Benutzerdaten	Benutzungsdaten	Umgebungsdaten
• Demographische Daten • Wissen & Kenntnisse • Fähigkeiten, Fertigkeiten • Interessen, Vorlieben • Ziele, Pläne	**Beobachtbare Benutzung** • Navigation, Selektion • Verweildauer bei Inhalten • Bewertungen, Empfehlungen • Abrufe, Käufe • Zustimmungsbekundung *Ausdrucken, Bookmarken von Seiten* **Benutzungsgewohnheiten** • Nutzungshäufigkeit, -frequenz • Situations-Aktionskorrelation *Antworten auf neue email* • Aktionsfolgen	**Software-Plattform** • Browser-Produkt & -version • Plug-ins, Skripte, Cookies etc. **Hardware-Fähigkeiten** • Bandbreite Netzwerk • Display-Größe & Auflösung • Eingabegeräte **Lokalisation** • Ort • Umgebung, Geräusche

Abbildung 5-15: Bestandteile eines Benutzermodells. Nach: [Kobsa01, S. 115ff]

Benutzungsmodelle können neben individuellen Präferenzen auch formale Aufgaben und Nutzungsrechte berücksichtigen. Die Beziehungen zwischen Aufgaben, hierfür benötigten Informationsobjekten und den erforderlichen Nutzungsrechten werden üblicherweise mittels expliziten *Rollenprofilen* administriert [Bomsdorf04, S. 117]. Aus den Nutzungsrechten definieren sich Hierarchien von starken und schwachen Rollen, wobei stärkere Rollen alle Nutzungsrechte schwächerer Rollen übernehmen. Teilweise werden Rollenhierarchien noch mit Negativrechten ergänzt, die den Ausschluss von einzelnen Rechten für übergeordnete Rollen ermöglichen [Bomsdorf04, S. 120]. Einen ähnlichen Ansatz verfolgen *Stereotypen*. Sie beschreiben anhand von charakteristischen Merkmalen Benutzerklassen mit voreingestellten Personalisierungsfunktionen [Kobsa01, S. 125]. Die Klassen

können wie Rollen hierarchisch geordnet sein. Jedem Benutzer wird das Stereotyp zugeordnet, dessen Merkmale am ehesten seinen Eigenschaften entsprechen. Bei hierarchischen Stereotypen ist die Wurzelklasse ein Standardtyp für neue oder gelegentliche Benutzer. Sind aus einer intensiveren Nutzung weitere Daten verfügbar, lassen sich feinere Stereotype erkennen und zuordnen [Essing01, S. 9].

5.3.1.2 Gegenstände der Personalisierung

Gegenstände der Personalisierung sind die dargebotenen Informationsinhalte, die visuelle Gestaltung und die Navigationsstruktur [Röhricht01, S. 172; Kobsa01, S. 113]. Zur Umsetzung existiert eine Fülle von Gestaltungsmöglichkeiten, siehe hierzu [Kobsa01, S. 135ff; Thiels08, S. 56f]:

- Typische Informationen, die einer Personalisierung unterliegen, sind neue, geänderte, erläuternde oder detaillierende Inhalte [Oberbichler06, S. 44]. Diesbezügliche Informationsinhalte sind zu klassifizieren, zu selektieren und den Bezugsobjekten beizustellen. Zusätzlich können Empfehlungen, Bewertungen und Hinweise auf ähnliche Informationen angeboten werden.
- Hinsichtlich der Präsentation sind insbesondere die Informationsdichte und die Modalität wesentliche Gestaltungselemente. Als Modalitäten für Intranetanwendungen sind Text, Bild und Videoformate möglich, die mit weiteren Formatierungen und Auszeichnungsmerkmalen kombinierbar sind.
- Zur Anpassung der Navigationsstruktur lassen sich Verweise umsortieren, deaktivieren, ausblenden oder um ähnlich gelagerte Referenzen ergänzen.

Weitere mittelbare Gegenstände der Personalisierung sind die Bildung von Interessengruppen, sogenannte „Communities of practice“ [Oberbichler06, S. 43], sowie die Suche nach Experten oder ähnlichen Vorgängen.

5.3.1.3 Verfahren der Personalisierung

Personalisierte Informationsangebote entstehen in einem mehrstufigen Prozess. Notwendige Schritte sind die Akquisition und Extraktion eines Benutzermodells, gefolgt von einer Generierung des personalisierten Inhalts [Kobsa01, S. 114]. Aufgabe der Akquisition ist die Festlegung der Personalisierungsmöglichkeiten und eines formalen Modells, die Identifikation der erforderlichen Informationen und Datenquellen sowie die Erhebung der initialen Daten. Im Zuge der Extraktion erfolgen eine Bereinigung der initialen Daten und gegebenenfalls eine sekundäre Inferenz von weichen Benutzerpräferenzen. Akquisition und Extraktion des Benutzermodells werden gemeinsam als Profilierung bezeichnet [Ziemer04, S. 60]. Der Generierungsschritt erstellt den personalisierten Inhalt entsprechend den Präferenzen des Benutzermodells.

Ausgehend von der Morphologie lassen sich *adaptierbare* von *adaptiven* Formen der Personalisierung unterscheiden. Adaptierbare Systeme verwenden vom Benutzer oder Administrator explizit vorgegebene, statische, auf harten Fakten beruhende und langfristig geltende Präferenzen. Eine adaptive Personalisierung erfolgt durch das System unter Anwendung von Inferenzmechanismen und wird z. B. für Empfehlungssysteme, Relevanzsuche oder die kollaborative Filterung auf Basis von Cliquen eingesetzt [Bruse05, S. 82; Kobsa01, S. 113; Schulz04, S. 32]. Adaptive Systeme sind dynamisch, verwenden implizite Präferenzen und berücksichtigen auch weiche Informationen. Mischformen von adap-

tierbarer und adaptiver Personalisierung sind in diversen Anwendungen realisiert [Kobsa01, S. 113].

Nach der Vorstellung der Grundlagen wird anschließend ein personalisiertes, mitarbeiterspezifisches Informationsangebot unter Einsatz der neuen Entwurfsmustersprache entwickelt. Verwendet wird ein Verfahren, das auf Stereotypen basiert, adaptierbare mit adaptiven Ansätzen kombiniert und eine Filterung der dargebotenen Informationsinhalte vorsieht.

5.3.2 Analyse des produktionsnahen Personalisierungsbedarfs

5.3.2.1 Personalisierungsmuster

In einem ersten Schritt ist der produktionsnahe Personalisierungsbedarf zu ermitteln. Die zu betrachtenden Elemente zeigt das Analysemuster PERSONALIZED CONTENT, s. Abbildung 5-16. Das Muster geht aus vom typischen Aufbau von interaktiven Webanwendungen und Portalen, die Daten bzw. Content von Präsentation und Navigation trennen und erst zur Laufzeit kombinieren. Navigiert ein Benutzer zu einem gesuchten Informationsangebot, wird ein passendes Präsentationstemplate von der Portal-Engine dynamisch mit dem ausgewählten Content befüllt und auf dem Front-End-Browser angezeigt. Diesen grundlegenden Ablauf erweitert das Muster PERSONALIZED CONTENT mit einem obligatorischen Benutzerprofil, einem Contentprofil und einem Personalisierungsschema.

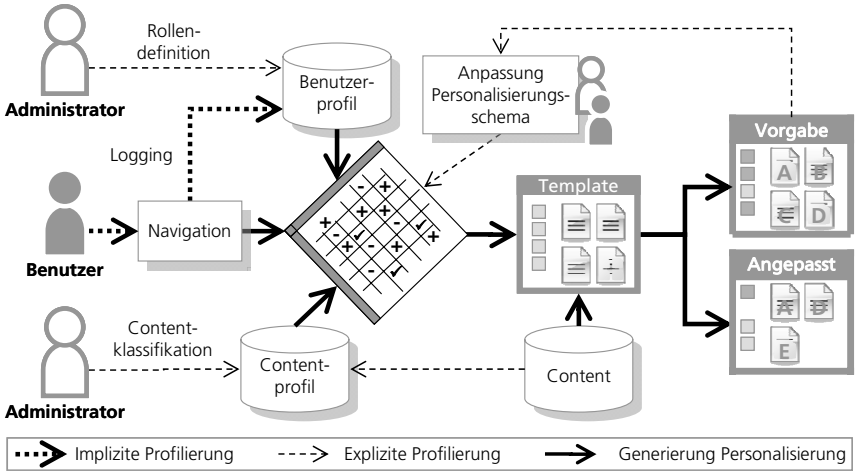


Abbildung 5-16: Analysemuster Personalized Content. Nach: [vanDuyne02, S. 272]

Das Contentprofil charakterisiert die auszugebenden Informationen und deren Anpassungsmöglichkeiten. Ein Personalisierungsschema kreuzt das Benutzerprofil mit dem Contentprofil und ordnet ihnen die gewünschten Anpassungen für Content zu. Bei der Befüllung eines Templates wird vom Portal das Personalisierungsschema berücksichtigt. Über das Profil des anzuzeigenden Contents und das Profil des angemeldeten Benutzers sind aus dem Profilierungsschema die vorzunehmenden Anpassungen ersichtlich. Neue Benutzer erhalten zunächst eine „Vorgabepersonalisierung“. Sie können die Anwendung sofort

und ohne umfangreiche Vorarbeiten einsetzen. Anschließend passen die Benutzer oder Administratoren das Personalisierungsschema an. Soll zusätzlich eine implizite Profilierung stattfinden, ist das Navigationsverhalten des Benutzers zu protokollieren. Das flexibel gehaltene Muster ist sowohl für explizite als auch implizite Profilakquisition einsetzbar und erlaubt adaptierbare Verfahren ebenso wie adaptive.

Ausgehend von diesem allgemeinen Muster zur Personalisierung ist der Aufbau des Benutzerprofils, des Contentprofils und des Personalisierungsschemas für den Einsatz im produktionsnahen Umfeld zu entwickeln.

5.3.2.2 Benutzerprofil

Bei einem Einsatz von Informationsportalen im produktionsnahen Bereich sind die Umgebungsdaten bekannt und üblicherweise homogener als in anonymen Internetumgebungen. Die Benutzer sind eindeutig identifizierbar, demografische Benutzerdaten liegen vor, Ziele und Pläne sind durch die Arbeitsaufgaben vorgegeben, persönliche Interessen und Vorlieben sind nur von untergeordneter Bedeutung. Hinsichtlich des Wissens, der Fähigkeiten und Kenntnisse kann unterstellt werden, dass die Mitarbeiter für reguläre Kernaufgaben hinreichend geschult, sowie mit den generellen Arbeitsbedingungen und der grundlegenden Arbeitsumgebung vertraut sind. Typischerweise wird der Schulungsstand der Mitarbeiter in Qualifikationsmatrizen expliziert [Clement06, S. 106]. Infolge des typischen Arbeitswechsels können jedoch variierende Routinegrade und Informationsstände für einzelne Arbeitsstation vorliegen, die eine angepasste Gestaltung des Informationsangebots erfordern, vgl. Kapitel 2.3.1.5. Im Benutzerprofil sind daher Benutzungsdaten bezüglich der Routine eines Benutzers an den einzelnen Arbeitsstationen abzubilden. Dabei sind die einschlägigen Datenschutzgesetze und Vorschriften einzuhalten.

Um eine Pflege der Benutzerprofile und Vorgaben auch für eine größere Anzahl von Benutzern effizient zu gestalten, ist eine Gruppenbildung vorzusehen [Bruse05, S. 121]. Ein Konzept zur Kategorisierung von Benutzern ist die Stereotypisierung [Kobsa01, S. 125]. Stereotypen verbinden Annahmen über Benutzereigenschaften mit Bedingungen für die Gruppenzuordnung. Ein Benutzer entspricht einem Stereotyp, falls sein Benutzerprofil die Zuordnungsbedingungen erfüllt. Die möglichen Stereotype sind üblicherweise vorgegeben. Zur Personalisierung von Kerninformationen wird eine Stereotypisierung vorgeschlagen nach den Kriterien Tätigkeitsspektrum, Einsatzhäufigkeit und Zeitraum seit letzter Ausführung. Mit diesen Kriterien lassen sich folgende fünf Routinestereotypen definieren, s. Abbildung 5-17:

- Tätigkeitsfremd an einer Arbeitsstation sind Mitarbeiter, die einen Mindesttätigkeitsumfang nicht beherrschen, gemäß ihrer Qualifikationsmatrizen.
- Eingewiesene Mitarbeiter beherrschen den Mindesttätigkeitsumfang und weisen die erforderliche Einarbeitungszeit auf. Unterstellt wird das in der Produktion verbreitete Training-on-the-job-Verfahren.
- Gelegentliche Mitarbeiter sind eingewiesen, haben eine gewisse Mindestanzahl an Einsatztagen absolviert und der letzte Einsatz fand innerhalb eines vorgegebenen Zeitraums statt.
- Routinierte und permanente Mitarbeiter erfüllen die gleichen Kriterien, an die allerdings höherer Ansprüche gestellt werden.

Die in Abbildung 5-17 beispielhaft dargestellten Werte entsprechen Anforderungen an angelernte Montagetätigkeiten. Je nach Tätigkeitsumfang und erforderlichem Qualifikationsniveau können die Werte zwischen verschiedenen Arbeitsstationen beträchtlich abweichen. Sie sind für jede Arbeitsstation bzw. Klassen von Arbeitsstationen gesondert festzulegen.

Stereotype	Routineschwelle je Station		
	Beherrschte Tätigkeiten **%**	Ausführungs-häufigkeit **Arbeitstage**	Letzte Ausführung **Arbeitstage**
Tätigkeitsfremd	**< 80**		
Eingewiesen	**> 80**	**> 2**	
Gelegentlich	**> 80**	**> 10**	**< 240**
Routiniert	**> 80**	**> 20**	**< 60**
Permanent	**> 80**	**> 40**	**< 20**

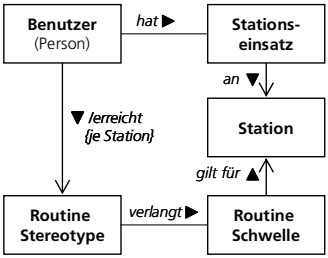


Abbildung 5-17: Routine-Stereotype für die Benutzerprofilierung

5.3.2.3 Contentprofil

Zur Erstellung eines Schemas für Contentprofile ist der Informationsraum zu betrachten, s. Tabelle 2-1. Die vorrangig zu personalisierenden Kerninformationen stammen aus dem Informationscluster Auftragsabwicklung. Verwendet werden neben Produktunterlagen viele tabellarische und textbasierte Informationen, z. B. für Stücklisten, Auftragsspezifikationen, Änderungsmitteilungen, Arbeitspläne, Arbeits- und Prüfanweisungen etc. Bei allen, typischerweise einer Personalisierung unterliegenden, ergänzenden und detaillierenden Informationen dominieren alphanumerische Daten und textbasierter Content.

Für das Contentprofilschema werden zwei Entwurfsmuster herangezogen, die textbasierte Informationen charakterisieren. DOCUMENT SUBJECT zeigt die datenbankgestützte Klassifikation von Dokumenten mit standardisierten Intentionen. INVERSE PYRAMID WRITING STYLE besagt, Textbeiträge in einem mehrgliedrigen Schema zu erstellen, bestehend aus zumindest einer Titelzeile, einer Zusammenfassung und einer ausführlichen Beschreibung. In Verbindung mit einer visuell prägnanten Darstellung der Gliederung erleichtert das Muster dem Anwender die Beurteilung der Informationsrelevanz und verkürzt die Perzeptionszeit. Beide Entwurfsmuster werden zu einem Schema für die Contentprofilierung von Kerninformationen ausgebaut, das zwei Klassifikationsaspekte Intention und Paragraph verwendet. Die Intention benennt den Zweck einer Information und sieht folgende Klassen vor:

- Standards enthalten variantenneutrale Informationen bezüglich obligatorischer Auftragsteile;
- Optionen kennzeichnen variantendefinierende Informationen bezüglich optionaler Auftragsumfänge;
- Instruktionen stehen für anleitende Inhalte, z. B. Arbeitsanweisungen;
- Hinweise ergänzen die bestehenden Informationen um Tipps oder aktuelle Vorgänge;
- Warnungen betreffen Aktivitäten oder Ereignisse, die sich störend auswirken könnten;
- Gefährdungen weisen auf sicherheitsrelevante Vorkehrungen und einzuhaltende Maßnahmen hin;

- Fragen betreffen aktuelle Aufgaben oder Vorschläge zur Produktionsumgebung und zu den Prozessen, z. B. im Rahmen von KVPs.;
- Probleme zeigen temporäre Störungen und Qualitätsdefekte im Produktionsprozess an, die ungelöst oder von unbekannter Ursache sind;
- Private Informationen hinterlegen Anwender zu beliebigen Objekten und Daten als ergänzende Informationen, z. B. für persönliche Merkhilfen.

Der zweite Klassifikationsaspekt, Paragraphen, beschreibt die möglichen Gliederungsabschnitte einer Informationseinheit und sieht folgende Abschnitte vor:

- Icons sind Piktogramme und kennzeichnen den Typ bzw. die Intention;
- Titel benennen eine Informationseinheit mit einer aussagekräftigen Überschrift;
- Abstracts enthalten kurze Zusammenfassungen der Informationen;
- Beschreibungen führen den Sachverhalt aus;
- Thumbs sind charakteristische Bilder der Gegenstände oder Sachverhalte;
- Begründungen erklären den Sachverhalt und weisen auf die Ursachen hin;
- Beispiele zeigen die Sachverhalte an konkreten Objekten;
- Hintergründe führen zu weiteren, Verständnis fördernden Informationen.

Abbildung 5-18 zeigt die Entwurfsmuster, die daraus abgeleiteten Klassifikationsaspekte und die Icons zur Visualisierung der Intention.

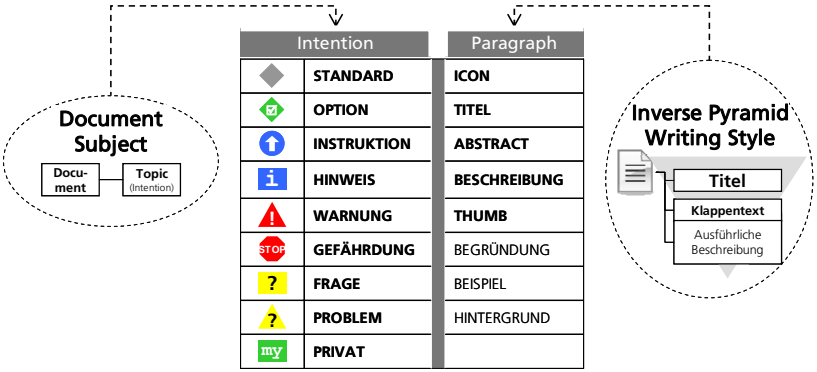


Abbildung 5-18: Analysemuster zur Contentprofilierung

Das Contentprofilschema ist abgestimmt auf die regulären Kerninformationen in der Produktion, und bei Bedarf einfach zu erweitern. Ebenso können Klassen ergänzt werden für weitere Gegenstands- und Aufgabenbereiche. Zum Beispiel sind für die Materialwirtschaft und den Einkauf Intentionen der Art <Anfrage> <Rückfrage> <Bestellung> <Mahnung> vorstellbar.

5.3.2.4 Personalisierungsschema

Ein Personalisierungsschema legt den Grad und die Form der Anpassung des Informationsangebots für einen Benutzer oder eine Benutzergruppe fest. Dabei ist darauf zu achten, ausufernde, die Mitarbeiter ablenkende Möglichkeiten der Personalisierung zu vermeiden. Eine zu große Varianz des Informationsangebots, dessen Aufbaus und visueller

Gestaltung kann im produktionsnahen Umfeld den routinierten Umgang behindern, eine gegenseitige Unterstützung durch Kollegen erschweren und dadurch kontraproduktiv wirken [Thiels08, S. 55]. Daher beschränkt sich das hier entwickelte Personalisierungsschema auf die Auswahl und Filterung der Informationsinhalte. Weitergehende Möglichkeiten zur Personalisierung des Informationsangebots, z. B. hinsichtlich des Layouts oder der Modalität sind nicht enthalten.

Das Personalisierungsschema sieht vor, den anzuzeigenden Informationsumfang auf die Routine Stereotype abzustimmen. Für jedes Routine-Stereotyp wird tabellarisch aufgelistet, welche Paragraphen einer Informationseinheit zu einer gegebenen Intention anzuzeigen sind, s. Tabelle 5-4. Zusätzlich werden neue und bekannte Informationen unterschieden, und so der anzuzeigende Informationsumfang in Abhängigkeit des Kenntnisstands des Mitarbeiters variabel gestaltet.

	Anzeigeobjekt: Montagereihenfolge			
Intention	Permanent	Routiniert	Gelegentlich	Eingewiesen
Standard	N: **T**	N: **T, A**	N: **T, A** B: T, A	N: **T, B, TH** B: T, B, TH
Option	N: **T** B: T	N: **T, A** B: T	N: **T, A** B: T, A	N: **T, B, TH** B: T, B, TH
Instruktion	N: **I**	N: **T** B: I	N: **T, A** B: T	N: **T, B, TH** B: T, B, TH
Hinweis	N: **I**	N: **T** B: I	N: **T, A** B: T	N: **T, B, TH** B: T, B, TH
Warnung	N: **T** B: I	N: **T, A** B: T	N: **T, A, TH** B: T, A	N: **T, B, TH** B: T, B, TH
Gefährdung	N: **T, A, TH** B: I	N: **T, B, TH** B: T	N: **T, B, TH** B: T, B, TH	N: **T, B, TH** B: T, B, TH
Frage	N: **I**	N: **T, A** B: I	N: **T, A** B: T, A	N: **T, B, TH** B: T, B, TH
Problem	N: **I**	N: **T, A** B: I	N: **T, A, TH** B: T, A	N: **T, B, TH** B: T, B, TH
Privat	N: I B: I	N: T, A B: T, A	N: T, A, TH B: T, A, TH	N: T, B, TH B: T, B, TH
Prinzip der Personalisierung	Nur notwendige Info, Hinweise auf neue Info	Alle neue Info, Hinweise auf bekannte Info	Gesamte Info, verkürzte Anzeige	Ausführliche Anzeige der gesamten Info

Paragraphen	
I	ICON
T	TITEL
A	ABSTRACT
B	BESCHREIBUNG
TH	THUMBNAIL
N	Neue Info, noch nicht gelesen
B	Bekannte Info, bereits gelesen

Tabelle 5-4: Personalisierungsschema für Kerninformationen

Das in Tabelle 5-4 beispielhaft dargestellte Personalisierungsschema folgt der Maxime der „5S - Informationsbereitstellung", die besagt, Informationsangebote auf die notwendigen Inhalte zu beschränken. Weiterführende und detaillierende Informationen sind für routinierte Mitarbeiter verzichtbar. Ein permanent an einer Station eingesetzter Mitarbeiter erhält Informationen zu variantenneutralen Standardtätigkeiten nur angezeigt, wenn diese für ihn neu sind, s. Eintrag in der obersten Zeile der linken Spalte. Optionale Umfänge werden kompakt angezeigt mit ihrem Titel, ohne weitere Beschreibungen. Einem erst eingewiesenen Mitarbeiter werden dagegen immer alle verfügbaren Informationen angeboten, sowohl zu Standardtätigkeiten als auch zu optionalen Umfängen, unabhängig davon, ob er die Informationen bereits zur Kenntnis genommen hat. Zudem bekommt er nicht nur den Titel der Informationseinheit angezeigt, sondern zusätzlich noch die Beschreibung eingeblendet und ein eventuell vorhandenes Miniaturbild ausgegeben. Offensichtlich werden mit steigendem Routinegrad sukzessive Informationen ausgeblendet. Dafür nehmen im Gegenzug die Informationsdichte und die Übersichtlichkeit zu. Das gezeigte Schema dient im nächsten Abschnitt zur Personalisierung des Informationsangebots bezüglich der Montagereihenfolge.

Neben der Erstellung sieht das Muster PERSONALIZED CONTENT die Pflege der Profile und Schemata vor, die den jeweiligen Benutzergruppen, dem Einsatzbereich und den verwendeten Templates anzupassen sind. Wie bei der Auslegung des Personalisierungsumfangs, sind im Produktionsumfeld ausufernde Möglichkeiten der Pflege und Administration der Profile zu vermeiden und die operativen Mitarbeiter von derartigen Aufgaben zu entlasten. Die Benutzerprofile basieren auf impliziten Verfahren und erfordern keine Eingriffe seitens der Anwender. Ihre Pflege beschränkt sich auf die Vorgabe der Routineschwellen je Station durch Administratoren. Das Personalisierungsschema ist abzustimmen auf die Templates und somit auf die Arbeitsstationen, an denen das Template verwendet wird. Es wird ebenfalls durch Administratoren gepflegt. Dabei kann die Rolle eines Administrators auch durch produktionsnahe Mitarbeiter wahrgenommen werden, z. B. in Person des Gruppensprechers oder Teamleiters.

5.3.3 Konzeption des mitarbeiterspezifischen Informationsangebots

Nach der Analyse des Personalisierungsbedarfs ist das Personalisierungsschema in dem bestehenden Informationsangebot für die Montage von Schaltschränken umzusetzen. Die Aufgabensequenz zum Ergänzen von Portalseiten sieht für die Konzeption von Informationsangeboten die Blöcke Objektnavigation, Seitendialog, Seitenlayout und Seitenverarbeitung vor. Das Personalisierungsschema wirkt unmittelbar auf die Auswahl der Informationsinhalte. Zur Personalisierung ist daher vorrangig der Seitendialog für den variablen Informationsumfang festzulegen. Die hierfür eingesetzten konzeptionellen Entwurfsmuster zeigt Abbildung 5-19.

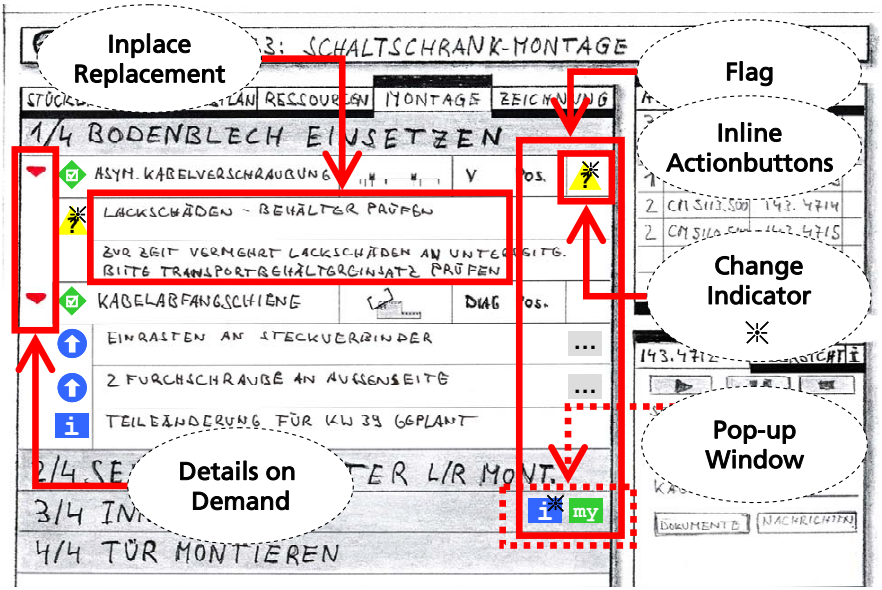


Abbildung 5-19: Seitennavigation im personalisierten Informationsangebot

Beim Seitendialog ist zu unterschieden zwischen den Arbeitsgängen und den Stücklistenpositionen, die in der Tabelle für die Montagereihenfolge angezeigt werden. Die Arbeits-

gänge dienen der Gliederung der Tabelle und sind nur mit ihrem Titel dargestellt. Vorhandene Zusatzinformationen werden daher mit Icons als FLAGS angezeigt und in POP-UP-WINDOWS geöffnet. Davon abweichend sind Stücklistenpositionen unterhalb der Arbeitsgänge als COLLAPSIBLE PANELS eingefügt. Die Anzahl der Zeilen ist variabel und ermöglicht die Verwendung des alternativen Entwurfsmusters INPLACE REPLACEMENT. Es besagt, Zusatzinformationen direkt unterhalt der jeweiligen Stücklistenposition in die Tabelle einzublenden. Beide Muster werden mit DETAILS-ON-DEMAND kombiniert, d. h., die Informationen werden nur auf Anforderung des Benutzers angezeigt und lassen sich mittels der kleinen dreieckigen Drehschalter ein- und ausblenden. Dabei folgt der Initialzustand dem Personalisierungsschema, s.Abbildung 5-20. Für routinierte Benutzer sind die meisten Zusatzinformationen zu Beginn ausgeblendet, und werden nur „on demand" eingeblendet. Gelegentliche oder erst eingewiesene Benutzer erhalten dagegen von Beginn an alle Informationen eingeblendet und ausführlich präsentiert.

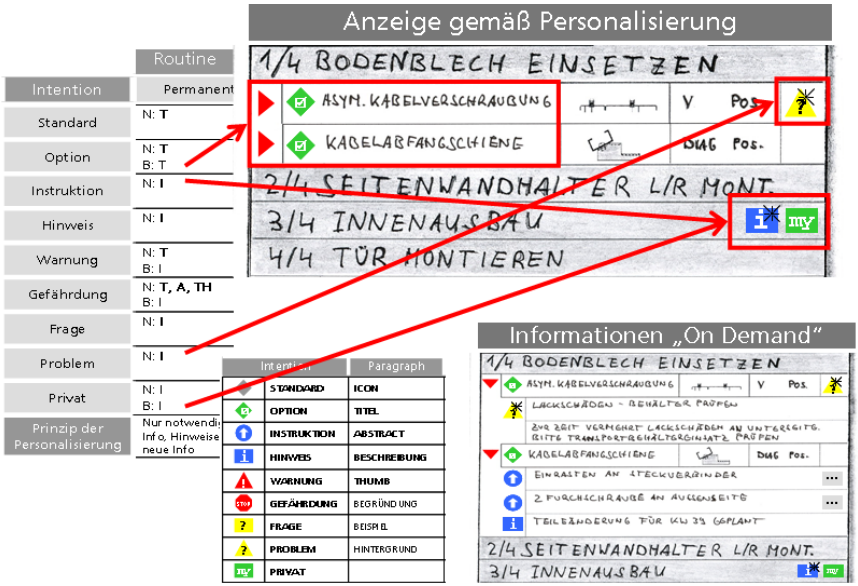


Abbildung 5-20: Präsentation der personalisierten Informationsinhalte

Weiterhin ist auf geänderte oder ergänzende Informationen hinzuweisen. Dargestellt wird der Hinweis als ein FLAG in Form des Intentions-Icons. Neue oder geänderte Inhalte erhalten einen CHANGE INDICATOR in Form eines kleinen Sterns auf dem FLAG. Alle FLAGS sind INLINE ACTIONBUTTONS, die entweder ein POP-UP-WINDOW öffnen oder das INPLACE REPLACEMENT auslösen. Ein INLINE ACTIONBUTTON ist mit dem Informationselement, auf das er sich bezieht, in einer Zeile dargestellt. Existieren mehrere Zeilen gleichartiger Informationselemente, verfügt jede Zeile über eigene INLINE ACTIONBUTTONS.

Das konzipierte Informationsangebot erfüllt die Anforderungen an eine arbeitsplatz- und mitarbeiterspezifische Informationsbereitstellung, wie in Kapitel 2.3.1 aufgestellt:

- Arbeitsplatzspezifische Kerninformationen bezüglich der Montagereihenfolge und der kundenindividuellen Auftragsoptionen sind auf das notwendige Minimum reduziert.
- Komplementärinformationen sind flankierend bereitgestellt und „ON DEMAND" direkt aus dem Kerninformationsangebot heraus erreichbar.
- Sporadische Notifikationsfunktionen sind im Kerninformationsangebot ersichtlich über CHANGE INDICATORS, die auf neue und geänderte Informationsstände hinweisen.
- Zur bidirektionalen Kommunikation sind Fragen, Problemberichte, Warnungen und Hinweise vorgesehen. Deren Ausgestaltung als Foren ist in der Mustersequenz in Kapitel 4 gezeigt. Zudem lassen sie sich als Annotationen den Stücklisten und Arbeitsgängen zuordnen und in der Tabelle für die Montagereihenfolge mit anzeigen. Beispielhaft zeigen Abbildung 5-19 und Abbildung 5-20 in der zweiten Zeile einen neuen Problembericht bezüglich aktuell auftretender Lackschäden.
- Die Informationen sind personalisierbar nach der Routine der Mitarbeiter und der anzuzeigende Informationsumfang ist steuerbar.

5.3.4 Implementierung einer mitarbeiterspezifischen Portalseite

Personalisierungsschemata verknüpfen Benutzerprofile mit Contentprofilen, und können für jedes Template erstellt oder für mehrere Templates gemeinsam benutzt werden. Sie müssen verwaltet und dauerhaft abgelegt werden und benötigen daher eine einheitliche Struktur. Bei der Implementierung stellen sich mehrere Entwurfsfragen:

- Der Content entstammt externen Informationsquellen. Er kann nur gelesen und aufbereitet, jedoch nicht verwaltet und angereichert werden.
- Die Intentionen betreffen heterogene Objekt- und Datenstrukturen. So könnten beispielsweise Instruktionen in Form von ERP-Arbeitsplänen vorliegen, Hinweise als ergänzende Textdokumente verfasst sein und Problemberichte im CAQ-System anfallen.
- Bei der Präsentation der Inhalte ist der Informationsstand des Anwenders bezüglich der externen Informationsquellen zu berücksichtigen.

Um die Generierung des personalisierten Inhalts möglichst zentral implementieren zu können, ist das Personalisierungsschema von den primären Daten- und Informationsquellen zu entkoppeln. Abbildung 5-21 zeigt die eingesetzten Entwurfsmuster und deren Aufgaben.

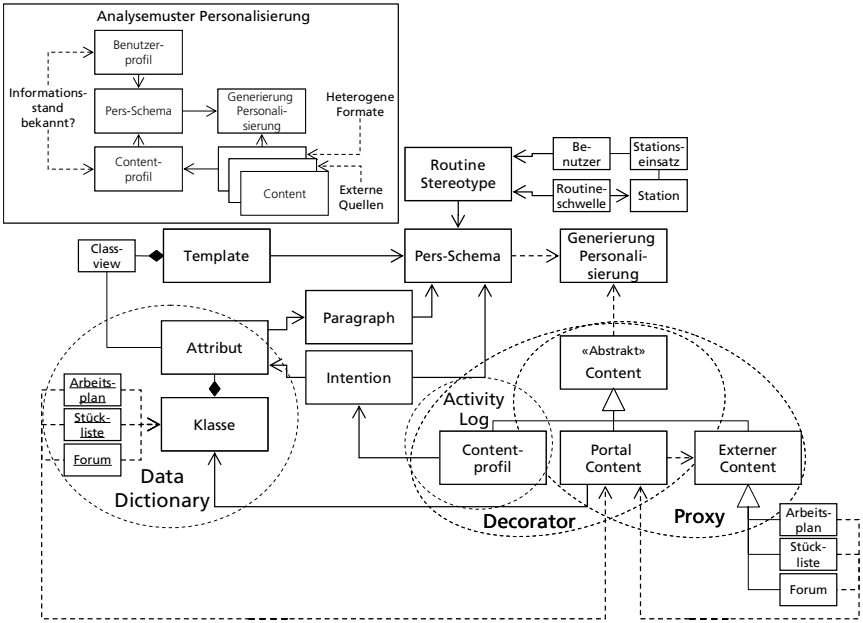


Abbildung 5-21: Implementierung der Personalisierung mit Entwurfsmustern

Der Kern der Lösung besteht aus einem DATA DICTIONARY, das die verwendeten externen Datenquellen explizit beschreibt. Für jede externe Contentklasse wird ein Eintrag im DATA DICTIONARY angelegt, der die Contentklasse, ihre Attribute, Eigenschaften und Verwendung repräsentiert. In dem im letzten Abschnitt konzipierten, personalisierten Informationsangebot für die Montagereihenfolge wurden die Contentklassen „Arbeitsplan", „Stückliste" und „Forum" verwendet, und sind in Abbildung 5-21 stellvertretend als Objekte des DATA DICTIONARIES dargestellt. Die weiteren Objekte des Personalisierungsschemas können nun einheitlich auf Eigenschaften der Contentklasse zugreifen und z. B. auswerten, welche Intention sich mit einer externen Datenquelle verbindet. Zur weiteren Entkopplung werden die externen Informationsquellen mit einem PROXY Objekt intern einheitlich repräsentiert. Somit setzt die Generierung der Personalisierung ebenfalls auf einem einheitlichen Datenobjekt auf und kann, wie gewünscht, zentral realisiert werden. Die interne Darstellung des externen Contents benötigt ebenfalls die Kenntnis des Aufbaus der externen Informationsquellen aus dem DATA DICTIONARY. Als letzte Maßnahme wird das Contentprofil in einer Klasse gekapselt und als ein DECORATOR dem PROXY beiseite gestellt. Der Contentprofilklasse wird zudem die Aufgabe übertragen, neue bzw. geänderte Informationsinhalte anzuzeigen. Dazu erhält die Klassen einen ACTIVITY LOG, aus dem ersichtlich ist, welche Informationsobjekte ein Benutzter wann zum letzten Mal gelesen hat.

5.3.5 Mustersequenz des mitarbeiterspezifischen Informationsangebots

Zur Realisierung des mitarbeiterspezifischen Informationsangebots wurden zwölf Entwurfsmuster eingesetzt, die wiederum eine lösungsorientierte Mustersequenz bilden, s. Tabelle 5-5.

Aufgabe	Aufgabensequenz „Portalseiten ergänzen“	Aufgabensequenz „Portal-Engine erstellen“
Analyse		
Personalisierung		**PERSONALIZED CONTENT** Ablauf der Personalisierung **INVERSE PYRAMID WRITING, DOCUMENT SUBJECT** Contentprofil
Konzeption		
Seitendialog	**DETAILS ON DEMAND** Anzeige ergänzender Informationsinhalte nur auf Anforderung **INPLACE REPLACEMENT** Ergänzende Informationen einblenden **POP-UP WINDOWS** Ergänzende Informationen in einem Zusatzfenster	
Seitenlayout	**INLINE ACTIONBUTTONS** Buttons in jeder Tabellenzeile **FLAG** Hinweis auf ergänzende Information **CHANGE INDICATOR** Anzeige geänderter Informationsstand	
Implementierung		
Spezielle Anwendungen	**PROXY** Entkopplung des externen Contents	
Unterstützende Funktionen		**ACTIVITY LOG** Monitoring von Informationsständen
Datenmapping	**DECORATOR** Verbindung des Contentprofils mit dem externen Content	
Datenhandling		**DATA DICTIONARY** Abstraktion von externen Contentklassen

Tabelle 5-5: Mustersequenz zur Personalisierung der Portalseite „Montageoptionen“

Eine Personalisierung sollte im Portal einheitlich sein und ist in der Portal-Engine zu verankern. Demzufolge entstammen die Muster für die Profilbildung und die Implementierung der Aufgabensequenz „Portal-Engine erstellen“. Das bestehende Informationsangebot wurde angepasst durch die Anwendung weiterer Muster der Aufgabensequenz „Portalseiten ergänzen“, hier am Beispiel der Montagereihenfolge dargestellt. Die konzipierte Form der Personalisierung erfordert Muster aus beiden Aufgabensequenzen. Sie ist bei der Erstellung der Portal-Engine vorzubereiten und im Zuge der Erstellung von Portalseiten zu vollenden. Jede Portalseite mit einem personalisierten Informationsangebot der

Form muss bei der Realisierung der Seite einen Seitendialog und ein Seitenlayout gestalten sowie die vorgegebenen Implementierungsmuster PROXY und DECORATOR umsetzen. Die Portal-Engine stellt dafür die Basisdienste und grundlegende Konzepte zur Personalisierung bereit. Die Mustersequenz beschreibt somit zugleich eine Teilung der Verantwortlichkeiten zwischen den Aufgabensequenzen „Portalseiten ergänzen“ und „Portal-Engine erstellen“.

Die gezeigte, aufwendige Form der Personalisierung ist dem Versuch geschuldet, die ergänzenden Informationen möglichst eng mit den Kerninformationen zu verzahnen, flankierend bereitzustellen und gezielt auf Änderungen im Informationsstand hinzuweisen. Ein Verzicht auf diese enge Kopplung könnte die Mustersequenz einfacher halten. So ist zum Beispiel CONTENT MODULES ein in vielen Internetportalen anzutreffendes Muster zur Personalisierung von Portalseiten, die kleine, unabhängige Abschnitte mit wählbaren Inhalten enthalten. Bei Anwendung dieses Musters könnten immer noch Hinweise, Problemberichte etc. mit der Montagereihenfolge auf einer Seite stehen. Allerdings in eigenen Anzeigeblöcken. Direkt ersichtliche Bezüge zur Montagereihenfolge, zu einem Arbeitsgang oder einer Stücklistenposition wären nicht mehr darstellbar.

5.4 Übertragbarkeit der Mustersequenzen

In der Arbeit wurden insgesamt drei lösungsorientierte Mustersequenzen erstellt, die einen häufig anzutreffenden Informationsbedarf decken und auf viele Einsatzfälle übertragbar sind unter folgenden Voraussetzungen:

- Die Mustersequenz für ein arbeitsplatzspezifisches Kerninformationsangebot bezüglich der Montagereihenfolge ist übertragbar, wenn für die Anweisungen und Optionen eine definierte Reihenfolge ermittelt werden kann, die Informationsinhalte tabellarisch darstellbar und vom Umfang begrenzt sind.
- Die Mustersequenz zur Personalisierung ist durch den Austausch der Routinestereotypen und die Intentionen des Contentprofils auf weitere Einsatzfälle anpassbar. Voraussetzung ist, dass die Intentionen der benutzten Informationen definiert sind und nur eine begrenzte Anzahl von Intentionsklassen vorliegt. Infrage kommen somit vorwiegend Routinetätigkeiten mit einem hohen Grad an Wiederholung und Standardisierung.
- Die Mustersequenz zur asynchronen Benachrichtigung ist geeignet für alle Einsatzfälle, bei denen kurze und verhältnismäßig einfach strukturierte Nachrichten oder Anmerkungen ausgetauscht oder dokumentiert werden sollen und ein hoher Bezug zu einem Gegenstand der Auftragsabwicklung oder des Produktionsumfelds gegeben ist.

Im hier betrachteten Gegenstandsbereich der diskreten Produktion von Realgütern dürften die Voraussetzungen häufig gegeben sein. Zudem ist das Implementierungsschema unabhängig von Systemplattformen und lässt sich in vielen Portalwerkzeugen umsetzen. Somit kann eine breite Übertragbarkeit der Mustersequenzen unterstellt werden.

Mit der Entwurfsmustersprache und den Mustersequenzen liegen nun die elementaren Bausteine einer konzeptionellen Unterstützung vor, für den Aufbau von produktionsnahen Informationsportalen.

6 Evaluierung in der kundenindividuellen Produktion

Zielsetzung der vorliegenden Arbeit ist eine effiziente Informationsversorgung bei kundenindividueller Produktion. Informationsportale sind hierfür ein probates, in anderen Unternehmensbereichen bewährtes Instrument. Für die produktionsnahe Bereitstellung von operativen Informationen werden sie jedoch selten genutzt. Dementsprechend mangelt es an Erfahrungen und Lösungskonzepten bei der Gestaltung von arbeitsplatz- und mitarbeiterspezifischen Informationsangeboten. Die vorliegende Arbeit verfolgt als Lösungsansatz die Unterstützung der Portalentwicklung mittels einer Methode auf Basis einer Entwurfsmustersprache. Von dem Lösungsansatz wird insbesondere die bedarfsgerechte Ausführung und Gestaltung von produktionsnahen Informationsportalen erwartet.

Eine diesbezügliche Evaluierung der Lösung fand statt in der Endmontage bei vier Unternehmen, die kundenindividuelle Investitionsgüter herstellen. Im nächsten Abschnitt wird das industrielle Umfeld der Evaluierung vorgestellt, die derzeitige Situation der Informationsnutzung beleuchtet und die Einsatzvoraussetzung von Informationsportalen in der Produktion geklärt. In den beiden darauf anschließenden Abschnitten werden zwei produktionsnahe Informationsportale vorgestellt, die als Grundlage zur Evaluierung des Lösungsansatzes herangezogen werden, dessen Bewertung im letzten Abschnitt erfolgt.

6.1 Industrielles Umfeld der Evaluierung

6.1.1 Charakterisierung der Unternehmen

Die Produkte der betrachteten Unternehmen weisen einen ausgeprägten Individualisierungsanteil aus. Deutliche Unterschiede bestehen in der Produktkomplexität, Fertigungstiefe und Integrationsart des Kunden in die Auftragsabwicklung. Insgesamt ergibt sich so ein breit gefächertes Bild der Anforderungen an eine produktionsnahe Informationsversorgung. Eine grobe Charakterisierung der betrachteten Unternehmen zeigt Tabelle 6-1.

	Integrationsart	Produkte	Produktion	Größe
Firma A	Engineer-to-order	Stromversorgung für Leistungselektronik	Kleinserien, verteilt auf längeren Zeitraum	Mittel
Firma B	Made-to-order, Assemble-to-order	Schaltschränke	Einzelaufträge, Kleinserien	Groß
Firma C	Assemble-to-order	Elektronische Messgeräte	Einzelaufträge	Mittel
Firma D	Engineer-to-order	Abfüllanlagen	Einzelaufträge	Groß

Tabelle 6-1: Charakteristik der untersuchten Unternehmen mit kundenindividueller Produktion

Unternehmen A stellt Stromumrichter vorwiegend für die Bahntechnik her. Die robusten und voluminösen Geräte werden kundenspezifisch entwickelt und in Kleinserien produziert, in Einzelfällen bis zu mehreren Hundert gleicher Geräte. Die Aufträge werden in Projektform abgewickelt. Bei umfangreichen Projekten zieht sich die Produktionsdauer der Geräte über mehrere Jahre.

Unternehmen B fertigt elektrische Schaltschränke. Dabei handelt es sich überwiegend um Katalogprodukte, die aus einem standardisierten Baukasten vom Kunden spezifiziert und zusammengesetzt werden. Die Produktkomplexität ist vergleichsweise gering, allerdings entsteht eine hohe kombinatorische Varianz durch die Vielfalt und Anordnung der Einbauteile. Die Produkte werden in einem standardisierten Prozess gefertigt und kundenindividuell montiert. Dem Unternehmen ist auch das Fallbeispiel für die lösungsorientierten Mustersequenzen entnommen, s. Kapitel 5.1.

Unternehmen C produziert hochwertige Messgeräte für die Infrarot-Spektralanalyse. Die Geräte sind modularisiert und werden in Abhängigkeit der vom Kunden gewünschten Messaufgabe mit den benötigten Zusatzmodulen ausgestattet. Die Individualisierung findet ausschließlich in der Endmontage statt.

Unternehmen D ist ein Hersteller von Abfüllanlagen für Nahrungsmittel, die aus mehreren, verketteten Systemkomponenten bestehen. Die Auftragsabwicklung erfordert einen hohen Anteil an Engineering und Projektierung. Die entstehenden, hochkomplexen Anlagen sind in Auslegung und in der Anordnungsgeometrie Unikate. Im Unterschied zum reinen Sondermaschinenbau sind die Systemkomponenten immer von gleicher Bauart und enthalten einen Mix aus Standardteilen und kundenindividuellen Teilen.

6.1.2 Informationsnutzung in den Unternehmen

Bei den vier Unternehmen wurde die erwartete Nutzung von produktionsnahen Informationsportalen überprüft. In strukturierten Einzelinterviews wurden für den Bereich der Endmontage die Nutzungsintensität der Informationscluster aus der Informationsanalyse und das realisierte Informationsangebot erhoben. Abbildung 2-14 zeigt die ermittelten Informationsprofile in einer vergleichenden Darstellung. Die Einstufung der Nutzungsintensität entspricht der Unterteilung des Intensitätsportfolios, vgl. Abbildung 2-7. Für das Informationsangebot wurden folgende Ausführungsformen unterschieden:

- nicht verfügbare Information;
- manuelle, papierbasierte Informationsträger;
- Informationsbereitstellung mit Standardsoftware, i. d. R. ERP-Systeme;
- standardisiertes Informationsangebot im Intranet;
- aufgabenspezifische Informationsbereitstellung.

Die Untersuchung der Informationsnutzung belegt die hohe Relevanz der kundenindividuell ausgeprägten Informationen bezüglich der Produktunterlagen, des Arbeitsplans, der Fertigungsaufträge, der Materialdisposition und für die Auftragsdokumentation bei den Unternehmen. Ebenfalls bestätigen sich die Aussagen bezüglich des praktizierten Informationsangebots, vgl. die Informationsanalyse in Kapitel 2.3.3. Bei drei der Unternehmen ist eine deutliche Diskrepanz zwischen der hohen Nutzungsintensität der Informationen und dem vorhandenen Informationsangebot zu konstatieren. Das vorgefundene Informationsangebot entspricht überwiegend nicht den Anforderungen an eine arbeitsplatz- und mitarbeiterspezifische Informationsbereitstellung. Die zur Auftragserfüllung unbedingt benötigten und permanent genutzten Informationen werden, trotz der hohen Nutzungsintensität, weiterhin aufwendig ausgedruckt und verteilt. Zum Teil werden auftragsindividuelle und variantendefinierende Informationen direkt aus den ERP-Systemen abgerufen. Dagegen finden sich im Intranet der Unternehmen vorwiegend selten genutzte, administ-

rative Informationen. Eine aufgabenspezifische Informationsbereitstellung war nur im Unternehmen D zum Zeitpunkt der Befragung prototypisch realisiert.

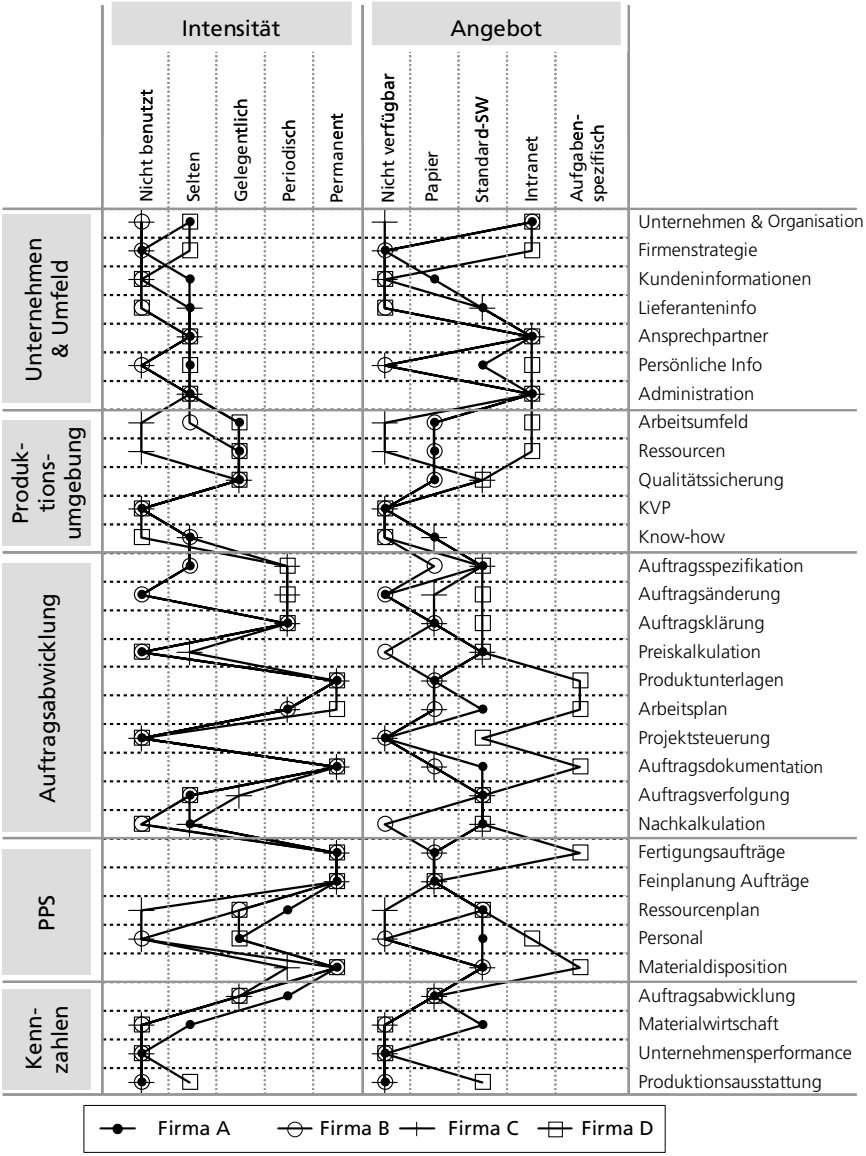


Abbildung 6-1: Exemplarische Relevanz und Informationsangebot bei KIP. Quelle: [Gerlach09]

6.1.3 Einsatzvoraussetzungen für Informationsportale in der Produktion

Ein produktionsnahes Informationsportal kann somit zur Effizienz der Informationsversorgung beitragen. In einem weiteren Schritt wurden in den Unternehmen die Einsatzvoraussetzungen für Informationsportale in der Produktion untersucht. Geprüft wurden die technischen Realisierungsmöglichkeiten und die Einsatzbedingungen vor Ort. Dabei zeigte sich, dass für Unternehmen A eine Informationsbereitstellung per Webbrowser nicht praktikabel ist. In der Endmontage erfolgt vorwiegend die Verdrahtung der elektrischen und elektronischen Bauteile der voluminösen Stromumrichter. Die Mitarbeiter benötigen hierfür detaillierte, großformatige Schaltpläne mit einem hohen Informationsgehalt, die nur ausschnittsweise auf Bildschirmen darstellbar sind. Zudem erlaubt die Dimension und Zugänglichkeit der Geräte keine Positionierung eines Monitors im Sichtfeld des Arbeitsraums. Bei den weiteren Unternehmen sind produktionsnahe Informationsportale prinzipiell einsetzbar. Folgende Situation wurde vorgefunden:

- Unternehmen B bietet alle Einsatzvoraussetzungen und zeigte großes Interesse an den Möglichkeiten zum Aufbau eines produktionsnahen Informationsportals.
- Unternehmen C erfüllt ebenfalls die technischen Voraussetzungen. Auch die Informationsrelevanz ist gegeben. Das Unternehmen produziert täglich nur ca. 6 Einheiten der hochpreisigen und ausgeprägt modularisierten Messgeräte. Die kundenindividuelle Ausgestaltung erfolgt dabei ausschließlich in der Endmontage, durch Kombination standardisierter Baugruppen. In der Konsequenz ist die absolute Nutzungsintensität der Kerninformationen vergleichsweise gering. Der Aufbau eines produktionsnahen Informationsportals ist daher nur sinnvoll, wenn weitere Bereiche außerhalb der Endmontage und zusätzliche Anwendungen mit erschlossen und genutzt werden.
- Unternehmen D war zum Zeitpunkt der Erhebung der Informationsnutzung bereits mit dem Aufbau einer ersten Pilotanwendung für ein Montageinformationsportal befasst.

Von den Unternehmen werden die Firmen B und D zur weiteren Evaluierung der neuen Methode zur Unterstützung des Aufbaus produktionsnaher Informationsportale herangezogen. In den folgenden Abschnitten werden für beide Unternehmen Informationsportale vorgestellt, deren Realisierung im ersten Fall unter Einsatz der Entwurfsmustersprache, und im zweiten Fall ohne Kenntnis der Entwurfsmustersprache erfolgte.

6.2 Fallbeispiel 1: Portalprototyp für kundenindividuelle Schaltschränke

Im ersten Fallbeispiel wurde ein Informationsportal für die Fertigung und Montage von elektrischen Schaltschränken unter Verwendung von Entwurfsmustern konzipiert und prototypisch implementiert. Die Konzeption des Portals und der Einsatz der Entwurfsmustersprache wurden im vorhergehenden Kapitel ausführlich vorgestellt, s. Abbildung 5-9. Wie an gleicher Stelle ausgeführt, erfüllt das Portal die Anforderungen an eine bedarfsgerechte, arbeitsplatz- und mitarbeiterspezifische Informationsversorgung. Die Umsetzung des konzipierten Informationsangebots für die Montagereihenfolge in dem Portalprototyp zeigt Abbildung 6-2. Die in dem Bildschirmabzug gezeigten Daten beruhen wiederum auf dem fiktiven Auftrag für kundenindividuelle Schaltschränke aus Kapitel 5.2.

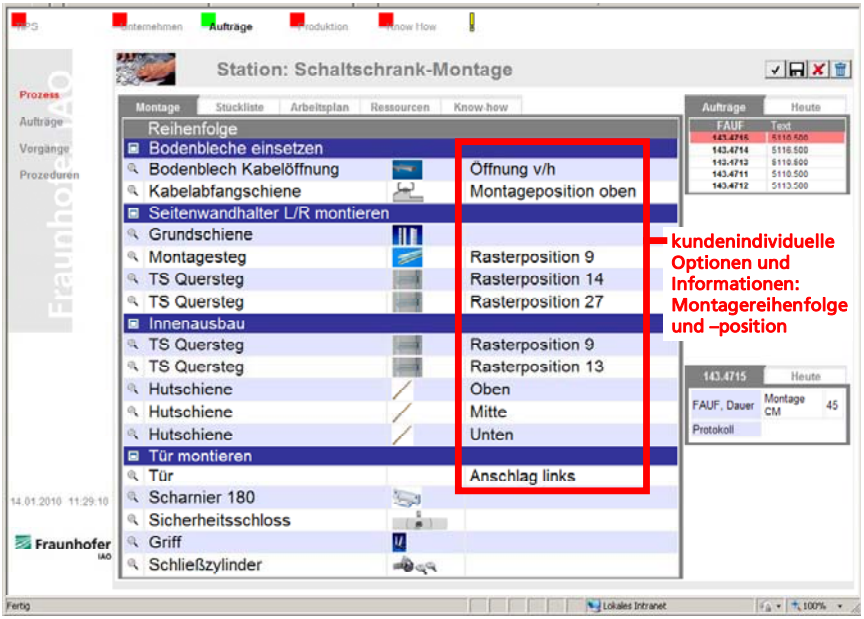


Abbildung 6-2: Bildschirmabzug des Informationsportals für die Endmontage von Schaltschränken

Implementierungsgrundlage des Portals ist eine Portal-Engine auf Basis der ASP-Technologie von Microsoft. Die Portal-Engine realisiert zahlreiche Muster insbesondere der Implementierungsphase. Weiterhin sind Funktionen enthalten für die Präsentation von Datenbankinhalten entsprechend den Entwurfsmustern für die Seitengestaltung. Abbildung 6-3 zeigt die Systemarchitektur der Portal-Engine und einen Ausschnitt der realisierten Implementierungsmuster.

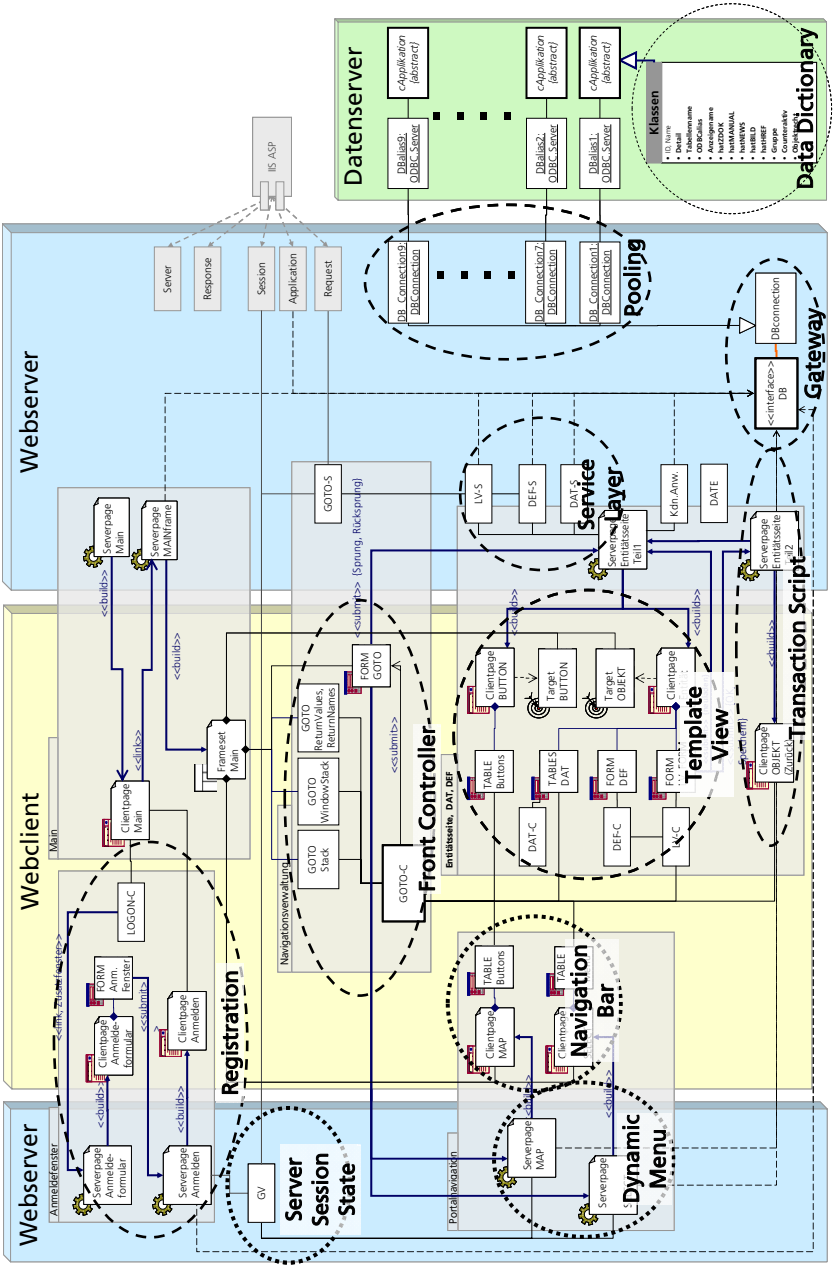


Abbildung 6-3: Technische Architektur des Portalprototyps mit Implementierungsmustern

Mit der Entwicklung der Portal-Engine war ein wechselndes Team von studentischen Praktikanten betraut. Die Implementierung des Portalprototyps für die Montagereihenfolge von Schaltschränken benötigte auf Basis der Vorarbeiten der Portal-Engine nur wenige Personentage. Folgende Erkenntnisse lieferten die Realisierung der Portal-Engine und des Portalprototyps:

- Die Praktikabilität der Entwurfsmustersprache konnte bestätigt werden:
  - die Entwurfsmuster sind anwendbar zur Realisierung produktionsnaher Informationsportale,
  - ihre Kooperationsbeziehungen zeigen sinnvolle Musterkompositionen,
  - die Gliederung der Sprache erlaubt einen zeitpunktgerechten Mustereinsatz,
  - Aufgabensequenzen forcieren ein strukturiertes, systematisches Vorgehen bei der Portalkonzeption.
- Der einmalige Einarbeitungsaufwand ist vergleichsweise gering:
  - maximal ein Personentag zum Kennenlernen des Aufbaus und der Anwendung der Entwurfsmustersprache,
  - weitere 2 bis 5 Personentage für das notwendige Studium der Musterbeschreibungen
- Ein Nutzen hinsichtlich der Entwurfsqualität ist von Beginn an gegeben:
  - die Entwurfsmuster inspirieren neue, im Entwicklungsteam noch unbekannte Lösungsvorschläge,
  - die Entwurfsmustersprache zeigt den Entwicklern alternative Lösungsmöglichkeiten auf und trägt so zur Absicherung der Entwurfsentscheidungen bei,
  - die Kooperationsbeziehungen der Entwurfsmuster erleichtern die Abschätzung von Entwurfsentscheidungen auf spätere Phasen der Portalentwicklung.
- Bei wiederholtem Einsatz der Entwurfsmustersprache stellt sich ein zusätzlicher Nutzen hinsichtlich einer effizienteren Portalentwicklung ein:
  - Entwurfsmuster liefern eine implizite Dokumentation der Portalrealisierung,
  - bewährte Lösungen lassen sich durch Mustersequenzen wiederverwenden und führen zu einer einheitlichen Portalanwendung und Portalimplementierung.

Insgesamt war der Eindruck im Entwicklungsteam, dass durch den Einsatz der Entwurfsmustersprache bereits mit der ersten Version des Portalprototyps ein hoher Reifegrad sowie eine gute Benutzbarkeit der Anwendung erreicht wurden. Zum Vergleich wurde ein zweites Portal betrachtet, dass vom Unternehmen D ohne Kenntnis der Entwurfsmuster realisiert wurde.

6.3 Fallbeispiel 2: Montageinformationsportal eines Anlagenherstellers

Als Pilotbereich für ein produktionsnahes Informationsportal wurde vom Unternehmen D die Endmontage von Behälterrüstteilen gewählt. Darunter sind Wechselgarnituren zu verstehen, mit denen die Anlagen im Betrieb für verschiedene Behälterformen gerüstet werden. In Abbildung 6-4 sind als Beispiele für Behälterrüstteile zu erkennen im Vordergrund eine Antriebsschnecke und in der Bildmitte eine Rotationsscheibe. Diese Teile sind in der Geometrie jeweils exakt an die Behälterformen angepasst.

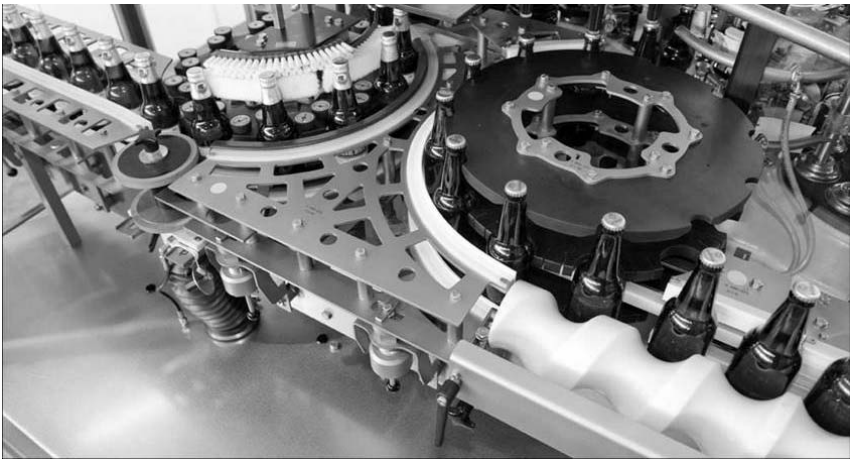


Abbildung 6-4: Wechselgarnituren einer Abfüllanlage. Quelle: [Kammermeier09, S. 251]

Für die Endmontage der Rüstgarnituren werden insbesondere Informationen aus den Auftragsstücklisten, Zeichnungen und der Auftragsspezifikation benötigt. Die Informationen wurden bisher in ausgedruckten, mehrseitigen Fertigungsaufträgen und CAD-Zeichnungen am Arbeitsplatz zur Verfügung gestellt. Der Aufwand für die Beschaffung und Sichtung der Informationen war hoch. In Kapitel 2 wurde aus demselben Unternehmen ein ähnlicher Arbeitsplatz für auftragsspezifische Vormontagen vorgestellt. Dort beträgt der Tätigkeitsanteil für die Informationsversorgung ca. 45 %. Aus diesem Grund hatte sich das Unternehmen entschlossen, ein Montageinformationsportal auf Basis des Werkzeugs Netweaver der Firma SAP zu realisieren. Jeder Arbeitsplatz in dem Pilotbereich ist mit einem berührungssensitiven Flachbildschirm ausgestattet, über den die Mitarbeiter ihren Auftragsvorrat einsehen können, alle erforderlichen Montageinformationen erhalten und Rückmeldungen absetzen können. Abbildung 6-5 zeigt einen Bildschirmabzug des Montageinformationsportals.

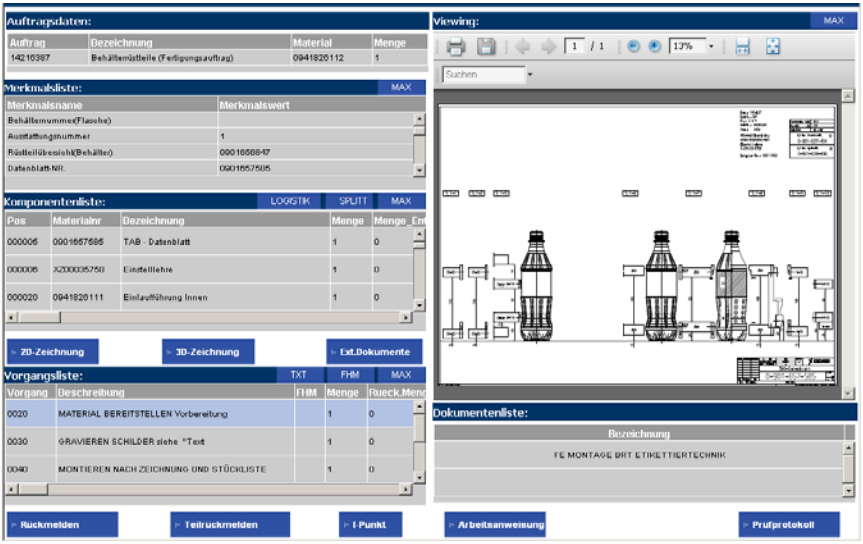


Abbildung 6-5: Bildschirmabzug des Montageinformationsportals. Quelle [Kammermeier09, S. 270]

Im linken Teil des Portals sind in mehreren Tabellen die benötigten Auftragsinformationen, Stücklisten und Arbeitspläne dargestellt. Rechts finden sich verschiedene Dokumente und Zeichnungen wieder. Auf den ersten Blick wirkt das Portal übersichtlich und klar strukturiert. Alle benötigten Informationen sind erkennbar angeordnet. Eine nähere Betrachtung offenbart jedoch deutliche Schwächen bei der Gestaltung des Informationsangebots, siehe Abbildung 6-6.

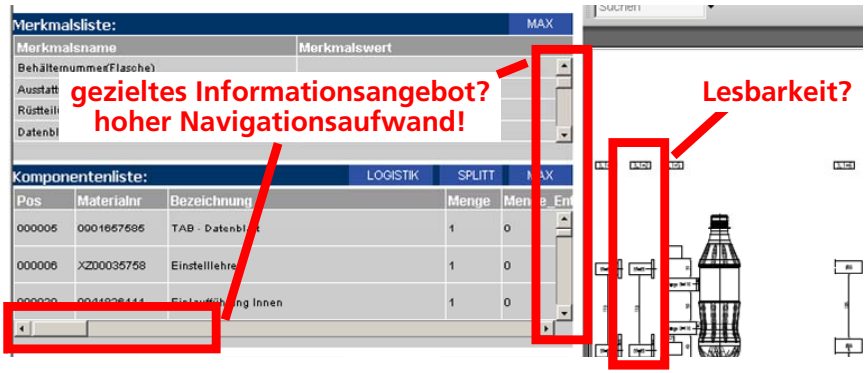


Abbildung 6-6: Navigationsprobleme in dem Montageinformationsportal

Die proportionalen Rollbalken an den Tabellenausschnitten für die Auftragsoptionen (Merkmalsliste) und die Auftragsstückliste (Komponentenliste) lassen einen hohen Interaktionsaufwand vermuten für die Navigation zu den gesuchten Informationen. Der horizontale Rollbalken an der Komponentenliste offenbart zudem, dass gar eine zweidimensionale Positionierung des dargestellten Tabellenausschnitts erforderlich ist. Dabei wird die An-

gabe der Position oder der Materialnummer nach links aus dem sichtbaren Bereich geschoben, sodass der Bezug zwischen Merkmalswert und Materialnummer verloren geht. Auch die Zeichnung in der rechten Bildschirmseite ist zu hinterfragen. Den dargestellten Höhenriss benötigen die Mitarbeiter häufig für die Angabe der Abstände zwischen dem oberen und unteren Halter. In dem dargestellten PDF-Dokument sind diese Angaben jedoch nur in einer starken Vergrößerung lesbar, was wiederum aufwendiges Positionieren des Zeichnungsausschnitts zur Folge hat. Erschwerend kommt hinzu, dass in einem Produktionsumfeld üblicherweise keine Maus als Eingabegerät zur Verfügung steht. Die Positionierung der Rollbalken erfolgt stattdessen umständlich mit speziellen Folientasten am Monitor.

So ist fraglich, ob diese Lösung tatsächlich geeignet ist, den Aufwand für die Informationsbeschaffung zu reduzieren. Ein Vergleich mit der Entwurfsmustersprache lässt auf Anhieb einige nicht adressierte Entwurfsaufgaben und Muster erkennen. Besonders auffällige Defizite sind:

- Eine Portalnavigation zum Beispiel in Form einer NAVIGATION BAR fehlt. Der Benutzer erhält keine Orientierungshilfen über das sonstige Informationsangebot und den Navigationspfad zur aktuellen Seite. Es gibt keine Möglichkeit zurück zu navigieren oder die Seite zu beenden mit einem HOME LINK oder mit GO BACK ONE STEP.
- Die Komponentenliste ist nicht vorselektiert. Eine Gruppierung mit einem ACCORDION oder COLLAPSIBLE PANELS könnte das Lokalisieren der relevanten Stücklistenpositionen erleichtern. Alternativ könnte ein TABLE FILTER nicht relevante Einträge ausfiltern, z. B. Standardbaugruppen und Befestigungsmaterial.
- Die Analysemuster THINGS OF THE ENTERPRISE und THINGS, THING TYPES AND CATEGORIES zeigen Lösungen für hierarchisch strukturierte beziehungsweise klassifizierte Ressourcenlisten. Damit ließe sich die Komponentenliste eventuell weiter untergliedern.
- Falls keine Selektion der Tabellen möglich ist, sollte wenigstens ein TABLE SORTER vorhanden sein. Damit ist durch Anklicken der Spaltenüberschriften die Tabelle einfach zu sortieren nach wechselnden Kriterien, zum Beispiel nach der Positionsnummer, der Artikelnummern oder der Artikelbezeichnung.
- Das horizontale Positionieren der Komponentenliste lässt sich durch eine geeignete Ausgabeorganisation vermeiden. Dafür ist die Komponentenliste auf die identifizierenden Merkmale in den Spalten Position, Materialnummer und Bezeichnung zu reduzieren. Alle weiteren Angaben der restlichen Spalten ließen sich beispielsweise mit OVERVIEW BESIDE DETAIL für ein markiertes Teil in einem gesonderten Bereich vollständig darstellen. Ersatzweise könnte ein FISHEYE die Werte als zusätzliche Zeilen in die Tabelle einblenden.

Die aufgeführten Defizite bei der Gestaltung des Portals belegen, dass die Entwurfsmustersprache zahlreiche Hinweise und Lösungsvorschläge für die Strukturierung und Aufbereitung der präsentierten Informationen bietet. In dem gezeigten Portal wurden jedoch die Ausgangsinformationen aus den Datenbanktabellen des ERP-Programms nahezu unverändert übernommen. Offensichtlich hätte die Verwendung und Beachtung der Entwurfsmustersprache erheblich zur Benutzbarkeit und Effizienz des Informationsangebots beitragen können.

6.4 Diskussion und Ausblick

Diskussion

Die Evaluierung zeigt, dass Informationsportale eine bedarfsgerechte und effiziente produktionsnahe Informationsversorgung ermöglichen und in vielen Bereichen einsetzbar sind. Vereinzelt vorliegende, unternehmensspezifische Ausschlussgründe sind üblicherweise evident. Gegebenenfalls sind diese vor einem konkreten Einsatz abzuprüfen.

Aus den beiden Fallbeispielen wird ersichtlich, dass die neue Entwurfsmustersprache Portalentwickler auf alle Entwurfsaufgaben hinführt und bewährte Lösungen hierfür vorhält. Offensichtliche Mängel bei der Gestaltung der Portale, wie in Fallbeispiel 2 aufgezeigt, werden systematisch vermieden. Zudem sichert die Bündelung und Gegenüberstellung von alternativen Lösungskonzepten für Entwurfsaufgaben die Entwurfsentscheidungen ab und erlaubt den Portalentwicklern, die jeweils beste Lösung auszuwählen. Bei der notwendigen arbeitsplatz- und mitarbeiterspezifischen Ausrichtung des Informationsangebots bietet die neue Entwurfsmustersprache die erforderliche konzeptionelle Unterstützung, s. Tabelle 6-2.

Anforderung		Erfüllung
Komplett: Konzepte für den gesamten produktionsnahen Informationsbedarf	+	Die Entwurfsmustersprache ist nicht auf einzelne Informationsinhalte beschränkt. Die enthaltenen Muster lassen sich bei der Realisierung aller produktionsnahen Informationsangebote anwenden. Allerdings sind für Produktionsinformationen nur relativ abstrakte Analysemuster beschrieben.
Problemorientiert konzeptionell: Konzepte für ein individualisiertes, selektives Informationsangebot nach dem Prinzip 5S und in den Dimensionen Arbeitsplatz und Mitarbeiter	+	Zur Beschreibung von problemorientierten Konzepten für Informationsangebote sieht die Methode Möglichkeiten zur Definition von Mustersequenzen vor. Mustersequenzen dokumentieren und standardisieren Informationsangebote in einem Portal, die den Anforderungen an eine aufgabenzentrierte Informationsbereitstellung genügen. Mustersequenzen können jedoch nur für beispielhafte Einsatzfälle vordefiniert werden.
Lösungsorientiert konzeptionell: Konzepte für die Realisierung eines produktionsnahen Portals und dessen Integration in die vorhandene Informationsarchitektur	++	Lösungsorientierte Konzepte bieten unmittelbar die Entwurfsmuster an. Die Entwurfsmustersammlung deckt dabei alle Funktionsblöcke der Lösungsdomäne ab.
Konstruktiv: Unterstützung aller Entwicklungssichten: Analyse, Entwurf, Implementierung	++	Die Entwurfsmustersprache bietet Entwurfsmuster für alle konstruktiven Sichten und zeigt deren Zusammenwirken durch die Vernetzung und die Aufgabensequenzen.

Tabelle 6-2: Bewertung der Methode zur Realisierung produktionsnaher Informationsportale

Abschließend ist festzustellen, dass die entwickelte Methode alle in Kapitel 2 aufgestellten Anforderungen gut bist sehr gut erfüllt. Der Nutzen der Methode ist zu sehen in einem reduzierten Aufwand bei der Portalrealisierung, vor allem aber in einer Steigerung der Benutzungseffizienz und einer Professionalisierung der Informationsangebote. Die Wiederverwendung erprobter Lösungen führt zu Anwendungen, die bereits bei der Einführung einen hohen Reifegrad aufweisen, einheitlich gestaltet und für die Benutzer erwartungskonform sind. Mit dem Einsatz der Methode ist eine deutliche Verbesserung der effizienten Informationsversorgung bei KIP zu erreichen.

Ausblick
Die Entwurfsmustersprache offeriert vielfältige Möglichkeiten der Gestaltung von produktionsnahen Informationsangeboten bei KIP. Als Voraussetzung muss die Datenbasis der Produktions- und Auftragsdaten die Individualisierungsmöglichkeiten aufzeigen und von den Standardumfängen unterscheiden. Die Evaluierung in den Unternehmen zeigte, dass die Daten häufig noch nicht in der erforderlichen Granularität, Qualität und Struktur vorliegen. Eine wesentliche Voraussetzung ist eine Unterscheidung und Kennzeichnung von individuellen vs. generellen Produktkomponenten. Zukünftig ist ein vermehrter Einsatz von Softwarewerkzeugen zur Auftragskonfiguration anzustreben. Eine bessere Integration und die Berücksichtigung derartiger Strukturmerkmale für Daten in den Konfigurationsregeln könnte eine weitere Verbesserung der produktionsnahen Informationsbereitstellung ermöglichen.

Aktuell erfolgt die Individualisierung von Produkten überwiegend in der Produktion. Ein Trend besteht hin zur weiteren Individualisierung unter Einbezug von konstruktiv zu behandelnden Merkmalen im Rahmen des sogenannten „Customer Co-Designs". Ebenfalls steigen die Anteile von Individualisierungslösungen über produktbegleitende Dienstleistungen. Zukünftig ist mit einer stärkeren Einbindung von Entwicklungs- und Serviceaufgaben in die Abwicklung von kundenindividuellen Aufträgen zu rechnen. Die Koordination, die Kommunikation und der Austausch von Information und Wissen zwischen Produktion, Entwicklung und Service wird sich weiter intensivieren. Portale könnten unterstützend wirken, durch den Einsatz von Elementen des sogenannten „Web 2.0". Erste Muster hierfür zeichnen sich ab. Zum Beispiel skizziert [Buschmann07, S. 238] Protopatterns einer diesbezüglichen Mustersprache. Für den Einsatz in der Produktion lassen sich die Muster und Formen des Webs 2.0 sicher nicht direkt übernehmen. Eine enge Verzahnung mit den operativen Auftragsinformationen ist anzustreben, wie dies in der vorliegenden Arbeit bereits für das Beispiel der asynchronen Benachrichtigung gezeigt wurde. Ein Ausbau der Entwurfsmustersprache mit Mustern für das Web 2.0 könnte beitragen, diese Techniken vermehrt auch in der Produktion zu nutzen, mit Informationsportalen als Medium für den umfassenden, mitarbeiterspezifischen Informationsaustausch und die Kommunikation.

Die Entwicklung der Entwurfsmustersprache erfolgte im Hinblick auf eine Informationsversorgung bei KIP mittels produktionsnahen Informationsportalen. Eine hinreichende Voraussetzung für den Aufbau von arbeitsplatz- und mitarbeiterspezifischen Informationsangeboten ist die hohe Relevanz der Informationsbedarfe in Verbindung mit der Nutzungsintensität. Die Voraussetzungen sind nicht notwendig auf eine KIP beschränkt. Auch bei standardisierten Produkten und Produktionsformen können die Voraussetzungen erfüllt sein. So kann zum Beispiel die Produktion sicherheitsrelevanter und dokumentationspflichtiger Teile erfordern, dass stets aktuelle Fertigungsunterlagen zur Verfügung stehen und diese auch tatsächlich beachtet wurden. Hier bieten sich Portale an zur Informationsbereitstellung mit einer sporadischen Notifikation und bidirektionaler Kommunikation zum Quittieren der geänderten Informationsstände. Auch für derartige Anwendungsgebiete ist eine Unterstützung durch die Entwurfsmustersprache beim Aufbau von Informationsportalen möglich.

7 Zusammenfassung

Die fortschreitende Informatisierung der Wertschöpfungskette überwindet die Trennung von massenhafter und individueller Fertigung im Konzept der kostenneutralen, kundenindividuellen Produktion (KIP). KIP ist eine hybride Produktionsstrategie, die sich zwischen den Alternativen der Kostenführerschaft und Differenzierung ansiedelt und eine möglichst individuelle Ausführung einzelner Produktexemplare bei lediglich moderaten Preisaufschlägen realisiert. Dabei erhalten Informationen über die Ausführungsoptionen der kundenindividuellen Produkte den Rang elementarer Produktionsfaktoren, die nicht substituierbar sind und einer permanenten Nutzung unterliegen. Variantenbildende Arbeitsgänge „verarbeiten" Information wie Material, die Präsenz beider Arten von Ressourcen ist für jedes zu fertigende Produktexemplar zwingend erforderlich. Die Bereitstellung von Information ist daher sorgfältig zu gestalten und am Informationsbedarf zu orientieren, der an jedem Arbeitsplatz tatsächlich vorliegt. Generalisierte Informationsüberangebote aus den Originalquellen der ERP-Applikationen sind nicht geeignet. Sie sind universell einsetzbar und decken einen unspezifizierten Informationsbedarf. In einem kundenindividuellen Umfeld mit seiner hohen Nutzungsintensität führen sie zu erheblichem Mehraufwand für die Beschaffung und Sichtung der Informationen.

Die vorliegende Arbeit verfolgt das Ziel einer effizienten Informationsversorgung bei KIP. Ein entsprechendes, schlankes Informationsangebot ist im Kern auf absolut notwendige, essenzielle Informationen über die kundenindividuellen Auftragsspezifikationen zu beschränken, übersichtlich und auf einen Blick erfassbar zu präsentieren sowie frei zu halten von unnötigen, Zeit verschwendenden Interaktionen. Offensichtlich ist es auf jeden Arbeitsplatz auszurichten. Darüber hinaus hat es variierende Informationsstände der Mitarbeiter zu berücksichtigen, die infolge der notwendigen Flexibilität einer KIP häufig ihre Arbeitsplätze wechseln. Eine effiziente und bedarfsgerechte Informationsversorgung bei KIP ist somit arbeitsplatzspezifisch und mitarbeiterspezifisch auszuführen. Ein probates Medium zur bedarfsgerechten und personalisierten Informationsversorgung sind Portale. Allerdings werden sie selten als Auftragsinformationssysteme für die Produktion genutzt. Es mangelt somit an Erfahrungen über die systematische Gestaltung und den Aufbau produktionsnaher Informationsportale. Eine unterstützende Methode muss sowohl problemorientierte Konzepte als auch deren Implementierung in einem Portal berücksichtigen.

Zur Lösung des Defizits wurde eine Methode auf Basis einer neuen Entwurfsmustersprache für produktionsnahe Informationsportale entwickelt. Sie zeigt umfassend die anstehenden Entwicklungsaufgaben für Informationsportale, bietet über 200 Entwurfsmuster hierfür, vermittelt Wechselwirkungen und Kooperationsbeziehungen zwischen Mustern und fasst sie in Aufgabensequenzen zusammen, die eine zielgerichtete und systematische Musterauswahl ermöglichen. Den Schwerpunkt der Entwurfsmuster bilden Lösungskonzepte für durchgängige und anforderungsgerechte Informationsangebote, die intuitiv zu bedienen, flexibel und den Bedürfnissen der Mitarbeiter anzupassen sind. Mittels der Entwurfsmustersprache konzipierte Portalseiten realisieren eine Folge von angewandten Mustern. Die so gefundenen Mustersequenzen lassen sich für weitere Informationsangebote und Portalseiten nutzen. Sie tragen bei zur Vereinheitlichung der Implementierung und des technischen Aufbaus eines Portals. Vor allem sorgen sie für konsistente und damit für den Benutzer erwartungskonforme und benutzbare Informationsangebote.

Bei konsequenter Umsetzung der Entwurfsmustersprache ist die Realisierung von produktionsnahen Informationsportalen mit schlanken, arbeitsplatz- und mitarbeiterspezifischen Informationsangeboten möglich, die die Ansprüche an eine effiziente Informationsversorgung bei KIP erfüllen. Das erforderliche Nutzenpotenzial für ein derart aufwendig und sorgfältig gestaltetes Informationsangebot ist in einer kundenindividuellen Produktion, mit ihrer hohen Nutzungsintensität der Informationen, in vielen Fällen gegeben.

8 Summary

The ongoing pervasion of the value creation chain by information technology permits the combination of the principles of mass production and individualised production in the concept of customised production (CP). CP is a hybrid strategy of production, which is settled between the alternatives of cost leadership and differentiation. It allows products, which are very close to the individual needs and demands of the customer at only moderate additional costs. All samples of customised products are different in some defined features, shape or functionalities and build upon varying order specifications. Hereby, information concerning the varying parts of the order specifications is essential to the workers at the shop floor. Since this information is needed for every sample of the product, it is also of high usage. Production activities regarding the customised parts of the products are relying on this information the same way they use materials. Both kinds of resources are essential, without their availability orders cannot be fulfilled and the customised product samples cannot be built. The provision of information for CP has to be designed carefully and strongly oriented on the exact demand for information of every workstation. Provision of generalised, overburdened information as it is provided by standardised information systems, for example ERP applications, is not sufficient. It is designed to be of universal usage according to an unspecific demand for information. In a customised production provision and inspection of generalised information results in considerable additional efforts by the workers.

The objective of this thesis is the efficient provision of information for workers in CP. A corresponding offer of nformation has to be reduced to the core of absolute necessary, essential information demands regarding the customised order specifications. It has to be clearly arranged, recognisable at a glance, free of all unnecessary pieces of information and has to avoid time consuming activities for navigation and interaction. Obviously it has to be individually designed for every workstation. Furthermore it has to account for varying levels of information for the workers. Therefore an efficient provision of information at CP has to be tailored for every workstation and personalised to the workers. Webportals are a suitable instrument for user oriented, personalised and tailored provision of information. However for provision of order information at the shop floor they are rarely used. Therefore it lacks of experience and expertise regarding the systematic design and realisation of webportals for the shop floor level. A supporting method is needed, which has to consider problem oriented concepts as well as their implementation in a portal based on web technology.

To solve the deficiency, a method was developed which utilises a new pattern language for shop floor related information portals. The language comprises a comprehensive overview of the typical design steps and tasks for information portals. It offers more than 200 design patterns to solve the design tasks, clarifies dependencies and co-operation between the patterns and ties them together in sequences. Therefore, it allows a systematic, target oriented selection. The emphasis is placed on concepts for user centered provision of information which is intuitive and easy to use and also flexible and adaptable to the individual demands of the workers. Pages of a portal which are designed and derived from the new pattern language are representing sequences of patterns. These sequences of patterns can be reused to design similar information offers and portal pages. They contribute to the harmonisation of portals, especially in terms of consistent and therefore usable user interfaces.

Thorough utilisation of the pattern language is the enabler for shop floor related information portals with a lean, user oriented, personalised and tailored provision of information. The exalted requirements for an efficient provision of information in a customised production can thus be fulfilled. The significant benefit to justify this thorough and elaborate designed provision of information is shown by numerous case studies of customised production with its high usage rate of information.

.

9 Anhänge

9.1 Literaturverzeichnis

9.1.1 Quellen der Erhebung des Informationsraums

[Augustin98] Augustin, R.: Integrierte Planung der Anwendung von Informations- und Kommunikationstechnologie (IKT): eine Methode zur Priorisierung von IKT-Verbesserungen für produzierende Unternehmen. Aachen: Shaker Verlag. Zugl.: Dissertation Technische Universität Darmstadt, 1998.

[Bach00] Bach, V.: Business Knowledge Management in der Praxis: Prozessorientierte Lösungen zwischen Knowledge Portal und Kompetenzmanagement. Berlin, Heidelberg [u.a.]: Springer Verlag, 2000.

[Bauer01] Bauer, H.: Unternehmensportale: Geschäftsmodelle, Design, Technologien. Bonn: Galileo Press, 2001.

[Bendler02] Bendler, A.: Wissensmanagement im internationalen Vertrieb produzierender Unternehmen: ein Modell zur Auswahl von Wissensmanagement-Maßnahmen in Abhängigkeit unterschiedlicher Internationalisierungsstrategien. Aachen: Shaker Verlag, 2002. Zugl.: Dissertation Technische Universität Darmstadt, 2001

[Block] Block, M.; Greif, H.: Smartville – Innovatives Automobilunternehmen der Zukunft. In: Digitale Fachbibliothek Innovationsmanagement/ Hrsg. Barske, H. [u.a.], Kap. 05.01. (Loseblattausgabe).

[Buck96] Buck, H.: Qualifikations- und lernförderliche Gruppenarbeit in der Monatege. In: Betriebliche Folgen veränderter Altersstrukturen in der Montage (1996), S. 128-156.

[Ebel08] Ebel, B.: Kompakt-Training Produktionswirtschaft. 2., völlig überarbeitete Auflage, Ludwigshafen am Rhein: Kiehl, 2008.

[Gaida00] Gaida, R.: Informationssysteme für die Gruppenarbeit in der Produktion zur Unterstützung eines umfassenden Qualitätsmanagements. Aachen: Shaker Verlag, Zugl.: Dissertation RWTH Aachen, 2000.

[Gerlach03] Gerlach, S.; Spath, D.: Informationen für Teams in Montage und Produktion. In: wt Werstattstechnik online Band 93 (2003) 9, S.624-626.

[Gurzki03] Gurzki, T.; Özcan, N.: Unternehmensportale: Kunden-, Lieferanten- und Mitarbeiterportale in der betrieblichen Praxis. Stuttgart: Fraunhofer IRB Verlag, 2003.

[Kindermann01] Kindermann, D.: Integrierte Softwaresysteme für ERP. In: PPS Management Band 6 (2001) 3, S.43-45.

[Kistner90] Kistner, K.-P.: Produktionsplanung. Heidelberg: Physica-Verlag, 1990.

[Klein95] Klein, T.; Endres, C.: Mitarbeitermotivation – unterstützende Ressource. In: Personal (1995) 6, S. 280-285.

[Kletti07] Kletti, J.: Konzeption und Einführung von MES-Systemen: zielorientierte Einführungsstrategie mit Wirtschaftlichkeitsbetrachtungen, Fallbeispielen und Checklisten; mit… 53 Tabellen. Berlin, Heidelberg [u.a.]: Springer Verlag 2007.

[Maier-Rothe] Maier-Rothe, C.: Benchmarking in der Produktion. In: Digitale Fachbibliothek Innovationsmanagement/ Hrsg. Barske, H. [u.a.], Kap. 06.02. (Loseblattausgabe).

[Mertens09] Mertens, P.: Integrierte Informationsverarbeitung 1. 17., überarbeitete Auflage, Wiesbaden: Gabler, 2009.

[Niefer93] Niefer, H.: Planung, Einführung und Optimierung von Gruppenarbeit in der Teilefertigung. München, Wien: Carl Hanser Verlag. Zugl.: Dissertation Technische Universität Berlin, 1993.

[Piller06] Piller, F. T.: Mass Customization: ein wettbewerbsstrategisches Konzept im Informationszeitalter. 4. aktualisierte Auflage, Wiesbaden: Gabler, 2006.

[Schuon00] Schuon, J.; Wilhelm, S.; Soltau, L.: Informations-Management in der Produktion. In:Computerwoche 27 (2000) 11, S.74-76.

[Springer] Springer, R.: Effektivität unterschiedlicher Formen von Gruppenarbeit. In: Digitale Fachbibliothek Das flexible Unternehmen/ Hrsg. Antoni, C. [u.a.], Kap. 01.01. (Loseblattausgabe).

[Trovarit05] Trovarit AG: Anwender-Zufriedenheit ERP/Business Software Deutschland 2005, Aachen: 2005.

[Ullrich] Ullrich, M.; Lindinger, C.: Die vitale Fabrik – systematische Unternehmensgestaltung bei Tally.

[Wollschlaeger02] Wollschlaeger, M.; Diedrich, C.: Webportal zur Instandhaltung von Maschinen und Anlagen. In: Industrie Management 18 (2002) 2, S.37-40.

9.1.2 Quellen der Muster des Entwurfsmusterkatalogs

[Anastopoulos01] Anastopoulos, M.; Romberg, T.: Referenzarchitekturen für Web-Applikationen. http://app2web.fzi.de/themen/ap3/-referenzarchitekturen.pdf, 2001. Datum des Aufrufs: 05.12.2006.

[Buschmann98] Buschmann, F. et al.: Pattern-orientierte Software-Architektur. Ein Pattern-System. Bonn: Addison-Wesley, 1998.

[Cunningham94] Cunningham, W.: The CHECKS Pattern Language of Information Integrity. http://c2.com/ppr/checks.html, 1994. Datum des Aufrufs: 27.01.2007.

[Celko04] Celko, J.: Joe Celko's Trees and hierarchies in SQL for smarties. Amsterdan, Heidelberg [u.a.]: Elsevier, Morgan Kaufmann, 2004

[vanDuyne02] van Duyne, D.K.; Landay, J.A.; Hong, J.I.: The Design of Sites: Patterns, Principles, and Processes for Crafting a Customer-Centered Web Experience. San Francisco: Addison-Wesley Professional, 2002.

[Eriksson00] Eriksson, H.-E.; Penker, M.: Business Modeling with UML Business Patterns at Work. Chichester: John Wiley & Sons Ltd., 2000.

[Fowler97] Fowler, M.: Analysis Patterns: Reusable Object Models. Menlo-Park: Addison-Wesley, 1997.

[Fowler02] Fowler, M.: Patterns of Enterprise Application Architecture. Addison-Wesley Professional, 2002.

[Fowler06] Fowler, M.: Patterns for things that change with time. http://martinfowler .com/ap2/timeNarrative.html, 2006. Datum des Aufrufs: 03.03.2006.

[Hay96] Hay, D. C.: Data Model Patterns: Conventions of Thought. Dorset House Publishing Co Inc., 1995.

[Gamma96] Gamma, E.; Helm, R.; Johnson, R.; Vlissides, J.: Entwurfsmuster - Elemente wiederverwendbarer objektorientierter Software. München: Addison-Wesley, 1996.

[Griffiths98] Griffiths, R. et al.: The Brighton Usability Pattern Collection. http://www.cmis.brighton.ac.uk/Research/patterns/Banner.html, 1998. Datum des Aufrufs: 09.04.2007.

[Keller98] Keller, W.: Object/Relational Access Layers A Roadmap, Missing Links and More Patterns. http://www.objectarchitects.de/ObjectArchitects/ papers /Published/ZippedPapers/or06_proceedings.pdf,1998. Datum des Aufrufs: 01.11.2006.

[Laakso03] Laakso, S. A.: User Interface Design Patterns. http://www.cs.helsinki.fi/ u/salaakso/patterns/, 2003. Datum des Aufrufs: 28.06.2005.

[Mahemoff06] Mahemoff, M.: Ajax Patterns. http://ajaxpatterns.org/Data_Dictionary, 2006. Datum des Aufrufs: 03.03.2008.

[Riehle97] Riehle, D. et al.: Serializer. In: Martin, Buschmann, Riehle (Hrsg.): Pattern Languages of Program Design 3, Reading, MA: Addison-Wesley, 1997.

[Rüping07] Rüping, A.: Software Architectures for Web Content Management - The Big Picture. http://hillside.net/europlop/europlop2007/workshops/F6.pdf, 2007. Datum des Aufrufs: 21.07.2009.

[Schümmer07] Schümmer, T.; Lukosch, S.: Patterns for computer-mediated interaction. Hoboken, NJ: Wiley, 2007.

[Tidwell99] Tidwell, J.: COMMON GROUND: A Pattern Language for Human-Computer Interface Design.http://www.mit.edu/~jtidwell/ interaction_patterns.html, 1999. Datum des Aufrufs: 26.01.2007.

[Tidwell06] Tidwell, J.: Designing Interfaces - Patterns for Effective Interaction Design. Beijing, Köln [u.a.]: O'Reilly, 2006.

[Welicki07] Welicki, L.: Patterns for Factoring Responsibilities when Working with Objects and Relational Databases. http://hillside.net/europlop/ workshops/ D5.pdf, 2007. Datum des Aufrufs: 06.09.2007.

[vanWelie07] van Welie, M.: A Pattern Library for Interaction Design. http://www.welie.com/, 2007. Datum des Aufrufs: 21.08.2007.

[Yoder98] Yoder, J.; Barcalow, J.: Architectural Patterns for Enabling Application Security In: Harrison, N. et al. (Ed.): Pattern Languages of Program Design 4. Reading: Addison Wesley, S. 301-336, 1998.

9.1.3 Textquellen

[Alexander77] Alexander, C.; Ishikawa, S.; Silverstein, M.; et al.: A Pattern Language: Towns, Buildings, Construction. New York: Oxford University Press, 1977.

[Amberg05] Amberg, M. et al.: Bayerischer Forschungsverbund Wirtschaftsinformatik - Abschlussbericht. http://www.bayfor.org/mount_media /downloads/ verbuende/forwin/fwn_abschlussbericht.pdf, 2005. Datum des Aufrufs: 09.04.2009.

[Ambler98] Ambler, S. W.: Process Patterns - Building Large-Scale Systems Using Object Technology. Cambridge: Cambridge University Press, 1998.

[Ambler99] Ambler, S. W.: More Process Patterns - Delivering Large-Scale Systems Using Object Technology. Cambridge: Cambridge University Press, 1999.

[Aneja00] Aneja, A.; Rowan, C.; Brooksby, B.: Corporate Portal Framework for Transforming Content Chaos on Intranets. http://developer.intel.com/-technology/itj/ q12000/pdf/portal.pdf, 2000. Datum des Aufrufs: 16.01.2007.

[Antoni09] Antoni, C. H.: Gruppen- und Teamarbeit. In: Bullinger, H.-J., Spath, D.; Warnecke, H. J.; Westkämper, E. (Hrsg.): Handbuch Unternehmensorganisation. Berlin [u.a.]: Springer, S. 213-222., 2009.

[Appleton98] Appleton, B.: Patterns and Software: Essential Concepts and Terminology. http://www.cmcrossroads.com/bradapp/docs/patterns-intro.html, 1998. Datum des Aufrufs: 19.01.2006.

[Austaller04] Austaller, G.; Hartl, A.; Lauff, M.; Lyardet, F.; Mühlhäuser, M.: Technologiebewusstes Design von Web-Anwendungen. In: Kappel, G. et al. (Hrsg.): Web Engineering. dpunkt-Verlag. S. 101-132, 2004.

[Bach05] Bach, V.: Rollenzentriertes Portalmanagement - Geschäfts- und Systemarchitekturen für individualisierte Mitarbeiter- und Konsumentenbeziehungen. Wiesbaden: Deutscher Universitäts-Verlag. Zugl.: Habilitation Universität St. Gallen, 2005.

[Balzert00] Balzert, H.: Lehrbuch der Software-Technik. Heidelberg: Spektrum Akademischer Verlag, 2000.

[Balzert01] Balzert, H.: Lehrbuch der Software-Technik: Software-Management, Software-Qualitätssicherung, Unternehmensmodellierung. 2. Auflage, Heidelberg: Spektrum Akademischer Verlag, 2001.

[Beck00] Beck, K.: Extreme Programming Explained. Boston, MA: Addison-Wesley Professional, 2000.

[Beck87] Beck, K.; Cunningham, W.: Using Pattern Languages for Object-Oriented Programs. OOPSLA 87 workshop on Specification and Design for Object-Oriented Programming, October 1987. http://c2.com/doc/oopsla87.html, 1987. Datum des Aufrufs: 09.03.2009.

[Bender99] Bender, K.: Analysemuster in der Architektur kommerzieller Informationssysteme. Lohmar: Josef Eul Verlag. Zugl.: Dissertation Technische Universität Dresden, 1999.

[Blecker07] Blecker, T.; Friedrich, G.: Mass Customization Manufacturing Systems. IEEE Transactions on EngineerignManagement, Vol. 54, No. 1, 2007.

[Boehm88] Boehm, B. W.: A Spiral Model of Software Development and Enhancement. http://www.sce.carleton.ca/faculty/ajila/4106-5006/Spiral % 20Model% 20Boehm.pdf, 1988. Datum des Aufrufs: 20.03.2007.

[Bomsdorf04] Bomsdorf, B.; Szwillus, G.: Rollen- und aufgabenbasierte Webmodellierung. R. Keil-Slawik, H. Selke, G. Szwillus (Hrsg.): Mensch & Computer 2004: Allgegenwärtige Interaktion. München: Oldenbourg Verlag, S. 115–126, 2004.

[Brandon08] Brandon, D. M. (Ed.): Software Engineering for Modern Web Applications: Methodologies and Technologies. Hershey, New York: Information Science Reference, 2008.

[Bruse05] Bruse, T.: Groupware-basiertes Workplace-Portal G8 : Konzeption und prototypische Implementierung eines Workplace-Portals zur Unterstützung des organisatorischen Wissensmanagements. Hochschulschrift. Zugl. Dissertation Universität Paderborn, 2005.

[Bullinger00] Bullinger, H.-J.; Gerlach, S.; Rally, P.: Dezentrale Verantwortungsbereiche in Produktionsnetzwerken. In: Kaluza, B., Blecker, Th. (Hrsg): Produktions- und Logistikmanagement in virtuellen Unternehmen und Unternehmensnetzwerken. Berlin [u.a.]: Springer-Verlag, 2000.

[Bullinger05a] Bullinger, H.-J. ; Schweizer, W.: Intelligent production-competition strategies for producing enterprises. In: International Journal of Production Research, Vol. 44: Nr. 18-19, S. 3575 — 3584. 2006

[Bullinger05b] Bullinger, H.-J. ; Schweizer, W.: Mass customization in process industries -from mass production to mass customization. In: Tseng, M.: Interdisciplinary world congress on mass customization and personalization: MCPC 2005 ; Hong Kong, September 18-21, 2005.

[Buschmann07] Buschmann, F.; Henney, K.; Schmidt, D. C.: Pattern-Oriented Software Architecture: On Patterns and Pattern Languages. Chichester: John Wiley & Sons Ltd., 2007.

[Ceri03] Ceri, S. et al.: Designing data-intensive Web applications. Amsterdam [u.a.]: Morgan Kaufmann, 2003.

[Chan02] Chan, M.; Chung, W.: A framework to develop an enterprise information portal for contract manufacturing. International Journal of Production Economics Volume 75, Issues 1-2, 10 January 2002, S. 113-126, 2002.

[Clement06] Clement, U.: Arbeit unterhalb der Facharbeiterqualifikation und ihre Herausforderungen für die europäische Berufsbildungspolitik. In: Clement, U.; Lacher, M.: Produktionssysteme und Kompetenzerwerb. Stuttgart: Franz Steiner Verlag, S. 93-112., 2006.

[Collins01] Collins, H.: Corporate Portals. New York: American Management Association, 2001.

[Conallen99] Conallen, J.: Building Web Applications with UML. Addison Wesley, 1999.

[Coplien95] Coplien, J. O.: A generative development-process pattern language. In: Coplien, Schmidt, Eds.: Pattern Languages of Program Design. New York: Addison-Wesley, S. 183-237, 1995.

[Coplien96] Coplien, J. O.: Software Patterns. New York: SIGS Books & Multimedia, 1996.

[Coplien97] Coplien, J. O.: Idioms and Patterns as Architectural Literature. Software, IEEE, Volume 14, Issue 1, Jan/Feb 1997, S. 36 - 42, 1997.

[Coplien98] Coplien, J. O.: Software Design Patterns: Common Questions and Answers. In: Rising L., (Ed.), The Patterns Handbook: Techniques, Strategies, and Applications. Cambridge: University Press, 1998.

[Cunningham04] Cunnigham, W.: Proto Patterns. http://c2.com/cgi/wiki?ProtoPattern, 2004. Datum des Aufrufs: 09.02.2009.

[Cunningham06] Cunnigham, W.: Anti Patterns. http://c2.com/cgi/wiki?AntiPattern, 2006. Datum des Aufrufs: 14.11.2006.

[Cunningham08] Cunnigham, W.: Architects Dont Code. http://c2.com/cgi/wiki?ArchitectsDontCode, 2008. Datum des Aufrufs: 09.02.2009.

[Czichy01] Czichy, T.: Musterbasierte Systementwicklung - Eine empirische Untersuchung. http://www.emailhub.org/patternsurvey/, 2001. Datum des Aufrufs: 13.08.2009.

[Davydov01] Davydov, M.: Corporate Portals and e-Business Integration. New York [u.a.]: McGraw-Hill, 2001.

[Dias01] Dias, C.: Corporate portals: a literature review of a new concept in Information Management. International Journal of Information Management. Volume 21, Issue 4, August 2001, S. 269-287, 2001.

[Dietrich07] Dietrich, A.: Informationssysteme für Mass Customization. Wiesbaden: Deutscher Universitäts-Verlag. Zugl.: Dissertation Universität Hohenheim, 2007.

[Dikel01] Dikel, D.; Kane, D.; Wilson, J.: Software-Architecture - Organizational Principles and Patterns. New Jersey: Prentice Hall PTR , 2001.

[Dumke03] Dumke, R.: Web engineering. München [u.a.]: Pearson Studium, 2003.

[Dyson04] Dyson, P.; Longshaw, A.: Architecting Enterprise Solutions - Patterns for High-Capability Internet Based Systems. Chichester: John Wiley & Sons Ltd., 2004.

[Eichinger04] Eichinger, C.: Architektur von Web-Anwendungen. In: Kappel, G. et al. (Hrsg.): Web Engineering. dpunkt-Verlag. S. 77-100, 2004.

[Escalona04] Escalona, M.; Koch, N.: Requirements Engineering for Web Applications – A Comparative Study. Journal of Web Engineering, Vol. 2, No.3 (2004). S. 193-212, 2004.

[Essing01] Essing, N.: Anwendungen der Benutzermodelle aus der Künstlichen Intelligenz: Repräsentationsmöglichkeiten für Benutzerprofile. Münster: Institut für Wirtschaftsinformatik der Westfälischen Wilhelms-Universität , 2001.

[Evans04] Evans, E.: Domain-Driven Design: Tackling Complexity in the Heart of Software. Boston, MA: Addison-Wesley, 2004.

[Feldmann04] Feldmann, K.; Gergs, H.-J.; Slama, S.; Wirth, U.: Montage strategisch ausrichten - Praxisbeispiele marktorientierter Prozesse und Strukturen. Berlin [u.a.]: Springer Verlag, 2004.

[Fingar00] Fingar, P.; Kumar, H.; Sharma, T.: Enterprise E-Commerce. Tampa: Meghan-Kiffer Press, 2000.

[Finkelstein00] Finkelstein, C.; Aiken, P.: Building Corporate Portals with XML. New York: McGraw-Hill Companies,Inc., 2000.

[Fowler99] Fowler, M.: Refactoring. Boston, MA: Addison-Wesley, 1999.

[Fowler98] Fowler, M.; Scott, K.: UML konzentriert: die neue Standard-Objektmodellierungssprache anwenden. Bonn: Addison-Wesley-Longman, 19989.

[Gabler04] o.V.: Gabler-Wirtschafts-Lexikon. 16., vollständig. überarbeitete und aktualisierte Auflage. Wiesbaden: Gabler, 2004.

[Gabriel07] Gabriel, R.: A Pattern Definition. http://hillside.net/patterns/definition.html, 2007. Datum des Aufrufs: 09.02.2009.

[Gaedke99] Gaedke, M.: Wiederverwendung von Komponenten in Web-Anwendungen.. In: Turowski, K. (Hrsg.): Tagungsband des 1. Workshops Komponentenorientierte betriebliche Anwendungssysteme, Magdeburg, S. 31-36, 1999.

[Galic05] Galic, M. et al.: A Portal Composite Pattern Using WebSphere Portal V5. San Jose, Calif.: IBM International Technical Support Organization, 2004.

[Gelhaar95] Gelhaar, I.: Ein rechnergestütztes, mulitifunktionalse Arbeitsplatz-Informations- und Kommunikationssystem. Bochum, Ruhr-Univ., Dissertation , 1995.

[Gerber08] Gerber, A.: Methode zur Konzeption eines unternehmensspezifischen Qualitätsinformationssystems für kleinere Unternehmen. Hochschulschrift Technische Universität Chemnitz. Zugl.: Dissertation, 2008.

[Gerlach00] Gerlach, S.: Produktionsinformationssysteme für Teams - Flexibel und bedarfsgerecht mit der Intranet-Technologie. In: Antoni, C. H., Eyer, E., Kutscher J. (Hrsg.):Das flexible Un-ternehmen, Wiesbaden: Gabler-Verlag, 2000.

[Gerlach09] Gerlach, S.: IuK-Portal. In: Reinhart, G.; Spath, D. (Hrsg.): Auftragsprozesse in der kundenintegrierten Montage. Stuttgart: IRB-Verlag, S. 161–179, 2009.

[Ginige01] Ginige, A.; Murugesan, S.: Web engineering: an introduction. Multimedia, IEEE, Volume 8, Issue 1, Jan.-March 2001, S. 14 - 18 , 2001.

[Glander06] Glander, G.: Erweiterte Kompetenzentwicklung durch integrierte Arbeits-, Lern-, und Kommunikationsprozesse - Das Beispiel Auto5000 GmbH. In: Clement, U.; Lacher, M.: Produktionssysteme und Kompetenzerwerb. Stuttgart: Franz Steiner Verlag, S. 181-192., 2006.

[Goodyear00] Goodyear, M.: Enterprise System Architectures - Building Client/Server and Web-based Systems. Boca Raton: CRC Press, 2000.

[Großmann05] Großmann, M.; Koschek, H.: Unternehmensportale - Grundlagen, Architekturen, Technologien. Berlin [u.a.]: Springer Verlag, 2005.

[Günthner06] Günthner, W. A. et al.: Produktion individualisierter Produkte. In: Lindemann, U.; Reichwald, R.; Zäh, M. (Hrsg.): Individualisierte Produkte. Berlin [u.a.]: Springer-Verlag, S. 63-88., 2006.

[Gurzki03] Gurzki, T.; Özcan, N.: Unternehmensportale - Kunden, Lieferanten und Mitarbeiterportale in der betrieblichen Praxis. Stuttgart: Fraunhofer IRB Verlag, 2003.

[Gurzki06] Gurzki, T.: Ein Architekturmodell für Portale im Technischen Vertrieb. Heimsheim: Jost-Jetter Verlag. Zugl.: Dissertation Universität Stuttgart, 2006.

[Hacker08] Hacker, W.: Informationsflussgestaltung als Arbeits- und Organisationsoptimierung. Zürich: vdf Hochschulverlag, 2008.

[Hagen05] Hagen, M.: Definition einer Sprache zur Beschreibung von Prozessmustern zur Unterstützung agiler Softwareentwicklungsprozesse. http://lips.informatik.uni-leipzig.de/pub/showDoc.Fulltext/ dokument.pdf?lang=de&doc=2005-11&format=pdf&compression=&rank=1, 2005. Datum des Aufrufs: 09.01.2007.

[Hahsler01] Hahsler, M.: Analyse Patterns im Softwareentwicklungsprozess. http://wwwai.wu-wien.ac.at/~hahsler/research/diss/diss.pdf, 2001.

[Henney06] Henney, K.: Context Encapsulation - Three Stories, a Language, and Some Sequences. In: Proceedings of the Tenth EuroPloP Conference, Irsee. Konstanz: Universitätsverlag, 2006.

[Hinderer05] Hinderer, H.: Eine Vorgehensweise zur Erstellung von Informationssystemen für die zwischenbetriebliche Zusammenarbeit im Vertrieb technischer Produkte. Heimsheim: Jost-Jetter. Zugl.: Dissertation Universität Stuttgart , 2005.

[Hinderer03] Hinderer, H.; Gurzki, T.: Eine Referenzarchitektur für Software zur Realisierung von Unternehmensportalen. In: Reimer, U. et al. (Hrsg.): WM 2003: Professionelles Wissensmanagement - Erfahrungen und Visionen; GI-Edition - Lecture Notes in Informatics (LNI), P-28 Bonner Köllen Verlag, 2003.

[Hoffmann01] Hoffmann, C.: Das Intranet - Ein Medium der Mitarbeiterkommunikation. Konstanz: UVK Verlagsgesellschaft. Zugl.: Dissertation Universität Hohenheim, 2001.

[Huang03] Huang, G. Q.; Mak, K. L.: Internet Applications in Product Design and Manufacturing. Berlin [u.a.]: Springer Verlag, 2003.

[Huber00] Huber, T.: Business Networking Architekturen. Bamberg: Difo-Druck. Zugl.: Dissertation Universität St. Gallen., 2000.

[Jablonski04] Jablonski, S.; Petrov, I.; Meiler, C.; Mayer, U.: Guide to Web Application and Platform Architectures. Berlin [u.a.]: Springer Verlag, 2004.

[Johnson92] Johnson, R. E.: Documenting Frameworks using Patterns. In: Conference Proceedings on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA), Vancouver: October 18 - 22, 1992. New York: ACM, S. 63-76., 1992.

[Kammermeier09] Kammermeier, A.; Raith, C.: Praxisbeispiel Krones AG. In: Reinhart, G.; Spath, D. (Hrsg.): Auftragsprozesse in der kundenintegrierten Montage. Stuttgart: IRB-Verlag, S. 255–281, 2009.

[Kamthan08] Kamthan, P.: Using Patterns for Engineering High-Quality Web Applications. In: Brandon, D. (ed.): Software Engineering for Modern Web Applications: Methodologies and Technologies. Hershey, New York: Information Science Reference, S. 100-122, 2008.

[Kappel04] Kappel, G.; Pröll, B.; Reich, S.; Retschitzegger, W.: Web Engineering. Systematische Entwicklung von Webanwendungen. Heidelberg: dpunkt-Verlag, 2004.

[Keller99] Keller, W.: Patterns in der Praxis. Köln: VAA-Fachtagung. http://www.gdv-online.de/vaa/vaafe_html/tagung1/pattprax.pdf, 1999. Datum des Aufrufs: 09.02.2008.

[Keller02] Keller, W.: Enterprise Application Integration - Erfahrungen aus der Praxis. Heidelberg: dpunkt Verlag, 2002.

[Klose02] Klose, M.: Design Patterns für digitale Produkte im digitalen Wirtschaftsraum. Dissertation Universität St.Gallen, 2002.

[Kobsa01] Kobsa, A.: Personalised hypermedia presentation techniques for improving online customer relationships. Cambridge: Cambridge University Press, 2001.

[Kuba97] Kuba, R.: Informations- und kommunikationstechnische Integration von Menschen in der Produktion. Berlin [u.a.]: Springer-Verlag. Zugl.: Dissertation Technische Universität München, 1997.

[Kuhlen04] Kuhlen, R. et al.: Grundlagen der praktischen Information und Dokumentation. 5., völlig neu gefasste Ausg. , München: K. G. Saur Verlag, 2004.

[Lacher06] Lacher, M.; Clement, U.: Standardisierung von Arbeitsprozessen - Standardisierung der Kompetenzen?. In: Clement, U.; Lacher, M.: Produktionssysteme und Kompetenzerwerb. Stuttgart: Franz Steiner Verlag, S. 7-14., 2006.

[Lang07] Lang, S.: Durchgängige Mitarbeiterinformation zur Steigerung von Effizienz und Prozesssicherheit in der Produktion. Bamberg: Meisenbach Verlag, 2007.

[Lay05] Lay, G.; Maloca, S.: Aufgabenintegration - Abkehr vom Taylorismus?. Mitteilungen aus der Produktionsinnovationserhebung Nr. 36, Karlsruhe: Fraunhofer ISI., 2005.

[Linsenmaier99] Linsenmaier, T.: Eine Architektur verteilter Objekte zur Integration von Produktionsinformationssystemen. Berlin [u.a.]: Springer Verlag. Zugl.: Dissertation Universität Stuttgart, 1999.

[Lotter06] Lotter, B.; Wiendahl, H.-P. (Hrsg.): Montage in der industriellen Produktion - Ein Handbuch für die Praxis. Berlin [u.a.]: Springer Verlag, 2006.

[Lowe99] Lowe, D.; Hall, W.: Hypermedia & the Web. Chichester: John Wiley & Sons Ltd, 1999.

[Martin01] Martin, D.; Rodden, T. et al.: Finding Patterns in the Fieldwork. In: Prinz, W. et al. (Eds.): Proc. of the 7th European Conference on Computer Supported Cooperative Work, Bonn, September 16 - 20, 2001. Norwell: Kluwer Academic Publishers, S. 39-58., 2001.

[McCormick98] McCormick, H. W.: What is an AntiPattern (and why should I care) ?. http://www.antipatterns.com/briefing/sld003.htm, 1998. Datum des Aufrufs: 14.11.2006.

[Menascé00] Menascé, D.; Almeida, V.: Scaling for E-Business. New Jersey: Prentice Hall PTR, 2000.

[Mendes06] Mendes, E.; Mosley, N.: Web Engineering. Berlin, Heidelberg : Springer Verlag , 2006.

[Mertens04] Mertens, P.: Integrierte Informationsverarbeitung 1 - Operative Systeme in der Industrie. 14. Auflage, Wiesbaden: Gabler Verlag, 2004.

[Meszaros97] Meszaros, G.; Doble, J.: A pattern language for pattern writing. In: Martin, C. et al. (Eds.): Pattern Languages of Program Design 3. Boston, MA: Addison-Wesley, S. 529-574, 1997.

[Murugesan01] Murugesan, M.; Deshpande, Y. (Eds.): Web Engineering - Managing Diversity and Complexity of Web Application Development. Berlin [u.a.]: Springer Verlag, 2001.

[Neuschwinger03] Neuschwinger, A.: Multimediales, informationsmodellbasiertes Arbeitsplatz-Kommunikationssysteme. Aachen: Shaker Verlag. Zugl.: Dissertation Ruhr-Universität Bochum, 2003.

[Noble98] Noble, J.: Classifying relationships between object-oriented design patterns. Proc. Australian Software Engineering Conference, 9-13 Nov 1998. http://homepages.mcs.vuw.ac.nz/~kjx/papers/classify.pdf, 1998. Datum des Aufrufs: 09.02.2009.

[Oberbichler06] Oberbichler, E.: Personalizing knowledge portals. Saarbrücken: VDM Verlag Dr. Müller e. K. und Lizensgeber, 2006.

[Piller07] Piller, F. T.: Extreme Customization. Massachusetts: Proceedings of the MCPC 2007 World Congress on Mass Customization and Personalization, 2007.

[Pine94] Pine, B. J.: Maßgeschneiderte Massenfertigung: neue Dimensionen im Wettbewerb. Wien: Wirtschaftsverlag Ueberreuter, 1994.

[Porter05] Porter, R.; Coplien, J. O.; Winn, T.: Sequences as a basis for pattern language composition. Science of Computer Programming, Volume 56, Issues 1-2, New Software Composition Concepts, April 2005, S. 231-249., 2005.

[Porter92] Porter, M. E.: Wettbewerbsvorteile - Spitzenleistungen erreichen und behaupten. 3. Auflage, Frankfurt a.M: Campus Verlag, 1992.

[Prechelt02] Prechelt, L.; Unger-Lamprecht, B.; Philippsen, M.; Tichy, W. F.: Two controlled experiments assessing the usefulness of design pattern documentation in program maintenance. In: IEEE Transactions on Software Engineering Volume 28, Issue 6, June 2002 S. 595 - 606, 2002.

[Pressman09] Pressman, R.; Lowe, D.: Web Engineering: A Practitioner´s Approach. New York [u.a.]: McGraw-Hill, 2009.

[Puschmann03] Puschmann, T.: Collaboration Portale - Architektur, Integration, Umsetzung und Beispiele. Bamberg: Difo-Druck. Zugl.: Dissertation Universität St. Gallen, 2003.

[Raupach08] Raupach, A.: Ergonomische Gestaltung multimedialer Arbeitsmittel. Wissenschaftliche Schriftenreihe des Instsituts für Betriebswissenschaften und Fabriksysteme, Heft 76. Zugl.: Dissertation TU Chemnitz, 2008.

[Reichwald03] Reichwald, R.; Piller, F. T.; Tseng, M.: Leading Mass Customization and Personalization from an Emerging Stage to a Mainstream Business Model. München: Proceedings of the MCPC 2003, 2003.

[Reichwald06] Reichwald, R.; Piller, F. T.: Interaktive Wertschöpfung - Open Innovation, Individualisierung und neue Formen der Arbeitsteilung. 1. Auflage, Wiesbaden: Gabler, 2006.

[Riehle96a] Riehle, D.: Describing and Composing Patterns Using Role Diagrams. In: Mätzel, K.-U; Frei, H.-P. (Hrsg.): Proceedings of the Ubilab Conference '96. Konstanz: Universitätsverlag. S. 137-152, 1996.

[Riehle96b] Riehle, D.; Züllighoven, H.: Understanding and Using Patterns in Software Development. In: Theory and Practice of Object Systems 2, 1 (1996). S. 3-13, 1996.

[Rising00] Rising, L.: The Pattern Almanac 2000. Reading, MA: Addison Wesley Publishing Company, 2000.

[Röhricht01] Röhricht, J.; Schlögel, C.: cBusiness - Erfolgreiche Internetstrategien durch Collabortive Business am Beispiel my SAP.com. München [u.a.]: Addison-Wesley, 2001.

[Rösch00] Rösch, H.: Internetportal, Unternehmensportal, Wissenschaftsportal. In: Knorz, G.; Kuhlen, R. (Hrsg.): Informationskompetenz – Basiskompetenz in der Informationsgesellschaft. Proceedings des 7. Internationalen Symposiums für Informationswissenschaft, Darmstadt, 8. – 10.10. 2000. Konstanz: UVK Verlagsgesellschaft. S. 245 – 264, 2000.

[Rossi06] Rossi, G.; Schwabe, D.: Model-Based Web Application Development. In: Mendes, E.; Mosley, N.: Web Engineering. Berlin, Heidelberg: Springer-Verlag. S. 303-334, 2006.

[Rossi07] Rossi, G.; Pastor, O.; Schwabe, D.; Olsina, L. (Eds.): Web Engineering: Modelling and Implementing-Web Applications. London: Springer Verlag, 2007.

[Salingaros00] Salingaros, N.: The structure of pattern languages. In: arq: Architectural Research Quarterly, Volume 4, pp 149-162, 2000.

[Sand06] Sand, M.: Patternbasierte Verifikation objektorientierter Modelle - Methodik, Semantik und Verfahren. Nürnberg-Erlangen, Universität, Dissertation, 2006.

[Schäffer-Külz04] Schäffer-Külz, U.: Mitarbeiterportale und Self-Service-Systeme. Frechen, Datakontext-Fachverlag GmbH. Zugl.: Dissertation Universität Kiel, 2005.

[Scheer09] Scheer, A.-W.; Thomas, O. : Verfahren und Werkzeuge der Informationsmodellierung. In: Bullinger, H.-J., Spath, D.; Warnecke, H. J.; Westkämper, E. (Hrsg.): Handbuch Unternehmensorganisation. Berlin [u.a.]: Springer, S. 544-568., 2009.

[Schelp02] Schelp, J.; Winter, R.: Enterprise-Portals und Enterprise Application Intergration - Begriffsbestimmung und Integrationskonzeption. In: Meinhardt, S.; Popp, K. (Hrsg.): Enterprise-Portale & Enterprise Application Intergration. HMD - Praxis der Wirtschaftsinformatik. 39. Jahrgang, Heft 225. Heidelberg: dpunkt Verlag, 2002.

[Schenk02] Schenk, M.; Seelmann-Eggebert, R.: Mass Customization Facing Logistics Challenges. In: Rautenstrauch, C. et al. (Eds.): Moving into Mass Customization. Berlin [u.a.]: Springer, S. 41-57., 2002.

[Schirrmeister03] Schirrmeister, E.; Warnke, P.; Dreher, C.: Untersuchung über die Zukunft der Produktion in Deutschland: Sekundäranalyse von Vorausschau-Studien für den europäischen Vergleich. Karlsruhe: Fraunhofer ISI, 2003.

[Schmidt95] Schmidt, D. C.: Using design patterns to develop reusable object-oriented communication software. In: Communications ACM 38, 10 (Oct. 1995), S. 65-74, 1995.

[Schoder08] Schoder, D.; Grasmugg, S.: Mass Customization: Definition und Charakteristika. http://www.oscar.de/newsletter/nl_masscustom_definition.pdf, 2008. Datum des Aufrufs: 07.07.2008.

[Schulz04] Schulz, N.: Formulierung von Nutzerpräferenzen in Multimedia-Retrieval-Systemen. Magdeburg, Universität, Dissertation, 2004.

[Schütze99] Schütze, M.: Eine musterbasierte Methode zur domänenspezifischen Modellierung und Generierung von Softwarekomponenten. Aachen: Shaker Verlag 1999. Zugl.: Dissertation Universität Kaiserslautern, 1999.

[Schwarze98] Schwarze, J.: Informationsmanagement : Planung, Steuerung, Koordination und Kontrolle der Informationsversorgung im Unternehmen. Herne, Berlin: Verlag Neue Wirtschafts-Briefe, 1998.

[Schwinger04] Schwinger, W.; Koch, N.: Modellierung von Web-Anwendungen. In: Kappel, G. et al. (Hrsg.): Web Engineering. dpunkt-Verlag. S. 49 - 76., 2004.

[Sekine95] Sekine, K.: Goldgrupe Fertigung - schnelle Steigerung der Produktivität. Landsberg, Lech: Verlag Moderne Industrie, 1995.

[Siddle07] Siddle, J.: Creating Software Architecture using Pattern Sequences. http://hillside.net/europlop/workshops/G5.pdf, 2007. Datum des Aufrufs: 06.09.2007.

[Slama04] Slama, S.: Effizienzsteigerung in der Montage durch marktorientierte Montagestrukturen und erweiterte Mitarbeiterkompetenz. Bamberg: Meisenbach Verlag, 2004. Zugl.: Dissertation Universität Erlangen, 2004.

[Spath03a] Spath, D. (Hrsg.): Ganzheitlich produzieren - Innovative Organisation und Führung. Stuttgart: LOG_X Verlag, 2003.

[Spath03b] Spath, D. (Hrsg.); Wilhelm. S.: Information und Kommunikation i der Produktion - Ergebnisse einer Unternehmensbefragung. Stuttgart: IRB Verlag, 2003.

[Spath03c] Spath, D.; Gerlach, S.: Informationen für Teams in Montage und Produktion – Visualisierung, Kommunikation, Know-how Dokumen-tation. wt Werkstattstechnik online 93 (2003) 9, S. 624-626. Düsseldorf: Springer VDI Verlag, 2003.

[Spath03d] Spath, D.; Gerlach, S.; Stallkamp, J.: Kundenindividuelle Produktion - Informationen für Teams. ZWF 98 (2003) 10, S. 522-525, München: Carl Hanser Verlag, 2003.

[Spath05a] Spath, D.; Hinderer, H. (Hrsg.): Marktübersicht Portalsoftware 2005.. Stuttgart: Fraunhofer IRB Verlag, 2005.

[Spath05b] Spath, D.; Gerlach, S.; Scholtz, O.: Visuelle Entscheidungsunterstützung für Teamarbeit auf Shop-Floor-Ebene. In: Kaluza, B., Blecker, T. (Hrsg.): Erfolgsfaktor Flexibilität – Strategien und Konzepte für wandlungsfähige Unternehmen. Berlin: Erich Schmidt Verlag, S.367 – 382, 2005.

[Spath09a] Spath, D.; Gerlach, S.: Kundenindividuelle Auftragsinformationen in der Montage. wt Werkstattstechnik online 99 (2009) 4, S. 226-231. Düsseldorf: Springer VDI Verlag, 2009.

[Spath09b] Spath, D., Gerlach, S.: Utilisation of Task-specific Information in Custom-ised Production Environments. In: Proceedings of the 20th International Conference on Production Research ICPR – 20, 2 – 6. August 2009, Shanghai, China. 2009.

[Spath09c] Spath, D.: Grundlagen der Organisationsgestaltung. In: Bullinger, H.-J.; Spath, D.; Warnecke, H. J.; Westkämper, E. (Hrsg.): Handbuch Unternehmensorganisation. Berlin [u.a.]: Springer, 2009.

[Spath09d] Spath, D.; Reinhart, G. (Hrsg.): Auftragsprozesse in der kundenintegrierten Montage. Stuttgart: IRB-Verlag, 2009.

[Stelzer04] Stelzer, D.: Portale - Einführung und Überblick. In: Gentsch, P.; Lee, S.. (Hrsg.): Praxishandbuch PortalManagement - Profitable Strategien für Internetportale. Wiesbaden: Gabler Verlag. S. 3 - 26., 2004.

[Sternemann99] Sternemann, K.-H.: Team-Informationssystem. Magdeburg, Otto-von-Guericke-Univ., Diss., 1999.

[Stotko03] Stotko, C. M.; Piller, F. T. : Mass Customization und Kundenintegration - Neue Wege zum innovativen Produkt. . Düsseldorf: Symposion Publishing, 2003.

[Strauß02] Strauß, R.; Schoder, D.: eReality: Das e-business-Bausteinkonzept. Strategien und Erfolgsfaktoren für das e-business-Management. Frankfurt am Main: FAZ-Inst. für Management-, Markt- und Medieninformationen, 2002.

[Sullivan04] Sullivan, D.: Proven Portals - Best Practices for Planning, Designing, and Developing Enterprise Portals. Boston: Pearson Education, Inc., 2004.

[Syska06] Syska, A.: Produktionsmanagement. Das A — Z wichtiger Methoden und Konzepte für die Produktion von heute. Wiesbaden: Gabler Verlag, 2006.

[Takeda04] Takeda, H.: Das synchrone Produktionssystem - Just-in-time für das ganze Unternehmen. 4. Auflage, Frankfurt: Redline Wirtschaft, 2004.

[Thiels08] Thiels, N.: Personalisierung von technischen Maschinenbenutzungsoberflächen auf Grundlage einer erweiterten Analysephase, Kaiserslautern, Dissertation, 2008.

[Tichy98] Tichy, W.: A catalogue of general-purpose software design patterns. Proc. Technology of Object-Oriented Languages and Systems (TOOLS 23), IEEE Computer Society. http://www.ipd.uka.de/~tichy/publications/- Catalogue.doc, 1998. Datum des Aufrufs: 09.02.2009.

[Tidwell07] Tidwell, J.: Designing "Designing Interfaces". UPA 2007: Patterns - Blueprints für Usability. 11-15.06.2007. Austin, Texas: Usability Professional Association. http://jtidwell.net/talks/UPA2007_Designing_Designing_Interfaces.ppt, 2007. Datum des Aufrufs: 30.08.2008.

[Tseng05] Tseng, M.: Proceedings of the Interdisciplinary World Congress on Mass Customization and Personalization. Hong Kong: September 18-21, 2005.

[vanWelie00] van Welie, M.; van der Veer, G.; Eliens, A.: Patterns as Tools for User Interface Design. In: International Workshop on Tools for Working with Guidelines, S. 313-324, 7-8 October 2000, Biarritz, France, 2000. Datum des Aufrufs: 08.03.2007.

[Wellhausen05] Wellhausen, T.: User Interface Design for Searching - A Pattern Language. 10th European Conference on Pattern Languages of Programs, Irsee, Germany, 06th-10th July 2005. http://www.tim-wellhausen.de/-papers/UIForSearching.pdf, 2005. Datum des Aufrufs: 03.04.2007.

[Westkämper09] Westkämper, E.; Zahn, E. (Hrsg.): Wandlungsfähige Produktionsunternehmen - Das Stuttgarter Unternehmensmodell. Berlin [u.a.]: Springer Verlag, 2009.

[Wiendahl04] Wiendahl, H.-P.; Gerst, D.; Keunecke, L.: Variantenbeherrschung in der Montage. Berlin [u.a.]: Springer Verlag, 2004.

[Wikner07] Wikner, J.; Naim, M.; Rudberg, M.: Exploiting the Order Book for Mass Customized Manufacturing Control Systems With Capacity Limitations. IEEE Transactions on Engineering Management, Vol. 54, No. 1, S. 145-155., 2007.

[Wilhelm06] Wilhelm, S.: Verfahren zur Einführung eines internetbasierten Content Managements für Qualitätsregelkreise in der Produktion. Heimsheim: Jost-Jetter. Zugl.: Dissertation Universität Stuttgart, 2006.

[Wissen04] Wissen, M.: Eine Vorgehensweise zur systematischen Modellierung webbasierter Anwendungen. Heimsheim: Jost-Jetter Verlag. Zugl.: Dissertation Universität Stuttgart, 2005.

[Witte03] Witte, K.-W.; Vielhaber, W. (Hrsg.): Flexible und wirtschaftli-che Serienmontage – Wege zu zukunftsstabilen Montagesys-temen. Aachen: Shaker Verlag, 2003.

[Woukeu03] Woukeu, A.; Carr, L.; Wills, G.; Hall, W.: Rethinking Web Design Models: Requirements for Addressing the Content. Technical Report ECSTR-IAM03 - 002, Department of Electronics and Comnputer Science, University of Southampton, 2003.

[Zdun06] Zdun, U.: Systematic pattern selection using pattern language grammars and design space analysis. In: Software: Practice and Experience, Volume 37, No. 9, S. 983-1016, 2007, 2007.

[Ziemer04] Ziemer, S.: Anwendung der Metro-Map-Metapher auf die Gestaltung von Unternehmensportalen, München, Dissertation, 2004

[Zimmer95] Zimmer, W.: Relationships Between Design Patterns. In: Coplien, J.; Schmidt, D. C. (Eds.): Pattern Languages of Program Design. Reading: Addison-Wesley, S. 345-364., 1995.

9.1.4 Bildnachweis

Die Verwendung der Fotos erfolgt mit freundlicher Genehmigung der nachfolgend genannten Bildquellen:

Abbildung 2-5:	Festo AG.
Abbildung 2-6:	Krones AG
Abbildung 5-1:	Rittal GmbH & Co. KG
Abbildung 5-2:	Rittal GmbH & Co. KG
Abbildung 5-3, 5-13:	Laserschneiden #15182187 © Alterfalter; Sparks #8506419 © Andrey Kiselev; Werkzeug #16491272 © Ro; Lieferung per Post #9661401 © Thaut Images; New metal brilliant Spray gun #21374908 © Konstantin Kulikov; Elektriker am Schaltschrank #21381627 © Cachaco – alle Fotolia.com. Montagevorgang, Prüfung Schaltschrank - Rittal GmbH & Co. KG.
Abbildung 5-6:	Montageschiene, Schienenpositionen - Rittal GmbH & Co. KG
Abbildung 5-11:	Laserschneiden #15182187 © Alterfalter; Sparks #8506419 © Andrey Kiselev; New metal brilliant Spray gun #21374908 © Konstantin Kulikov – alle Fotolia.com. Montagevorgang, Prüfung Schaltschrank - Rittal GmbH & Co. KG.
Abbildung 6-4:	Krones AG
Abbildung 6-5:	Krones AG

9.2 Entwurfsmustersprache für produktionsnahe Informationsportale

	Analyse (A.)	
	A.Geschäftsprozesse (A.G.)	
OR	**Basic Business Process** Modellierung von Geschäftsprozessen, deren Zielsetzung und den verwendeten Ressourcen mit UML.	[Eriksson00] Muster
AND	**Process Interaction** Modellierung von Informationsaustausch und Interaktionsbeziehungen zwischen Geschäftsprozessen. **ADD** A.K.Asynchrone Benachrichtigung	[Eriksson00] Muster
AND	**Time-to-Consumer (Kundenentkopplungspunkt)** Modellierung von Geschäftsprozessen mit einer auftragsanonymen Vorproduktion und einer auftragsspezifischen Endfertigung.	[Eriksson00] Muster
AND	**Process Layer Supply** Modellierung von komplexen Ablauforganisationen in mehrstufigen Geschäftsprozessen mit unterstützenden Prozessen. **ADD** A.B.Bewegungsdaten.Produktionsauftrag	[Eriksson00] Muster
AND	**Process Layer Control** Modellierung von komplexen Aufbauorganisationen in mehrstufigen Geschäftsprozessen mit Berichtsstrukturen. **ADD** A.B.Stammdaten.Party	[Eriksson00] Muster
AND	**Action Workflow** Modellierung von Abstimmungs- und Verhandlungsaktivitäten in einem Geschäftsprozess. **ADD** A.B.Bewegungsdaten.Contracts **ADD** A.B.Temporale Daten	[Eriksson00] Muster
	A.Business Objekte (A.B.)	
	A.B.Stammdaten	
OR	**The Enterprise and its World / Party** Datenmodell aller in und mit einer Unternehmung agierenden Parteien, d. h. Mitarbeiter, Kunden, Lieferanten und deren Organisationsstruktur. Parteien werden in einer gemeinsamen Entität namens Party abgebildet. **USE** K.I.SL.Ausgabeelemente.Hierarchien darstellen **USE** I.T.Datenmapping.Hierarchie	[Hay96] [Eriksson00] Muster
OR	**Things of the Enterprise** Datenmodell aller materiellen Güter einer Unternehmung, d. h. Produkte, Materialien, Betriebsmittel etc. Alle Güter werden in einer gemeinsamen Entität abgebildet. **USE** K.I.SL.Ausgabeelemente.Hierarchien darstellen **USE** I.T.Datenmapping.Hierarchie	[Hay96] Muster
AND	**Addresses / Locations** Modellierung von Adressen und deren Zuordnung zu Organisationen. **USE** A.B.Temporale Daten.Historic Mapping	[Hay96] [Eriksson00] Muster
AND	**Things, Thing Types, and Categories** Modellierungsschema für Typdefinition und zugehörigen konkreten Instanzen.	[Hay96] Muster

	A.B.Bewegungsdaten	
OR	**Produktionsauftrag (Procedures and Activities)** Datenmodell für Handlungsvorgänge und geplante Aktivitäten einer Unternehmung, z. B. Produktionsaufträgen, Wartung etc. **USE** K.I.SL.Ausgabeelemente.Hierarchien darstellen **USE** I.T.Datenmapping.Hierarchie	[Hay96] Muster
OR	**The Laboratory** Datenmodell für die Laborbelegung und die Testplanung.	[Hay96] Muster
OR	**Material Requirements Planning** Datenmodell für die Planung von benötigten Materialien und sonstigen Ressourcen. **USE** A.B.Stammdaten.Things of the Enterprise	[Hay96] Muster
OR	**Process Manufacturing** Datenmodell zur Planung der kontinuierlichen Prozessfertigung. **USE** A.B.Stammdaten.Things of the Enterprise	[Hay96] Muster
OR	**Contracts** Datenmodell für Verträge einer Unternehmung, z. B. Rahmenverträge, Lieferantenverträge etc. **USE** A.B.Stammdaten.Party **ADD** A.P.Contentprofilierung	[Hay96] Muster
OR	**Product Data Management** Datenmodell zur Abbildung der Beziehungen von Ressourcen und Dokumenten. **ADD** A.P.Contentprofilierung	[Eriksson00] Muster
AND	**Observation** Datenmodell für die quantitative und qualitative Protokollierung von Ereignissen und Beobachtung.	[Fowler97] Muster
	A.B.Temporale Daten	
XOR	**Audit Log** Protokoll der Änderungen an einem Datenobjekt. **ADD** I.A.Unterstützende Funktionen.Portalmanagement.Activity Log **ADD** I.P.Programmstruktur.Interaktion.Service Layer	[Fowler06] Muster
XOR	**Effectivity** Angabe des Zeitraums der Gültigkeit eines Objekts als Attribut im Objekt.	[Fowler06] Muster
XOR	**Snapshot** Speicherung von Zuständen eines Datenobjekts zu einem bzw. mehreren Zeitpunkten. **ADD** I.P.Programmstruktur.Interaktion.Service Layer **ADD** K.V.Portalnavigation.Inhalte gliedern.Chronological Organization	[Fowler06] Muster
XOR	**Temporal Object** Versionierte Objekte mit einer expliziten Versionshistorie der Objekte. **ADD** K.V.Portalnavigation.Inhalte gliedern.Chronological Organization	[Fowler06] Muster
XOR	**Historic Mapping / Temporal Property** Darstellung von zeitabhängigen Assoziationen und Objektbeziehungen. **ADD** K.V.Portalnavigation.Inhalte gliedern.Chronological Organization	[Fowler97] [Fowler06] Muster

	A.Kommunikation (A.K)	
	A.K.Asynchrone Benachrichtigung	
OR	**Forum / Threaded Discussion / Message Board** Schwarzes Brett, offen für alle Benutzer. In der "threaded" Version mit Antwortmöglichkeit als Diskussionforum. **ADD** I.A.Unterstützende Funktionen.Benutzermanagement **USE** K.I.SL.Ausgabeelemente.Hierarchien darstellen **USE** I.T.Datenmapping.Hierarchie.Struktur **ADD** A.P.Contentprofilierung **ADD** A.K.Erfahrungsaustausch	[Schümmer07] [vanDuyne02] [vanWelie07] Muster
OR	**FAQ / Frequently Asked Questions** Sammlung häufig wiederkehrender Fragen mit einer Lösung. **ADD** A.P.Contentprofilierung	[Schümmer07] [vanDuyne02] [vanWelie07] Muster
OR	**Event Calender** Anzeige aller anstehende Ereignisse, Aktivitäten und Aufgaben ab dem aktuellen Datum mit einer Wahlmöglichkeit für das Datum. **ADD** I.A.Unterstützende Funktionen.Benutzermanagement **ADD** Datumseingabe **ADD** A.P.Contentprofilierung	[vanWelie07] Muster
AND	**Shared Annotation** Verwendung von Anwendungsobjekten (Artefakten) als virtuelles Register für Dokumente, Notizen etc. **ADD** K.V.Portalnavigation.Navigationsstrategie.UnifiedBrowsingHierarchy **ADD** K.I.SD.Ausgabeorganisation.Heterogene Info.Pop-Up-Window **USE** I.T.Datenmapping.Objektergänzung.UniversalDatalink	[Schümmer07] Muster
AND	**Spontaneous Collaboration** Möglichkeit eine Kommunikation direkt von einem Referenzobjekt oder Thema aus zu starten, das automatisch als Betreff eingesetzt wird. **ADD** I.T.Datenmapping.Objektergänzung.UniversalDatalink	[Schümmer07] Muster
AND	**Group** Kommunikation sowohl für Einzelpersonen als auch zwischen Benutzergruppen vorsehen. Benutzer können benannte Adresslisten bilden und wie Einzelpersonen adressieren. **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[Schümmer07] Muster
	A.K.Erfahrungsaustausch	
OR	**Community Conference / Community Building** Benutzergruppen mit ähnlichen Aufgaben und Interessen, die "Erfahrungskreise" bilden, zum Meinungsaustausch, zur Problemlösung etc. Auch in moderierter Form möglich. **USE** I.A.Unterstützende Funktionen.Benutzermanagement **ADD** A.K.Asynchrone Benachrichtigung	[vanDuyne02] [vanWelie07] Muster
OR	**Expert Finder** Suchen nach und Auffinden von Mitarbeitern mit ausgewiesener Kompetenz zu ausgewählten Themen. **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[Schümmer07] Muster
OR	**Abbreviations and Glossary** Glossar mit Begriffen zur Erklärung der Terminologie sowie der Vereinfachung, Vereinheitlichung und Präzisierung der Texterstellung.	[Schümmer07] Muster

AND	**Letter of Recommendation** Bewertung der Kompetenz der Autoren und Experten und Liste mit Empfehlungen weiterer kompetenter Experten. **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[Schümmer07] Muster
AND	**Feedback Loop** Benutzer sollen Autoren von Nachrichten einfach kontaktieren und Rückmeldung geben können, über die Nützlichkeit und Qualität der Beiträge. **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[Schümmer07] Muster
	A.Personalisierung (A.P.)	
	A.P.Personalisierungsverfahren	
	Personalized Content Anpassung des Informationsangebots einer Seite an das Benutzerverhalten und Wege zur Personalisierung aufzeigen **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[vanDuyne02] Muster
	A.P.Benutzerprofilierung	
OR	**Role Models** Rollen beschreiben Gruppen von Aufgaben und die hierfür zuständigen Personen **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[Hay96] Muster
AND	**Rollenrechte** Rechte werden Gruppen und Rollen zugeordnet. **USE** I.A.Unterstützende Funktionen.Zugriffskontrolle **USE** I.A.Unterstützende Funktionen.Benutzermanagement	[Yoder98] Muster
AND	**Organigramm-Benutzer und Rechte** Organe zugleich als Portalbenutzer zulassen und Rechte entlang der Organigrammstruktur vererben **USE** I.A.Unterstützende Funktionen.Zugriffskontrolle **USE** I.A.Unterstützende Funktionen.Benutzermanagement **USE** A.B.Stammdaten.Party	Regel
	A.P.Contentprofilierung	
OR	**Documents** Modell von Aufbau, Inhalt und Verwendung von Dokumenten.	[Hay96] Muster
OR	**Inverse-Pyramid Writing Style** Längere Texte wie Zeitungen immer dreigliedrig erfassen und darstellen: große Überschrift, fette Zusammenfassung und Langtext.	[vanDuyne02] Muster

	Konzeption (K.)	
	K.Vernetzung (K.V.)	
	K.V.Portalnavigation	
	K.V.Portalnavigation.Navigationsstrategie festlegen	
OR	**Browsable Content** Gliederung des Inhalts in einer vernetzten Struktur, in der ein Anwender zur gewünschten Information intuitiv navigieren und sich einen Überblick über das Informationsangebot verschaffen kann. **USE** K.V.Portalnavigation.Inhalte gliedern **USE** K.V.Navigationspräsentation.Navigationswege darstellen	[vanDuyne02] Muster
OR	**Search Action Module** Auffinden der gewünschten Information über eine assoziative Suchfunktion, die von jeder Seite aus erreichbar ist.	[vanDuyne02] Muster
OR	**Bookmarks (Collector)** Benutzer können Verweise zu favorisierten Seiten in einem überall zugänglichen Speicher (Collector) ablegen und daraus aufrufen **USE** A.Personalisierung (A.P.) **ADD** K.V.Navigationspräsentation.Navigationswege darstellen.Shortcut Box	[vanWelie07] Muster
AND	**Unified browsing hierarchy** Wenn ein Portal mehrere Funktionen (content, community, commerce) zu Sachinhalten anbietet, dann die Navigation nach den Inhalten gliedern und auf jeder Seite alle Funktionen zu einem Inhalt anbieten	[vanDuyne02] Muster
AND	**Multiple Ways to Navigate / Intriquing Branches** Mehrer Navigationswege vorsehen und beide Suchmotivationen "Intention" (gezielte Suche) und "Anreiz" (impulsive Neugier) bedienen	[vanDuyne02] [Tidwell06] Muster
AND	**Service Navigation (Meta Navigation)** Reservierter Bereich auf jeder Webseite mit inhaltsneutralen Navigationsmöglichkeiten, z. B. für Homelink, Impressum, Bookmarks. Suchen **ADD** A.Personalisierung (A.P.) **ADD** K.V.Navigationspräsentation.Navigationswege darstellen.Shortcut Box	[vanWelie07] Muster
AND	**Customizable Windows / Content Modules** Bereiche vorsehen, deren Inhalte Benutzer auswählen und konfigurieren können **USE** A.Personalisierung (A.P.)	[vanWelie07] [vanDuyne02] Muster
	K.V.Portalnavigation.Inhalte gliedern	
OR	**Hierarchcal Organization** Gliederung umfangreicher Inhalte in hierarchische Kategorien mit möglichst sprechenden Namen und ggf. Beispielen. **USE** K.V.Portalnavigation.Navigationsstrategie festlegen.Browsable Content	[vanDuyne02] Muster
OR	**Task-based Organization** Gliederung des Inhalts entsprechend dem Informationsbedarf für typische Aufgaben **ADD** A.P.Benutzerprofilierung.Role Models	[vanDuyne02] Muster
OR	**Alphabetical Organization** Gliederung des Inhalts nach dem Alphabet, sofern entsprechende Namen allgemein bekannt sind	[vanDuyne02] Muster

OR	**Chronological Organization** Gliederung des Inhalts in chronologischer Reihenfolge, ggf. in verfeinernden Abschnitten	[vanDuyne02] Muster
OR	**Popularity-based Organization** Gliederung des Inhalts nach seiner Popularität, wobei Popularitätsmaß (Aufrufe, Wertung …) und Periode deutlich erkennbar sein muss. **USE** I.U.Portalmanagement.ActivityLog	[vanDuyne02] Muster
OR	**Aspektorientierte Organisation (Faceted Navigation)** Gliederung des Inhalts in nach Aspekten, die ggf. überlappend sein können	[vanWelie07] Muster
	K.V.Portalnagivation.Navigationspfade festlegen	
AND	**Navigable Spaces / Hub and spoke** Aufteilung in Abschnitte (Spaces) mit jeweils einem Einstiegspunkt, Subnavigationspfaden und Rückkehr- bzw. Wechselmöglichkeiten.	[Tidwell99] [Tidwell06] Muster
XOR	**Main Navigation / Global Navigation** Wahlfreier Einstieg in jeden Hauptinhalt mittels einem permanent dargestellten Hauptmenü.	[vanWelie07] [Tidwell06] Muster
XOR	**One-window drilldown** Schrittweise Navigation eine Ebene vor oder zurück durch eine Hierarchie, wobei das sichtbare Menü jeweils vollständig sein Untermenü ersetzt.	[Tidwell06] Muster
XOR	**Pyramid** Zentraler Einstiegspunkt in geordnete, vertikal mit dem Einstiegspunkt vernetzten Unterpunkten, und einer zusätzlichen, horizontalen Schrittnavigation vor und zurück zum benachbarten Unterpunkt.	[Tidwell06] Muster
AND	**Go Back One Step** Möglichkeit zur vorigen Seite zu springen. **ADD** I.T.Interaktionsleistung.Aufrufhandling.Call-Stack	[Tidwell99] Muster
AND	**Home Link / Go Back to a Safe Place** Erstelle ein Element, das mit der Startseite verlinkt ist. Dieses Element muss auf jeder Seite vorhanden sein und dort oben platziert werden.	[vanWelie07] [Tidwell99] Muster
	K.V.Navigationspräsentation	
	K.V.Navigationspräsentation.Navigationswege darstellen	
XOR	**Navigation Bar / Inverted L Menu** Permanenter Zugriff auf die wichtigsten Navigationselementen in einer Text- oder Iconleiste am oberen und / oder linken Rand	[vanDuyne02] [vanWelie07] Muster
XOR	**Tab Rows** Horizontale Registerblätter mit einem hervorgehobenen aktuellen Blatt.	[vanDuyne02] Muster
XOR	**Double Tabs** Tabs, in zwei Reihen. Zweite Reihe enthält hierarchische Unterpunkte. **USE** K.V.Portalnavigation.Inhalte gliedern.Hierarchcal Organization	[vanWelie07] Muster
XOR	**Icon Menu** Menüdarstellung mit Piktogrammen. Beim Zeigen auf ein Piktogramm wird ein erklärender Text angezeigt. **ADD** K.I.SL.Ausgabeelemente.Graphiken darstellen.Data Tips	[vanWelie07] Muster
XOR	**Fly-out Menu** Hauptmenü als Leiste darstellen und Menüeinträge einblenden, bzw. nach unten aufklappen, wenn Benutzer auf Hauptmenü zeigt	[vanWelie07] Muster
XOR	**Retractable Menu** Menu seitlich am Fenster vorsehen. Auf Anforderung ausblenden und die gesamte Darstellungsfläche freigeben.	[vanWelie07] Muster

XOR	**Shortcut Box** Ausklappbares Menu mit häufig frequentierten Informationsangeboten	[vanWelie07] Muster
	K.V.Navigationspräsentation.Orientierungshilfen vorsehen	
OR	**Breadcrumbs** Stelle den aktuellen Pfad der Seite über alle Ebenen als Links dar. Der Pfad muss bei der Startseite beginnen **USE** K.V.Portalnavigation.Inhalte gliedern.Hierarchcal Organization	[vanWelie07] [Tidwell06] Muster
OR	**Repeated Menu** Bei langen Seiten eine erneute Navigationsmöglichkeit am unteren Fensterrand vorsehen.	[vanWelie07] Muster
OR	**Site Map / Map of Navigable Spaces** Eine strukturierte Übersicht der Inhalte, bzw. Webseiten eines Portals in graphischer oder textueller Form **USE** I.T.Interaktionsleistung.Aufrufhandling.Dynamische Menüs	[vanWelie07] [Tidwell99] Muster
OR	**Category Pages / Color-Coded Sections** Verschiedene Sektionen über das Layout anzeigen, z. B. Sektionsfarben, und mit einer einheitlichen Navigation verbinden	[vanDuyne02] [Tidwell06] Muster
OR	**Distinctive HTML Titles** Setze sinnvolle Fenstertitel (Windowtitle) für jede Seite, da Fenstertitel in Bookmarks, Suchmaschinen etc. auftauchen.	[vanDuyne02] Muster
	K.V.Objektnavigation	
	K.V.Objektnavigation.Suche	
XOR	**Table Filter** Filtermöglichkeiten der anzuzeigenden Entitäten in der Kopfzeile einer Tabelle anbieten, ggf. für mehrere Spalten **USE** I.T.Datenhandling.Transporthandling.LazyLoad **ADD** I.A.Unterstützende Funktionen.Benutzermanagement.Rule Storage	[vanWelie07] Muster
XOR	**Continuous Filter** Kontinuierliche Suche mit synchronisierter Anzeige der gefundenen Ergebnisse. Nach jeder Ergänzung oder Änderung des Suchfilters werden unmittelbar die Ergebnisse mit dem Suchfilter angezeigt und die übereinstimmenden Kriterien in den Ergebnissen markiert **USE** I.T.Datenhandling.Transporthandling.LazyLoad	[Laakso03] Muster
XOR	**Simple Search** Einfache Suchfunktion mit einem Textfeld zur Eingabe des Suchwortes, optional eine Filterfunktion und einen Button zum Starten der Suche. **ADD** Tabelle darstellen **ADD** K.I.SD.Ausgabeorganisation.Homogene Informationen	[vanWelie07] Muster
XOR	**Advanced Search** Wenn Benutzer einen speziellen Begriff suchen, stelle eine erweiterte Suchfunktion mit einer erweiterten Eingabemöglichkeit, einem Gültigkeitsbereich und Ausgabeoptionen zur Verfügung.	[vanWelie07] Muster
AND	**Organized search results** Suchergebnissen aufbereiten, Zusammenfassung anzeigen, gruppieren, Korrekturen für Schreibfehler in Suchtexten anbieten, weiterführende Suchmöglichkeiten etc.	[vanDuyne02] Muster
	K.V.Objektnavigation.Auswahl	
OR	**Klassen-Objektbrowser** Anzeige einer Übersicht aller Klassen und ihrer Objekten, die in einer Datenbank abgelegt sind.	Regel

	USE I.T.Datenhandling.Metadaten.Data Dictionary	
OR	**Rollenkontext** Einschränkung der Objekte auf einen rollenabhängigen Kontext des Anwenders, z. B. ZUSTÄNDIGE, ERSTELLTE, OFFENE Objekte etc. **USE** A.P.Benutzerprofilierung.Rollen	Regel
OR	**Objektgruppierung (View)** Gruppierung der Objekte nach einem Kriterium, Einschränkung der Auswahl auf eine Gruppe und Wahlmöglichkeit des Kriteriums. **USE** K.V.Portalnavigation.Inhalte gliedern	[vanWelie07] Muster
	K.Information (K.I.)	
	K.I.Portallayout	
	K.I.Portallayout.Raster festlegen	
OR	**Grid Layout / Tiled Working Surfaces** Verwendung eines einheitlichen Rasters für die grundlegende Anordnung typischer Informationsblöcke in Webseiten	[vanDuyne02] [vanWelie07] [Tidwell99] Muster
OR	**Above the Fold** Anzeige aller wichtigen Informationen in einem zentralen Bereich der auch bei geringer Monitorauflösung stets sichtbar ist.	[vanDuyne02] Muster
	K.I.Portallayout.Größe festlegen	
XOR	**Expanding-Width Screen Size / Liquid Layout** Navigationselementen feste Größen, restlichen Inhalte variable Größen zuweisen und somit Größenanpassung durch Browser zulassen.	[vanDuyne02] [vanWelie07] Muster
XOR	**Fixed-width screen Size** Darstellungselementen feste Größen zuweisen, insbesondere bei langen Texten um Lesen zu vereinfachen.	[vanDuyne02] Muster
	K.I.Seitendialog (K.I.SD)	
	K.I.SD.Eingabeorganisation	
OR	**Modal Panel** Seite oder Fenster, dessen Eingabe durch den Benutzer abgeschlossen sein muss, bevor andere Navigationen und Aktionen möglich sind. **USE** I.T.Interaktionsleistung.Aufrufhandling	[Tidwell06] Muster
OR	**Process Funnel / Wizard / Wizard** Bei mehreren notwendigen Arbeits- und Dialogschritten die auszuführenden und bearbeiteten Schritte anzeigen, unnötige Information vermeiden, Schrittnavigation vorsehen, jeden Schritt und Konsequenzen erklären. **ADD** K.V.Portalnagivation.Navigationspfade festlegen.Go Back one Step	[vanDuyne02] [vanWelie07] [Tidwell06] Muster
AND	**Sequence Map** Zeige bei mehreren notwendigen Arbeits- und Dialogschritten die Folge graphisch an, verbunden mit einer Positionsanzeige	[Tidwell06] Muster
	K.I.SD.Ausgabeorganisation	
	K.I.SD.Ausgabeorganisation.Heterogene Informationen	
XOR	**Stack of Working Surfaces / Tabs** Verteilung des Informationsangebots auf mehrere Kacheln mit Reitern zur Auswahl der anzuzeigenden Kachel.	[Tidwell99] [vanWelie07] Muster
XOR	**Central Working Surface** Verteilung des Informationsangebots auf mehrere Seiten und einer zentralen Seite, von der aus alle weiteren, sie ersetzenden, Seiten erreichbar	[Tidwell99] Muster

	sind. Rückkehr zur zentralen Seite mit den Hauptinformationen	
XOR	**Pop-Up Windows / Pile of Working Surfaces** Verteilung des Informationsangebots auf möglichst wenige Zusatzfenster für ergänzende Informationen.	[vanDuyne02] [Tidwell99] Muster
	K.I.SD.Ausgabeorganisation.Homogene Informationen	
XOR	**Two-Panel Selector / Overview Beside Detail / Overview by Detail** Permanente Inhaltsübersicht in einem festen Bereich, oft links der Detailansicht für einen ausgewählten und markierten Inhalt.	[Tidwell06] [Laakso] [vanWelie07] Muster
XOR	**Collapsible Panels** Mehrere Informationsbereiche, die unabhängig voneinander geöffnet und auf einen Platzhalter reduziert werden können	[vanWelie07] Muster
XOR	**Accordion** Variante von Collapsible Panels mit nur einem geöffneten Bereich. Zuvor geöffnete Bereich werden beim Wechsel automatisch geschlossen.	[vanWelie07] Muster
XOR	**Extras on Demand / Details on Demand** Unterscheidung des Informationsangebots in Hauptinformationen und Nebeninformationen, die nur bei Bedarf eingeblendet werden.	[Tidwell06] [vanWelie07] Muster
XOR	**Minesweeping** Matrix von Symbolen oder Piktogrammen, die ihre Bedeutung beim Zeigen auf ein Symbol aufdecken. Inverse Darstellung zu Extras on Demand: Bei Minesweeping belegen viele Elemente eine große Fläche, Details werden für ein Element klein dargestellt.	[vanWelie07] Muster
XOR	**Fisheye / Local Zooming / Inplace Replacement** Komprimierte Übersicht der gesamten, umfangreichen Informationsmenge mit einem positionierbaren Fokus zur punktuellen Vergrößerung der Darstellung oder Erweiterung des Informationsinhalts.	[Laakso] [Tidwell06] [vanWelie07] Muster
	K.I.Seitenlayout (K.I.SL)	
	K.I.SL.Eingabeelemente	
	K.I.SL.Eingabeelemente.Objektmethoden	
XOR	**Inline Tableactions** Für jedes Tabellenelement die möglichen Objektmethoden als Actionbuttons in der Tabellenzeile aufzeigen und ggf. ausführen **ADD** K.I.SV.Aktion auslösen **ADD** K.I.SV.Aktion auslösen.EnlargedClickarea	Regel
XOR	**Button Groups** Für verwandte Aktionen die Auslöser (z. B. Buttons) räumlich gruppiert anordnen. **ADD** K.I.SV.Aktion auslösen	[Tidwell06] Muster
XOR	**Action Panel** Größere Anzahl von möglichen Aktionen in einer permanent sichtbaren, strukturierten Liste anzeigen **ADD** K.I.SV.Aktion auslösen	[Tidwell06] Muster
	K.I.SL.Eingabeelemente.Objektassoziationen	
XOR	**List Builder (van Welie)** Darstellung der gesamten Liste mit Editierfunktionen für jede Zeile **USE** K.I.SL.Eingabeelemente.Objektmethoden.InlineTableactions	[vanWelie07] Muster
XOR	**Parts Selector / List Builder (Tidwell)** Darstellung der zuordenbaren und der zugeordneten Objekte in zwei Listen nebeneinander, mit der Möglichkeit Objekte zu transferieren.	[vanWelie07] [Tidwell06] Muster

XOR	**List Entry Row (List Entry View)** Darstellung der gesamten Liste mit einer Eingabezeile für neue Elemente oberhalb der Listeneinträge	[vanWelie07] Muster
	K.I.SL.Eingabeelemente.Objekthierarchie	
XOR	**Cascading Lists** Jede Hierarchiestufe durch eine Liste darstellen. Untergeordnete Stufen sind dabei in einer eigenen Liste rechts der Oberstufe dargestellt. Den Pfad durch Markieren der Elemente auswählen und eine Hierarchieänderung durch direkte Manipulation oder Buttons vorsehen	[Tidwell06] Muster
XOR	**Einstufige Pfadbearbeitung** Zeige zu einem Knoten den Pad zur Wurzel. Hierarchiebearbeitung durch hoch- oder tiefstufen des Knotens um eine Stufe ermöglichen. **USE** I.T.Datenmapping.Hierarchie.Struktur.Composite	Regel
	K.I.SL.Eingabeelemente.Nominale Werte	
XOR	**Pulldown Button** Aufklappbare Textliste zur Auswahl und Anzeige der gewählten Option	[vanWelie07] Muster
XOR	**Dropdown Chooser** Aufklappbares Auswahlfeld mit anklickbaren Elementen, die eine graphische, visuelle Darstellung der Optionen vermitteln.	[Tidwell06] Muster
	K.I.SL.Eingabeelemente.Skalare Werte	
XOR	**Good Defaults** Eingabefelder, wo immer möglich, mit einem sinnvollen Standardwert vorbelegen, den viele Benutzer übernehmen können.	[Tidwell06] Muster
XOR	**Input Prompt** Textfelder mit einer Frage hinsichtlich des einzugebenden Wertes vorbelegen, falls keine sinnvollen Defaultwerte ermittelbar sind.	[Tidwell06] Muster
XOR	**Fill-in-the-Blanks** Texteingabe mit einem oder mehreren Eingabefelder als Teil eines selbsterklärenden Lückentextes.	[Tidwell06] Muster
AND	**Forgiving Format** Interpretation von vielfältigen Eingabeformaten, z. B. Lokationen über Ortsname oder PLZ.	[Tidwell06] Muster
AND	**Date Selector** Datumseingabe mit einem grafischen, anklickbaren Kalender und einem Textfeld zur Anzeige des gewählten Datums.	[vanWelie07] Muster
	K.I.SL.Ausgabeelemente	
	K.I.SL.Ausgabeelemente.Tabellen darstellen	
OR	**Table Sorter / Sortable Table** In Tabellen eine Sortiermöglichkeit durch Anklicken der Überschrift vorsehen.	[vanWelie07] [Tidwell06] Muster
OR	**Alternating Row Colours / Row Striping** Bei langen Tabellen wird jeder zweiten Zeile eine andere Farbe für eine bessere Zeilenunterscheidung gegeben.	[vanWelie07] [Tidwell06] Muster
OR	**Paging / Sliding Scale** Teile eine große Anzahl gleichartiger Eintrage in mehrere Seiten auf, und biete eine Navigationsmöglichkeit am Beginn und am Ende der Eintragsliste. Typisch z. B. für Suchergebnisse **USE** I.T.Datenhandling.Transporthandling.LazyLoad	[vanWelie07] [Tidwell99] Muster
OR	**Ampeltabellen** Tabelle, die um eine Statusanzeige erweitert wird.	Regel

	K.I.SL.Ausgabeelemente.Hierarchien darstellen	
XOR	**Hierarchical Set (Explorer)** Hierarchiedarstellung mit einer Baumstruktur, deren Äste beliebig ein- und ausgeblendet werden können. **ADD** K.I.SD.Ausgabeorganisation.Homogene Informationen.OverviewBesideDetail **ADD** I.T.Datenmapping.Hierarchie.Struktur.Composite	[Tidwell99] Muster
XOR	**Tree Table** Hierarchiedarstellung in einer geordneten Tabelle mit Angabe der Stufe und Einrückung der Unterelemente. **ADD** I.T.Datenmapping.Hierarchie.Struktur.Stückliste	[Tidwell06] Muster
XOR	**Tree Map** Hierarchien durch eine zweidimensionale Karte mit Rechtecke darstellen, die untergliedert sind und zusätzlich durch Größe, Farbe, Schrifttyp etc. weitere Attribute visualisieren, z. B. Alter, Volumen.	[Tidwell06] Muster
XOR	**Pfad** Darstellung der Hierarchiebeziehung eines Knotens durch den Pfad zur Wurzel.	Regel
	K.I.SL.Ausgabeelemente.Graphiken darstellen	
XOR	**Thumbnail** Dem User soll ein Überblick gegeben werden. Dazu werden in einer Bildergalerie viele Bilder verkleinert dargestellt. Durch Anklicken eines Bildes wird dieses vergrößert dargestellt. **ADD** K.I.SD.Ausgabeorganisation.Heterogene Informationen.Pop-Up Windows	[vanWelie07] Muster
XOR	**Panning Navigator / Overview Plus Detail** Großformatige oder detailreiche Graphiken mit einem Übersichtsbild und einer Markierung des vergrößert dargestellten Ausschnitts anzeigen. Positionieren des Details durch Verschieben der Markierung.	[vanWelie07] [Tidwell06] Muster
AND	**Data Tips** Beim Zeigen auf eine Graphiken weitere Informationen und Daten zu dem Detail in einem Tooltip oder Textbox einblenden. **ADD** K.I.SD.Ausgabeorganisation.Heterogene Informationen.Pop-Up Windows	[Tidwell06] Muster
	K.I.SL.Ausgabeelemente.Beziehungen darstellen	
OR	**Important Message** Bei wichtigen Mitteilungen zeige die Nachricht sichtbar und wenn nötig mit Sound an.	[Tidwell99] Muster
OR	**Flag** Wichtige Inhalte mit einer eindeutigen Markierung anzeigen	[Schümmer07] Muster
OR	**Change Indicator** Neue oder geänderte Inhalte mit einer eindeutigen Markierung anzeigen **USE** I.U.Portalmanagement.Activity Log	[Schümmer07] Muster
OR	**Periodic Report** Benutzer periodisch über neue oder geänderte Inhalte informieren **USE** I.U.Portalmanagement.Activity Log	[Schümmer07] Muster
OR	**External Links / Outgoing Link** Bei externen Links anzeigen, dass die aktuelle Website verlassen wird, z. B. durch ein Icon vor dem Link	[vanDuyne02] [vanWelie07] Muster
	K.I.Seitenverarbeitung (K.S.V)	

	K.I.SV.Aktion auslösen	
XOR	**Action Buttons** Ein Button, der nach dem Anklicken die durch den Text des Buttons beschriebene Aktion auslöst.	[vanDuyne02] [vanWelie07] Muster
AND	**Enlarged Clickarea** Kleine Schaltflächen mit ebenfalls sensitiven, anklickbaren Bereichen umgeben, zur leichteren Bedienung.	[vanWelie07] Muster
AND	**Immediate Value Processing** Eingabe eines Wertes sofort auf syntaktische Korrektheit prüfen, z. B. Datumsformate	Regel
XOR	**Immediate Form Processing** Eingaben sofort verarbeiten, z. B. beim Verlassen von Feldern oder nach erfolgten Auswahlen aus Komboboxen. Anwendbar bei einfach strukturierten Dialogen mit wenigen, unabhängigen Eingabefeldern, z. B. Filterdialogen. Vermeidet Extraklicks zum Bestätigen der Eingabe.	Regel
	K.I.SV.Eingabewerte kontrollieren	
OR	**Give a Warning** Warnung, falls eine Aktion evtl. unbeabsichtigte Konsequenzen hat, mit der Möglichkeit zum Abbrechen oder Fortsetzen. **ADD** K.I.SD.Ausgabeorganisation.Heterogene Informationen.Pop-Up Windows **USE** K.I.SD.Eingabeorganisation.Modal Panel	[Griffiths98] Muster
OR	**Structured Text Entry / Unambigous Format / Preventing Errors / Show The Format Required** Verhindere eine falsche Benutzereingabe durch klare Angaben, was in das leere Feld geschrieben werden soll (durch ein Label neben dem Text). Ebenfalls kann eine feste Anzahl von leeren Stellen verwendet werden und gebe die richtigen Felder an, zum Beispiel bei einem Datum für den Tag zwei Leerstellen.	[Tidwell99] [vanWelie07] [vanDuyne02] [Griffiths98] Muster
OR	**Reality Check / Think Twice** Vor dem Durchführen einer Aktion soll erklärt werden, was sie bewirkf, beziehungsweise welche evtl. negativen Konsequenzen dies nach sich zieht. Anschließend die Durchführung vom Benutzer bestätigen lassen, mit der Möglichkeit des Abbruchs. **USE** K.I.SD.Eingabeorganisation.ModalPanel **ADD** K.I.SD.Ausgabeorganisation.Heterogene Informationen.Pop-Up Windows	[Tidwell99] [Griffiths98] Muster
	K.I.SV.Rückmeldung geben	
OR	**Echo Back / Interaction Feedback** Erfolg oder Mißerfolg von Aktionen und dabei evtl. vorgenommene Korrekturen an Format oder Wert von Eingaben sofort rückmelden **ADD** K.I.SD.Eingabeorganisation.ModalPanel	[Cunningham94] [Griffiths98] Muster
OR	**Same-Page Error Messages / Input Error Message / Meaningfull Error Messages** Fehlermeldungen in der verursachenden Seite, möglichst unmittelbar neben der Fehleingabe anzeigen, beschreiben worin das Problem besteht und wie der Fehler gelöst werden kann.	[Tidewll06] [vanWelie07] [vanDuyne02] Muster
OR	**Processing Page / Show Computer is Thinking** Bei längerer Verarbeitungsdauer die andauernde Aktivität anzeigen: 1. Bei kalkulierbarer Zeit, die ungefähre Dauer. 2. Bei nicht kalkulierbarer Zeit, aber verschiedenen Phasen, die abgearbeitet und noch anstehenden Phasen. 3. Ansonst die Anzahl der verarbeiteten Einheiten.	[vanWelie07] [Griffiths98] Muster

	Implementierung (I.)	
	I.Portalarchitektur (I.P.)	
	I.P.Programmstruktur	
	I.P.Programmstruktur.Gesamtsystem	
XOR	**MVC - Model-View-Control** Gliedert die Architektur, bzw. verteilt Benutzerinteraktionen auf 3 Rollen: Model - Domänendaten + Applikationslogik; View - Datendarstellung; Controll - Eingabeverarbeitung. Die Rollen sind keine Schichten. View kann direkt auf Model zugreifen für Datenausgabe.	[Buschmann98] [Fowler02] Muster
XOR	**PAC - Presentation - Abstraction – Control** Definiert für eine interaktive Anwendung eine Struktur, durch die Verwendung von Agenten. Diese Agenten sind hierarchisch angeordnet und kooperieren miteinander. Jeder Agent besitzt die Komponenten Präsentation, Abstraktion und Kontrolle und ist für einen Anwendungsbereich zuständig.	[Buschmann98] Muster
	I.P.Programmstruktur.Datenpersistierung	
XOR	**Gateway** Ein Objekt, das den Zugriff auf externe Systeme und Quellen kapselt. **ADD** I.P.Programmstruktur.Interaktion.Transaction Skript	[Fowler02] Muster
XOR	**Data Mapper** Eine Vermittlungsschicht zwischen Applikationsobjekten (Domain Model) und einer relationalen DB, die für alle DB-Zugriffe und die erforderliche Datenkonvertierung zuständig sind.	[Fowler02] Muster
AND	**Two Layer Persistency Subsystem** Verwendung einer Zweischichten-Architektur für den Zugriff auf relationale Datenbanken. Die erste Schicht stellt alle objektorientierten Sichten bereit, der zweite Schicht regelt den Zugriff auf die Datenbanken	[Keller98] Muster
	I.P.Programmstruktur.Präsentation	
XOR	**Template view** Dynamische Inhalte werden in ein vorgegebenes Template ergänzt.	[Fowler02] Muster
XOR	**Transform view** Dynamische Inhalte werden mittels Stylesheets formatiert.	[Fowler02] Muster
ADD	**Two step view** Daten werden zuerst in einer logischen Seite angeordnet, die in einem zweiten Schritt zu einer anzeigenden Seite endgültig formatiert wird.	[Fowler02] Muster
	I.P.Programmstruktur.Interaktion	
XOR	**Transaction Script** Die Applikationslogik für einen Geschäftsvorfall, der zugleich eine Interaktion, ein Anwendungsfall und eine Transaktion ist, wird in einer Prozedur gebündelt. Sie kann zum Beispiel Teil einer Server Page sein. **ADD** I.P.Programmstruktur.Datenpersistierung.Gateway	[Fowler02] Muster
XOR	**Table Module** Ein schlankes OO-Modell der Applikations-Logik, das für jede Tabelle der relationalen Datenstruktur eine Klasse und genau ein instantiiertes Objekt vorhält, das die gesamte Datenverarbeitung koordiniert und vornimmt.	[Fowler02] Muster
XOR	**Domain Model** Ein vollständiges OO-Modell der Applikationslogik, das Daten und Methoden in SW-Objekten vorhält.	[Fowler02] Muster
AND	**Service Layer** Eine Schicht, die höherwertige Dienste für mehrfach benötigte Applikati-	[Fowler02] Muster

	onsfunktionen anbietet und auf Grundfunktionen der Applikation aufsetzt. Somit wird die Applikationslogik in zwei Ebenen geteilt	
	I.P.Systemplattform	
	Portalarchitekturen orientieren sich an den verwendeten Systemplattformen, die häufig Entwurfsmuster für ihre korrekte Umsetzung kennen. In der vorliegenden Arbeit sind keine Entwurfsmuster berücksichtigt, um Abhängigkeiten von Plattformen zu vermeiden. Die Aufgabe ist als Platzhalter vorgesehen, für Muster im konkreten Einsatzfall.	
	I.Anwendungsfunktionen (I.A.)	
	I.A.Spezielle Anwendungen	
OR	**Visitor** Trennt komplexe Operationen von verzweigten Objektstrukturen (Graphen), ermöglicht die Wiederverwendung der Operationen und hält sie unabhängig von den den Objektstrukutren	[Gamma96] Muster
OR	**Mediator** Koordiniert die Kommunikation von Objekten in verzweigten Objektstrukturen (Graphen)	[Gamma96] Muster
OR	**Proxy** Platzhalter für konkrete Objekte in einer entfernten Umgebung oder einem nicht benötigten Zustand	[Gamma96] Muster
	I.A.Unterstützende Funktionen	
	I.A.Unterstützende Funktionen.Zugriffskontrolle	
OR	**Check Point** Erstelle eine zentrale Instanz zur Überprüfung von Zugriffsrechten Für jede Sicherheitspolice wird ein Objekt angelegt, das die Zugriffsrechte beschreibt. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung **USE** I.T.Datenhandling.Zugriffsobjekt	[Yoder98] Muster
OR	**Secure Access Layer** Ein Sicherheitsschicht, die als Vermittler zur den Sichererungsmechanismen des Betriebssystems und anderer gesicherter Systeme verwendet werden kann. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung **USE** I.T.Datenhandling.Zugriffsobjekt	[Yoder98] Muster
	I.A.Unterstützende Funktionen.Portalmanagement	
OR	**Logging** Führen eines Protokolls für auszuwählende Systemereignisse **USE** I.T.Datenhandling.Zugriffsobjekt	Regel
OR	**Activity Log** Protokolliert Aktivitäten der Benutzer, ihre Zugriffe auf Webseiten, bzw. auf einzelne Objekte **USE** I.T.Interaktionsleistung.Aufrufhandling? **USE** I.T.Datenhandling.Zugriffsobjekt	[Schümmer07] Muster
	I.A.Unterstützende Funktionen.Benutzermanagement	
OR	**Registration / Sign-In** Dialog vorsehen zur Benutzeranmeldung, Zugangsbeantragung für neue Benutzer und zum Erfragen vergessener Passwörter **USE** I.T.Interaktionsleistung.Sitzungsverwaltung	[vanWelie07] [vanDuyne02] Muster
OR	**Account Management** Eine zentrale Seite vorsehen, mit allen über einen Benutzer gespeicherten	[vanDuyne02] Muster

	Daten und ggf. der Möglichkeit zur Anpassung der Daten	
OR	**Collector** Ermögliche dem Benutzer eine Liste mit Seiten (Bookmarks), die er zu einem späteren Zeitpunkt besuchen möchte. Stelle hierfür auf jeder Seite einen Link zum Hinzufügen dieser Seite und einen Link, um zu den Bookmarks zu gelangen, zur Verfügung.	[vanWelie07] Muster
XOR	**Rule Storage** Wenn User die selben Daten immer wieder abfragen möchte, so soll er seine Abfrage (Kriterien und Werte) speichern können. **USE** I.T.Datenhandling.Metadaten.Repository	[Laakso] Muster
XOR	**Data Storage** Wenn User die selben gesuchten Daten immer wieder betrachten möchte, so soll er seine Abfrageergebnisse speichern können. **USE** I.T.Datenhandling.Metadaten.Repository	[Laakso] Muster
	I.Technische Funktionen (I.T.)	
	I.T.Datenmapping	
	I.T.Datenmapping.Vererbung	
XOR	**Single table inheritance / One Inheritance Tree One Table** Alle Unterklassen werden auf eine relationale Tabelle abgebildet, die alle Unterklassenfelder für alle Klassen enthält (platzineffizient).	[Fowler02] [Keller98] Muster
XOR	**One Inheritance Path One Table** Jeder Ast von Unterklassen wird auf eine relationale Tabelle abgebildet	[Keller98] Muster
XOR	**Concrete table inheritance / One Class One Table** Jede Unterklasse wird auf eine dezidierte relationale Tabelle abgebildet	[Fowler02] [Keller98] Muster
	I.T.Datenmapping.Assoziationen	
OR	**Identity Field** Speichert den DB-Datensatzschlüssel in einem ID-Attribut, um die korrekte Zuordnung von Objekten im Hauptspeicher und DB-Entitäten zu gewährleisten.	[Fowler02] Muster
AND	**Foreign Key Mapping** Assoziationen zwischen Objekten werden durch Kopieren der Schlüsselfelder auf DB-Entitäten übertragen.	[Fowler02] Muster
AND	**Association Table Mapping** Speicherung von mehrwertigen Assoziationen zwischen Objekten mittels Relationstabellen für die Schlüsselpaare.	[Fowler02] Muster
OR	**Serialized LOB** Abbildung von Graphen in ein serialisiertes Objekt, das in einem DB-Feld (LOB, large object) gespeichert wird. **ADD** I.T.Datenhandling.Transporthandling.Serializer	[Fowler02] Muster
	I.T.Datenmapping.Komplexe Attribute	
OR	**Embedded Value** Abbildung von leichtgewichtigen Objekten als Feldgruppen der DB-Entität, z. B. Datums- oder Währungsobjekte.	[Fowler02] Muster
	I.T.Datenmapping.Hierarchie	
	I.T.Datenmapping.Hierarchie.Struktur	
XOR	**Composite** Abbildung der Hierarchie durch Verweisattribut, das von den Söhnen auf den jeweiligen Vater zeigt	[Gamma96] Muster

	ADD I.T.Datenmapping.Hierarchie.Effizientes Auslesen	
XOR	**Stückliste** Stellt eine "flache" Speicherung von Hierarchien dar, durch Angabe von laufender Nummer und Stufe	Regel
XOR	**Nested Sets** Stellt eine "flache" Speicherung von Hierarchien dar, durch Vergabe von laufenden Knotennummern bei einer Depth-First-Reihenfolge	[Celko04] Protopattern
	I.T.Datenmapping.Hierarchie.Effizientes Auslesen	
OR	**Wurzelverweis** Jeder Knoten zeigt auf die Wurzel um alle Baumelemente in einer Datenbankanfrage auslesen zu können	Regel
OR	**Sohnflag** Jeder Knoten führt ein Flag, ob Söhne vorhanden sind. Reduziert die Anzahl notwendiger Selects beim Durchwandern	Regel
OR	**Path Enumeration** Jeder Knoten führt ein Attribut mit dem vollständigen Pfad bis zur Wurzel. Ermöglicht Auslesen eines Teilbaums **USE** I.T.Datenmapping.Assoziationen.SerializedLOB	[Celko04] Protopattern
	I.T.Datenmapping.Objektergänzung	
OR	**Universal Link** Link zu beliebigen Entitäten (Tabellen) bzw. Verweise zu beliebigen Objekten. **USE** I.T.Datenhandling.Metadaten.Data Dictionary	Regel
OR	**Objektbild** Speichern eines Bildes zu einem beliebigen Objekt im Portal. **USE** I.T.Datenhandling.Metadaten.Data Dictionary **ADD** I.P.Programmstruktur.Interaktion.Service Layer	Regel
OR	**Decorator** Dynamisches Ergänzen von Objekte mit neuer Funktionalität	[Gamma96] Muster
OR	**History** Führen einer Objekthistorie im Portal. **USE** I.T.Datenhandling.Metadaten.Data Dictionary **ADD** I.P.Programmstruktur.Interaktion.Service Layer	Regel
	I.T.Datenhandling	
	I.T.Datenhandling.Metadaten	
OR	**Data Dictionary** Eine Aufstellung der verwendeten Datenstrukturen, ihrer Attribute und Beziehungsinformationen. Häufig auch Angbaen zu Eventhandlern.	[Mahemoff06] Muster
AND	**Metadata Mapping** Generische Routinen für DB-Zugriff und Datenkonvertierung, ausgehend von Metadaten über die Datenstrukturen.	[Fowler02] Muster
AND	**Repository** Vermittelt zwischen Applikation und einer Schicht zur Datenkonvertierung. Akzeptiert Abfragekriterien und stellt der Applikation die gefundenen Objekte in Form einer Kollektion zur Verfügung. Für den Applikationscode entspricht das Repository ungefähr einer OO-DB. **USE** I.P.Programmstruktur.Interaktion.Domain Model	[Fowler02] Muster
	I.T.Datenhandling.Koordination Nebenläufigkeit	
XOR	**Pessimistic locking**	[Fowler02]

	Wenn ein Benutzer eine Datei (Objekt, Entität) auscheckt, wird diese auf dem Server gelockt (gesperrt) und blockiert den Zugriff von anderen Benutzern. Vorteil: Es kann zu keinem Konflikt bei der Entwicklung kommen. Nachteil: Andere Benutzer werden in ihrer Arbeit gestoppt, wenn sie diese Datei benötigen. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung	Muster
XOR	**Optimistic locking** Beim Auschecken einer Datei wird kein Lock gesetzt. Viele Benutzer können die Datei auschecken. Beim Schreiben wird auf Konflikte geprüft, und ggf. der Schreibvorgang zurückgewiesen durch einen Lock. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung	[Fowler02] Muster
ADD	**Implicit lock** Locks werden von einem Framework oder einer DB-Zugriffsschicht automatisch gesetzt.	[Fowler02] Muster
	I.T.Datenhandling.Transporthandling	
OR	**Unit of work** Führt ein Verzeichnis aller von einer Applikation geänderten Objekte und koordiniert die Datenbank-Änderungen. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung	[Fowler02] Muster
OR	**Identity Map** Führt ein Verzeichnis aller von der DB gelesenen Objekte und verhindert mehrfaches Laden derselben Objekte. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung	[Fowler02] Muster
OR	**Lazy load** Lädt vernetzte Objekte zum tatsächlichen Bedarfszeitpunkt und nicht bereits mit dem verweisenden Objekt. Lazy load stellt ein Objekt zur Verfügung, das nicht alle Daten enthält, aber die Zugriffsmöglichkeit kennt. **USE** I.T.Interaktionsleistung.Sitzungsverwaltung	[Fowler02] Muster
OR	**Serializer** Überführt beliebige, komplexe Objektstrukturen in lineare Speicherstrukturen, z. B. Textfiles oder relationale DB und abstrahiert dabei sowohl vom Speichermedium als auch der Objektklasse	[Riehle97] Muster
	I.T.Datenhandling.Zugriffsobjekt	
XOR	**Table Data Gateway** Ein Objekt, das als Schnittstelle zu einer Tabelle oder einer relationalen Datenbank agiert und alle Zeilen der Tabelle verarbeitet. Das Objekt kapselt den gesamten SQL-Code für den Zugriff auf die repräsentierte Tabelle.	[Fowler02] Muster
XOR	**Row Data Gateway** Ein Objekt, das als Schnittstelle zu einer Entität einer relationalen DB agiert. Das Objekt kapselt den Zugriff auf die repräsentierte Zeile einer Tabelle.	[Fowler02] Muster
OR	**Active Record** Ein Objekt, das als Schnittstelle zu einer Entität einer relationalen DB agiert und den Zugriff auf die repräsentierte Zeile einer Tabelle kapselt. Bietet zudem Funktionen der Applikationslogik (ggs. zu Row Data Gateway).	[Fowler02] Muster
OR	**Query Object / Data Source Object** Ein Objekt, das eine DB-Operation, das heißt eine SQL-Abfrage repräsentiert und so von der Datenbank-Struktur als auch von den SQL Details abstrahiert.	[Fowler02] [Welicki97] Muster

AND	**Abstract Cursor** Eine Klasse mit einem Cursorkonzept für den Datenzugriff unabhängig von der konkreten Zugriffsmethode. Anwendungsfunktionen können somit immer mit einem Cursor auf Relationsmengen zugreifen	[Welicki97] Muster
AND	**Base Data Acess Object** Eine Basisklasse für alle Gateways und Cursorklassen. Die Basisklasse stellt die elementaren Zugriffsmethoden bereit und zentralisiert die hierfür erforderliche Realisierung.	[Welicki97] Muster
	I.T.Interaktionsleistung	
	I.T.Interaktionsleistung.Sitzungsverwaltung	
XOR	**Client session state** Speichert Sitzungsinformation auf dem Client	[Fowler02] Muster
XOR	**Server session state** Speichert die Sitzungsinformation auf dem Server.	[Fowler02] Muster
XOR	**Database session state** Speichert Sitzungsinformation in der Datenbank.	[Fowler02] Muster
AND	**Effective Serve-Side Session Management** Anforderungen an einen serverseitigen Sessionstate bestehen hinsichtlich des benötigten Speichers und des Rechenaufwands zur Beschaffung der Sessiondaten. Muster zeigt die notwendige Balance zwischen den Anforderungen	[Rüping07] Muster
	I.T.Interaktionsleistung.Aufrufhandling	
XOR	**Page Controller** Ein Objekt, das den Aufruf genau einer Webseite oder einer Interaktion abhandelt und koordiniert. Dazu gehört das Auslesen von Parametern aus der URL, Laden der Domänendaten und Auswahl der Ausgabesicht.	[Fowler02] Muster
XOR	**Front Controller** Ein Controller, der alle Seitenaufrufe einer Website abwickelt und anhand der Parameter entscheidet, welche Webseite zurück geliefert wird.	[Fowler02] Muster
OR	**Dynamische Menüs** Navigationsstruktur für die Webseiten in Form eiiner hierarchischen Menütabelle in der DB abbilden. **ADD** I.T.Datenmapping.Hierarchie	Regel
AND	**Call-Stack** Realisiert Rücksprungfunktion, ermöglicht Rückgabe von Ausgabeparameter an aufrufende Seite	Regel
	I.T.Interaktionsleistung.Lastverteilung	
OR	**Disciplined Client-Side Interaction** Interaktionsreiche Anwendungen sind reagibler mit clientseitigen Interaktionen (Ajax), aber stärker abhängig von Browsereinstellungen. Clientseitige Interaktionen daher nur zum Aktualisieren von Daten verwenden. Komplexe Navigationsschritte immer serverseitig durchführen.	[Rüping07] Muster
OR	**Caching** Erzeuge einen Zwischenspeicher, in dem alle erzeugten Objekte gesammelt werden und nicht mehr verwendete Objekte wieder gelöscht werden	[Anastopoulos01] Muster
OR	**Pooling** Viele Clients verwenden die vom Server bereitgestellten Ressourcen. Diese müssen bei jedem Zugriff neu initialisiert werden. Um dies zu vermeiden, wird ein Pool mit Objekten, die vielfach verwendet werden, eingesetzt. In diesem Pool sind diese Objekte bereits initialisiert.	[Anastopoulos01] Muster

Tabelle 9-1: Entwurfsmustersprache für produktionsnahe Informationsportale

9.3 Beispiel Entwurfsmuster DECORATOR

Nachfolgend wird das Entwurfsmuster DECORATOR in kompakter Form wiedergegeben. Das Entwurfsmuster ist Gamma et al. entnommen und wird dort auf insgesamt 13 Seiten ausführlich dargestellt und anhand von Beispielcode erläutert [Gamma96, S. 199ff].

- **Name** – DECORATOR (DEKORIERER)
- **Problem und Kontext** – Dynamische Erweiterung einzelner Objekte auf vielfältige Art und Weise, ohne die Objektklasse selbst zu ändern. Typische Anwendungen bestehen in grafischen Benutzungsschnittstellen, wo Basisobjekte mit Zusatzelementen „dekoriert" werden. Abbildung 9-1 zeigt ein Beispiel hierfür. Eine Textanzeige soll mit einem Rahmen und Rollbalken dekoriert werden, sodass eine positionierbare Anzeige von Textabschnitten entsteht.

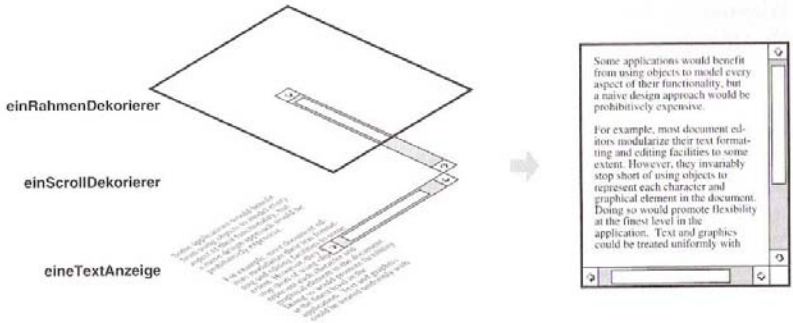


Abbildung 9-1: Anwendungsbeispiel für einen Decorator. Quelle: [Gamma96, S. 200]

- **Lösung** – Definition einer Hüllklasse für die zu dekorierende Komponente. Die Hüllklasse bildet die Schnittstelle der zu dekorierenden Klasse nach und fügt weitere Funktionen hinzu. An einem DECORATOR sind somit mindestens drei Klassen beteiligt, s. Abbildung 9-2: die zu dekorierende Komponente, eine Dekorationsklasse mit erweiternden Funktionen und eine Hüllklasse für die gemeinsame Schnittstelle. Für unterschiedliche Dekorationsfunktionen kann die Dekorationsklasse Unterklassen aufweisen. Zur Laufzeit werden Operationsaufrufe vom Dekorationsobjekt an das dekorierte Objekt weitergeleitet, wobei zuvor oder hernach erweiternde Dekorationsfunktionen ausgeführt werden. Die Aggregationsbeziehung vom Dekorierer zur Hülle erlaubt mehrstufige Dekorationen, wie im Objektdiagramm rechts gezeigt.

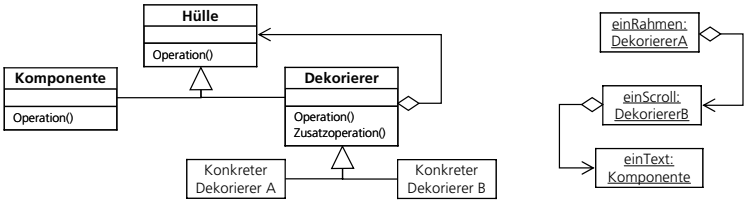


Abbildung 9-2: UML-Diagramm Decorator. Nach: [Gamma 96, S. 202]

- **Entwurfsziele** – DECORATOR sind flexibler als die Unterklassenbildung mit statischer Vererbung. Sie gestatten mehrfaches Dekorieren mit verschiedenen Elementen, zudem werden die Dekorationsfunktionen dynamisch zur Laufzeit festgelegt und auch wieder

entfernt. Andererseits erschweren DECORATOR die Verständlichkeit von Systemen. Die konkreten Objektstrukturen entstehen erst zur Laufzeit und sind aus den Quellcodes nur schwer ersichtlich.

- **Konsequenzen** – das Entwurfsmuster kann unmittelbar implementiert werden. Eine Anwendung weiterer Muster ist nicht erforderlich.

9.4 Beispiel Mustersequenz

Nachfolgend ist ein Beispiel einer Mustersprache aus [Henney06] wiedergegeben, das Mustersequenzen, ihre Auswirkungen auf die entstehenden Softwareartefakte und ihr Zusammenhang mit Musterbeziehungen zeigt.

Die Mustersprache dient der Verkapselung von Kontextobjekten und umfasst vier Entwurfsmuster. Kontextobjekte sind eine Möglichkeit, um lose gekoppelte Programmteile mit einem gemeinsamen Kontext zu versehen. Typische Beispiele hierfür sind Webseiten. Jede Seite ist ein eigenständiges Programm, ein verbindender Kontext zur Speicherung von Sitzungsdaten mit Benutzername, Benutzerrechten, Warenkorb etc. muss explizit aufgebaut und zwischen den Aufrufen der Webseiten zwischengespeichert werden.

Ein ECO (ENCAPSULATED CONTEXT OBJECT) kapselt den Kontext in einem Objekt, das Attribute enthält, die den Kontext repräsentieren, z. B. den oben genannten Benutzernamen. Das Kontextobjekt wird zwischengespeichert und den Kontextanwendern als Parameter übergeben. Eine virtuelle Interfaceklasse (DCI: DECOUPLED CONTEXT INTERFACE) entkoppelt die Kontextnutzer vom ECO. Eine Bindung von ECO und Kontextnutzer erfolgt erst zur Laufzeit. Das ECO kann jederzeit durch eine andere Implementierung ersetzt werden, zum Beispiel für Testzwecke durch Mockobjekte. In kontextreichen Anwendungen entwickeln ECOs Tendenzen zu überfrachteten Objekten. Auch benötigen die einzelnen Anwendungskomponenten selten den gesamten Kontext. Ein umfangreiches Objekt verschleiert die Verwendungsabhängigkeiten zwischen Kontext und Anwendungskomponenten. Mehr Klarheit bringt ein nach Rollen getrennter Kontext (RPC: ROLE-PARTITIONED CONTEXT). Jede Anwendungskomponente nutzt dann nur noch den tatsächlich benötigten Kontextanteil. Zur Implementierung von RPCs kann das ECO in rollenspezifische Kontextobjekte (RSCO: ROLE-SPECIFIC CONTEXT OBJEKT) zerlegt werden. Jedes RSCO realisiert nur den rollenspezifischen Kontextanteil. Falls ein DCI verwendet wird, ist dieses ebenfalls nach den Rollen aufzutrennen.

In Abbildung 9-3 ist das Diagramm der Musterbeziehungen mit Sequenzierungsbedingungen gezeigt. Die Darstellung ist an die Feature Modeling Notation angelehnt. Offene Kreise stehen für optionale Sequenzen, gefüllte Kreise für notwendige Sequenzen und Kreisbögen zeigen alternierende Sequenzen an, von denen mindestens eine Sequenz ausgeführt sein muss.

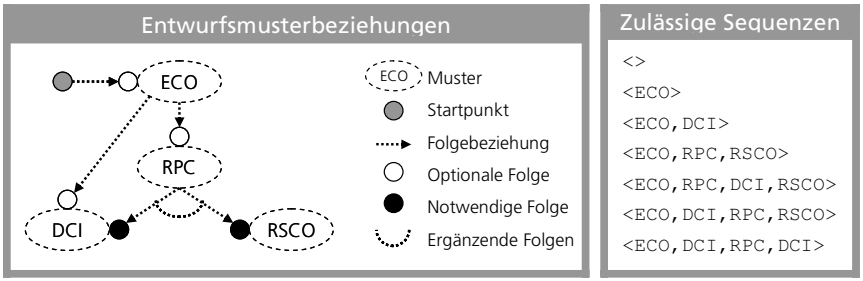


Abbildung 9-3: Vergleich von Musterbeziehungen und Mustersequenzen. Nach: [Henney06, S. D1-25]

Aus dem Diagramm lassen sich alle mit den Musterbeziehungen verträglichen Mustersequenzen ableiten. In der Abbildung aufgeführt ist die leere Sequenz, die einem nicht pas-

senden Musterkontext entspricht. Ebenfalls zulässig ist die einfachste Realisierung mit nur einem ECO. Umfangreichere Anwendungen benötigten mehrere Muster aus der Mustersprache.

Dabei können unterschiedliche Reihenfolgen der gleichen Muster zu differierenden Implementierungen führen [Henney06, S. D1-26]. Abbildung 9-4 zeigt das Resultat der Anwendung zwei verschiedener Sequenzen, die jeweils alle vier Muster, aber in unterschiedlichen Reihenfolgen einsetzen.

Die links gezeigte Teilsequenz <ECO, DCI> führt zunächst ein ECO und eine darauf abgestimmte virtuelle Interfaceklasse ein. Beide Klassen werden mittels der Anschlusssequenz <RPC, RSCO> weiter in rollenspezifische Unterklassen verfeinert, die ihre jeweiligen Teilfunktionalitäten von den Oberklassen erben. Bei der rechts gezeigten Sequenz wird das ECO unmittelbar in rollenspezifische Teile zerlegt. Für diese Teile werden virtuelle Interfaces DCI und Einzelobjekte RSCO implementiert. Diese Sequenz führt somit zu unabhängigen Kontextobjekten und Interfaceklassen ohne gemeinsame Oberklasse.

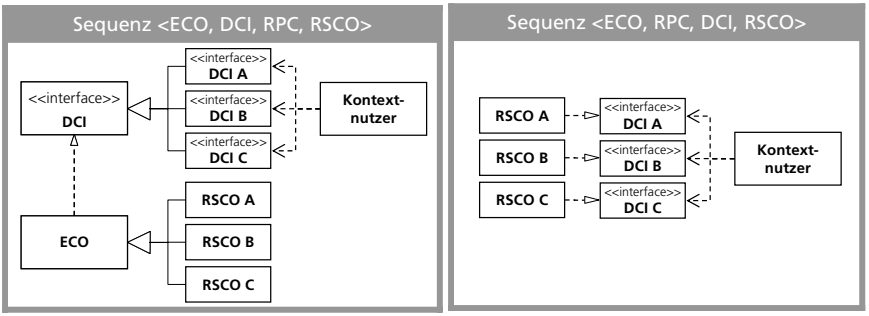


Abbildung 9-4: Beispielhafte Mustersequenzen und Implementierungsresultate