

Inhaltsverzeichnis

Abbildungsverzeichnis	XIII
Abkürzungsverzeichnis	XVII
1 Einleitung	1
1.1 Motivation	1
1.2 Zielsetzung und Positionierung	2
1.3 Aufbau	4
2 Wiederverwendung in der Softwareentwicklung	7
2.1 Wiederverwendung als genereller Problemlösungsansatz	7
2.2 Bedeutung der Wiederverwendung für die Softwareentwicklung	8
2.2.1 Begriffsverständnis und Historie	8
2.2.2 Motivation und Potentiale der Wiederverwendung	9
2.2.3 Probleme der Wiederverwendung	11
2.2.4 Dimensionen der Wiederverwendung	13
2.2.5 Wiederverwendung in der Praxis	15
2.3 Formen und Dokumente der Wiederverwendung in der Softwareentwicklung	17
2.3.1 Objektorientierte Diagramme	17
2.3.2 Entity-Relationship-Diagramme	21
2.3.3 Prozeßdiagramme	24
2.3.4 Entwurfsmuster	26
2.3.5 Funktions- und Klassenbibliotheken	28
2.3.6 Frameworks	29
2.3.7 Algorithmen	31
2.3.8 copy & paste	32
2.3.9 Architekturen	32
2.3.10 Komponenten und ihre Beziehungen zu anderen Dokumenten und Formen der Wiederverwendung	36
2.3.11 Abschließende Bemerkungen	39
3 Komponentenorientierte Softwareentwicklung aus softwaretechnischer Sicht	41
3.1 Definitorischer Rahmen	41
3.2 Historie	46
3.3 Komponentenmodelle und -infrastrukturen	52
3.3.1 Java Beans	52
3.3.2 CORBA	60

3.3.3	(D)COM, OLE und ActiveX	65
3.3.4	SAP Business Framework	70
3.3.5	Vergleichende Betrachtung	75
3.4	Auswirkungen auf die Softwareerstellung	78
3.4.1	Entwicklung von Komponenten	78
3.4.2	Entwicklung mit Komponenten	82
4	Komponentenorientierte Softwareentwicklung aus betriebswirtschaftlicher Sicht	85
4.1	Marktumfeld	85
4.2	Motivation und Potentiale für Softwareunternehmen	90
4.3	Derzeitiger Einsatz von KOS in Softwareunternehmen	96
4.4	Anforderungen an Softwareunternehmen	98
5	Vorgehensmodelle zur Einführung und Etablierung von Wiederverwendung und Komponentenorientierter Softwareentwicklung	105
5.1	Bewertungskriterien für Vorgehensmodelle	105
5.2	Modell von Lim	108
5.3	Modell von Karlsson	112
5.4	Modell von Jacobson, Griss, Jonsson	118
5.5	Weitere Modelle	123
5.6	Vergleichende Betrachtung	124
6	Vorgehensmodell zur Einführung und Etablierung von Komponentenorientierter Softwareentwicklung	127
6.1	Rahmenwerk	128
6.2	Analyse und Managementunterstützung	130
6.2.1	Analyse der KOS-Bedeutung	130
6.2.2	Managemententscheidung und -unterstützung	148
6.2.3	Initiierung und Finanzierung der Untersuchung	152
6.3	Planung	152
6.3.1	Softwaretechnische Aspekte	154
6.3.2	Prozesse	160
6.3.3	Organisation und Rollen	167
6.3.4	Personal	173
6.3.5	Marketing	177
6.3.6	Rechtliche und vertragliche Aspekte	177
6.3.7	Metriken	179
6.3.8	Einführungsstrategie	182
6.3.9	Managemententscheidung und Strategieformulierung	185

6.4	Realisierung	186
6.5	Zusammenfassung der KEF und des methodologischen Rahmenwerks . . .	188
6.6	Abgleich mit den Anforderungen	192
7	Abschließende Betrachtung	195
7.1	Zusammenfassung	195
7.2	Ausblick	197
	Literaturverzeichnis	199

Abbildungsverzeichnis

Abb. 1.1	Aufbau der Arbeit	4
Abb. 2.1	Hürden der Wiederverwendung	11
Abb. 2.2	Dimensionen der Wiederverwendung	13
Abb. 2.3	Bedeutende Formen bzw. Dokumente heutiger Wiederverwendung	18
Abb. 2.4	Klassendiagramm in UML	19
Abb. 2.5	Sequenzdiagramm in UML	20
Abb. 2.6	Zustandsdiagramm in UML	21
Abb. 2.7	Aktivitätsdiagramm in UML	22
Abb. 2.8	Entity-Relationship-Diagramm zur Verwaltung von Kundendaten	23
Abb. 2.9	Ereignisgesteuerte Prozesskette der Beschaffungslogistik	25
Abb. 2.10	MVC (<i>Model - View - Controller</i>)-Prinzip	27
Abb. 2.11	Publisher-Subscriber Design Pattern	28
Abb. 2.12	Architektur des San Francisco-Frameworks	31
Abb. 2.13	Informationssystem-Architektur von KRCMAR	33
Abb. 2.14	Informationssystem-Architektur CIM-OSA	34
Abb. 2.15	Informationssystem-Architektur von SCHEER	35
Abb. 2.16	Intention der Komponentenorientierten Softwareerstellung	37
Abb. 3.1	Beziehung von Komponenten zu den Dimensionen der Wiederverwendung	45
Abb. 3.2	Spektrum zwischen Individual- und Standardsoftware durch die KOS	47
Abb. 3.3	Pipes and Filter in der UNIX-Umgebung	48
Abb. 3.4	Object Management Architecture der OMG	50
Abb. 3.5	Traditionelle Übersetzung von Programmen	54
Abb. 3.6	Übersetzung von Java-Programmen	55
Abb. 3.7	Zugriffsmöglichkeiten von Java Beans hinsichtlich Anwendungen bzw. Server in Netzen	57
Abb. 3.8	Umgebung von Enterprise Java Beans	58
Abb. 3.9	CORBA-IIOP-Anbindung von Java Enterprise Beans	59
Abb. 3.10	Object Management Architecture der OMG	62
Abb. 3.11	CORBA Object Request Broker als Vermittlungsstelle bei einem Client- Server-Aufruf	63
Abb. 3.12	Interoperabilität von CORBA Object Request Broker mittels GIOP/IIOP	63
Abb. 3.13	COM-Komponente	67
Abb. 3.14	Zugriff auf eine (D)COM-Komponente	67

Abb. 3.15	Business Framework von R/3	72
Abb. 3.16	Aufbau eines Business Objects in R/3	73
Abb. 3.17	Beziehung zwischen Komponenten- und Anwendungsentwicklung	79
Abb. 4.1	Rolle von Komponenten in der betrieblichen Softwareerstellung	87
Abb. 4.2	Potentiale der Komponentenorientierten Softwareentwicklung	90
Abb. 4.3	Spektrum zwischen Individual- und Standardsoftware	94
Abb. 4.4	Derzeitige Probleme der Komponentenorientierten Softwareentwicklung	97
Abb. 4.5	Projektübergreifende Bedeutung der KOS	100
Abb. 4.6	Kritische Erfolgsfaktoren bei der Einführung und Etablierung der KOS	101
Abb. 5.1	Kriterien zur Bewertung von Vorgehensmodellen	107
Abb. 5.2	Adoptionsmodell von LIM	108
Abb. 5.3	Adoptionsmodell von KARLSSON	113
Abb. 5.4	Adoptionsmodell von JACOBSON, GRISS und JONSSON	118
Abb. 5.5	Zusammenfassende Bewertung der Vorgehensmodelle	125
Abb. 6.1	Vorgehensmodell zur Einführung und Etablierung Komponentenorientierter Softwareentwicklung	129
Abb. 6.2	Voraussetzungen für eine erfolgreiche Einführung der KOS	131
Abb. 6.3	Dimensionen bei der Beurteilung der Wiederverwendbarkeit von Komponenten (Homogenitätsmatrix HOMA)	132
Abb. 6.4	Beispielhafte Ergebnisse einer Clusteranalyse	134
Abb. 6.5	Wettbewerbsstrategisches Dreieck	136
Abb. 6.6	Kosten der KOS	139
Abb. 6.7	Nutzen der KOS	142
Abb. 6.8	Entscheidungsmatrix BEDEIG hinsichtlich der KOS-Einführung	149
Abb. 6.9	Facetten der KOS-Planung	153
Abb. 6.10	Aufgaben eines Komponentenrepositories	155
Abb. 6.11	Prozeßgruppen der KOS	161
Abb. 6.12	Temporale Typen der Komponentenerstellung	162
Abb. 6.13	Vorteile und Nachteile der temporalen Typen der Komponentenerstellung (Entscheidungsunterstützungsmatrix VNERTY)	165
Abb. 6.14	Projektorientiertes Organisationsmodell	168
Abb. 6.15	Zentralisiertes Organisationsmodell	168
Abb. 6.16	Domänenorientiertes Organisationsmodell	169
Abb. 6.17	Vorteile und Nachteile diverser Organisationsformen (Entscheidungsunterstützungsmatrix VNORFO)	170
Abb. 6.18	Mögliche Rollen bei der Komponentenorientierten Softwareerstellung	171

Abb. 6.19	Personalprobleme	173
Abb. 6.20	Framework für Wiederverwendungsmetriken	181
Abb. 6.21	Einführungsstrategien	183
Abb. 6.22	Vorteile und Nachteile diverser Einführungsstrategien (Entscheidungs- unterstützungsmatrix VNEINFSTR)	185
Abb. 6.23	KEF- und Methodologie-Rahmenwerk (1)	189
Abb. 6.24	KEF- und Methodologie-Rahmenwerk (2)	190
Abb. 6.25	KEF- und Methodologie-Rahmenwerk (3)	191
Abb. 7.1	Vorgehensmodell zur Einführung und Etablierung Komponentenorien- tierter Softwareentwicklung	197