

# Inhaltsverzeichnis

<b>Geleitwort</b> .....	<b>v</b>
<b>Vorwort</b> .....	<b>vi</b>
<b>Danksagung</b> .....	<b>viii</b>
<b>Inhaltsverzeichnis</b> .....	<b>ix</b>
<b>1 Erkenntnistheoretische Grundlagen und Begriffsbildung</b> .....	<b>1</b>
<b>1.1 Objekt – Attribut – Beziehung</b> .....	<b>1</b>
1.1.1 Wahlfreiheit bei der Strukturierung des Weltbildes ....	3
1.1.2 Identität und Wiedererkennen .....	5
1.1.3 Gleichheit .....	5
1.1.4 Kompositionen .....	5
<b>1.2 Klasse – Typ – Exemplar</b> .....	<b>7</b>
1.2.1 Einfache Klassifizierung .....	7
1.2.2 Hierarchische Klassifizierung .....	9
1.2.2.1 Baumartige Klassifizierung .....	9
1.2.2.2 Allgemeine hierarchische Klassifizierung .....	11
1.2.3 Abstraktion und Konkretisierung .....	15
<b>1.3 Typmehrdeutigkeit und Polymorphie</b> .....	<b>18</b>
1.3.1 Typmehrdeutigkeit .....	18
1.3.2 Polymorphie .....	19
1.3.3 Unterscheidung Typmehrdeutigkeit – Polymorphie ....	20
1.3.4 Migration .....	20
<b>1.4 Das programmierte System und seine Komponenten</b> .....	<b>21</b>
1.4.1 Das programmierte System .....	21
1.4.2 Komponenten programmierter Systeme .....	25
<b>2 Änderbarkeit dynamischer Systeme</b> .....	<b>30</b>
<b>2.1 Einführung in Software–Qualität und Änderbarkeit</b> .....	<b>31</b>
2.1.1 Qualität – grundlegende Begriffe und Definitionen ....	32
2.1.2 Erkenntnistheoretische Ansätze .....	36
2.1.3 Die Darstellung der Beziehungen zwischen Qualitäts- merkmalen in der Literatur .....	40

<b>2.2</b>	<b>Das Verhältnis zwischen Änderbarkeit und Entwicklungskonzept .....</b>	<b>51</b>
2.2.1	Bearbeitungsspezifische Einflüsse auf Änderungsaufwände .....	53
2.2.2	Aufgabenspezifische Einflüsse auf Änderungsaufwände .....	58
2.2.3	Die Problematik der Bestimmung konzeptioneller Einflüsse .....	61
<b>2.3</b>	<b>Die Problematik quantitativer Maße .....</b>	<b>64</b>
2.3.1	Die Problematik von Aufwandsmaßen .....	66
2.3.2	Die Problematik eines absoluten Änderbarkeitsmaßes ..	68
2.3.3	Die Problematik eines relativen Änderbarkeitsmaßes ..	71
<b>2.4</b>	<b>Konkretisierung der Begriffe Änderungsaufwand und Änderbarkeit .....</b>	<b>72</b>
2.4.1	Der Prozeß der Systementwicklung .....	72
2.4.2	Klassifizierung von Änderbarkeit bezüglich des Änderungsgrundes .....	81
2.4.3	Phasenorientierte Zerlegung von Änderungsaufwänden	86
2.4.4	Weitere Einflüsse auf Änderbarkeit .....	91
2.4.4.1	Änderbarkeit durch Entkopplung .....	92
2.4.4.2	Änderbarkeit durch Universalität .....	94
2.4.4.3	Änderbarkeit durch reduzierte Redundanz .....	97
<b>3</b>	<b>Konzepte der Objektorientierung .....</b>	<b>99</b>
<b>3.1</b>	<b>Erkenntnistheoretische Grundlagen der Objektorientierung .....</b>	<b>100</b>
<b>3.2</b>	<b>Wesenszüge objektorientierter Programmierung .....</b>	<b>101</b>
3.2.1	Das Basismodell objektorientierter Systeme .....	105
3.2.2	Struktur und Strukturvarianz .....	114
3.2.3	Klassifizierung von Objektakteuren .....	116
3.2.4	Abgrenzung zur Implementierungsvererbung .....	122
3.2.5	Identifikation von Akteuren und Polymorphie .....	124
<b>3.3</b>	<b>Optionale Merkmale objektorientierter Systeme .....</b>	<b>128</b>
3.3.1	Klassenakteure und Pseudo-Metaklassen .....	128
3.3.2	Parametrisierbare Klassenbeschreibungen .....	132
3.3.3	Zusammengesetzte Objektakteure .....	134
3.3.4	Zusammenfassung: Relationen zwischen Objekten, Klassen und Metaklassen .....	137

<b>4</b>	<b>Einfluß von Codierungskonzepten auf die Änderbarkeit programmierter Systeme ...</b>	<b>141</b>
4.1	<b>Abstraktion und Kapselung .....</b>	<b>142</b>
4.1.1	Abstraktion und Kapselung von Operatoren .....	142
4.1.1.1	Abstraktion von der Operationsmethode .....	142
4.1.1.2	Verbergen von Operationsmethoden und Operatoren .....	149
4.1.2	Abstraktion und Kapselung von Operanden .....	151
4.1.2.1	Kapselung temporärer Operanden .....	151
4.1.2.2	Nichttemporäre Operanden und Methodenparameter .....	152
4.1.2.3	Komposition und Typisierung komplexer Datenobjekte .....	158
4.1.2.4	Abstraktion und Kapselung nichttemporärer Operanden .....	165
4.1.3	Einheiten der Kapselung und höhere Abstraktionen ...	169
4.1.3.1	Objektakteure als Einheiten der Kapselung .....	171
4.1.3.2	Klassen als Einheiten der Kapselung .....	172
4.1.3.3	Klassenübergreifende Einheiten der Kapselung ...	174
4.1.3.4	Abstrakte Datentypen und abstrakte Datenobjekte in klassisch prozeduralen Programmiersprachen ..	177
4.2	<b>Hierarchische Klassifikation .....</b>	<b>179</b>
4.2.1	Klassenbeschreibungen .....	179
4.2.1.1	Erweiterte Klassenbeschreibungen .....	180
4.2.1.2	Inkrementelle Klassenbeschreibungen .....	181
4.2.1.3	Inkrementelle, erweiterte Klassenbeschreibungen .	182
4.2.2	Die Kopplung zwischen inkrementellen Klassenbeschreibungen .....	182
4.2.2.1	Leistungsnutzung durch Subklassen .....	184
4.2.2.2	Leistungsanforderung durch Oberklassen .....	188
4.2.2.3	Leistungsnutzung durch Oberklassen .....	193
4.3	<b>Dynamische Bindung .....</b>	<b>193</b>
4.3.1	Universalität durch indirekte Methodenidentifikation ..	194
4.3.2	Verständnisprobleme durch dynamische Bindung .....	198
4.4	<b>Granularität im Widerspruch .....</b>	<b>199</b>
4.4.1	Zerlegung der Systemfunktionalität .....	199
4.4.2	Verlust der Anschauung durch Abstraktion .....	203
4.4.3	Kooperation verschiedenartiger Objektakteure .....	205
<b>5</b>	<b>Ausblick .....</b>	<b>206</b>
	<b>Literaturverzeichnis .....</b>	<b>207</b>
	<b>Sachwortverzeichnis .....</b>	<b>217</b>