

# Software-Entwicklungswerkzeuge: Methodische Grundlagen .

Von *DT-*, rer. pol. Frank Schöenthaler  
PROM ATIS Informatik, Straubenhardt  
und Dipl.-Wi.-Ing. Tibor iNjemet ..  
PROMATIS Informatik, Straubenhardt

2., überarbeitete Auflage



B. G. Teubner Stuttgart 1992

# Inhalt

<b>1</b>	<b>Einleitung</b>	<b>13</b>
1.1	CASE	13
1.2	Vorgehensmodelle - Methoden.^ Tools:-	15
1.3	Grundlegende Entwicklungsstrategien • • • • - •	16
1.3.1	Abstraktion und Konkretisierung	16
1.3.2	Funktions- und Datenorientierung	19
1.4	Phasen eines Entwicklungsprojekts	20
1.4.1	Vorstudie"	21
1.4.2	Anforderung'sanalyse	22
1.4.3	System-und Software-Entwurf	23
1.4.4	Implementation und Komppnententest,-	24
1.4.5	Systemtest	27
1.4.6	Betrieb und Wartung	27
1.5	Überblick	28
<b>2</b>	<b>Universell einsetzbare Techniken</b>	<b>33</b>
2.1	Dekompositionsdiagramme ^	33
2.1.1	Graphische Repräsentation	33
2.1.2	Gebräuchliche Erweiterungen -,i ,',• 1 !-/•• ..	36
2.1.3	Praktische Anwendung . - 2, " "•	41
2.2	Entscheidungstäbellen und -bäume	43
2.2.1	Grundlagen der Entscheidungstäbellen-Technik	43
2.2.2	Erweiterungen	47
2.2.3	Entscheidungsbäume v	51
2.3	Endliche Automaten	53

<b>Methoden und Sprachen für die Strukturierte Analyse</b>	<b>58</b>
3.1 Structured Analysis	58
3.2 Datenflußdiagramme nach DeMarco	62
3.2.1 Graphische Elemente	62
3.2.2 Ebenenbildung	65
3.2.3 Detailspezifikation von Prozessen und Daten	69
3.3 Datenflußdiagramme nach Gane und Sarson	72
3.3.1 Graphische Elemente	72
3.3.2 Ebenenbildung	76
3.4 Objektflußdiagramme	79
3.4.1 Graphische Elemente	80
3.4.2 Ebenenbildung.	83
<b>Datenmodellierung</b>	<b>86</b>
4.1 Grundlagen der Datenmodellierung	86
4.1.1 Ebenenarchitektur eines Datenbanksystems	86
4.1.2 Prinzipien der Konzeptuellen Modellierung	90
4.1.3 Datenmodelle	91
4.2 Semantisch-hierarchisches Objektmodell	93
4.2.1 Abstraktionen	93
4.2.2 Vererbung	97
4.2.3 Identifikation von Objekten	100
4.3 Entity-Relationship-Modelle	102
4.3.1 Grundlegende Konzepte	102
4.3.2 Krähenfuß-Notation	109
4.3.3 Zusätzliche Konzepte	112
4.4 Methodische Aspekte	115
4.4.1 Die Datenmodellierung im Life-Cycle-Modell	115
4.4.2 Informationsbedarfsanalyse	118
4.4.3 Strategische Informationsplanung	132

<b>5</b>	<b>Spezifikation des Systemverhaltens</b>	<b>135</b>
5.1	Vorüberlegungen	135
5.2	Structured Analysis/Real-Time	137
5.2.1	Der Ansatz von Hatley und Pirbhäi	138
5.2.2	Der Ansatz von Ward und Mellor	153
5.3	Petri-Netze	158
5.3.1	Bedingungs/Ereignis-Netze	158
5.3.2	Stelle/Transitions-Netze	161
5.3.3	Prädikat/TransitidnstNetze	164
5.3.4	Spezifikation weiterer <sup>5</sup> Systemaspekte	169
5.3.5	Anmerkungen zum Einsatz, von Petri-Netzen	171
5.3.6	Analyse	171
<b>6</b>	<b>Entwurf von Benutzerschnittstellen</b>	<b>178</b>
6.1	Begriffe	179
6.1.1	Dialogklassen	180
6.1.2	Dialogunabhängigkeit	183
6.2	Werkzeugunterstützung	185
6.2.1	User-interface.Toolkits	185
6.2.2	User-interface.Development Systems	187
6.3	Sprachen	188
6.3.1	Kontextfreie Grammatiken	188
6.3.2	Zustandsdiagramme	190
6.3.3	Ereignisbasierte Darstellungen	191
6.3.4	Spezifikation durch Bildschirmwürfe	193
6.3.5	Wissensbasierte Repräsentation	195
6.4	Zustandsdiagramme nach Wassermari	196
<b>7</b>	<b>Funktionsorientierter Programmentwurf</b>	<b>200</b>
7.1	Composite/Structured Design	201
7.1.1	Modularisierung	201
7.1.2	Strukture Charts	202
7.1.3	Organisation modularer Systeme	208
7.1.4	Entwurf modularer Systeme	211
7.2	Pseudo-Code	214
7.3	Struktogramme	220

<b>Datenorientierter Programmmentwurf</b>	<b>226</b>
8.1 Jackson-strukturierte Programmierung	226
8.1.1 Entwurfsidee . . . .	226
8.1.2 Jackson-Diagramme	229
8.1.3 Entwurfsschritte	232
8.1.4 Strukturkonflikte ' ' *	238
8.2 Programmmentwurf nach Warnier und Orr,	242
8.2.1 Entwurfsidee - /; • • • „>	242
8.2.2 Warnier/OrrTDiagramme v ... ;	243
8.2.3 Entwurfsschritte,- , - • . ,	246
8.3 Überlegungen zum datenorientierten Programmmentwurf	252
<b>Objektorientierter Entwurf</b>	<b>255</b>
9.1 Vorüberlegungen und Begriffe	255
9.1.1 Objekt ' ' "	256
9.1.2 Vererbung	259
9.2 Programmmentwurf mit Eiffel ..	262
9.2.1 Objektorientierte. Systeme'	263
9.2.2 Bestimmung der Objektklassen	263
9.2.3 Beschreibung der Objektklassen ^	264
9.2.4 Klassenbeziehungen: Wiederverwendung und Erweiterung von Eigenschaften	269
9.2.5 Deferred Classes •	271,,
9.2.6 Ausführbarer Programmmentwurf	272
9.3 Objektorientierter strukturierter Entwurf , .	275
9.3.1 Klassendefinition	275
9.3.2 Behandlung von Ausnahmesituationen *	277
9.3.3 Instanziierung von Objekten	277
9.3.4 Vererbung •	278
9.3.5 Generische Einheiten	280
9.3.6 Weitere Notationen	281
9.3.7 Entwurf objektorientierter Systeme	282

<b>10</b>	<b>Alternative Ansätze zur Software-Entwicklung</b>	<b>283</b>
10.1	Motivation	283
10.2	Software-Wiederverwendung	284
10.2.1	Arten der Wiederverwendung	284
10.2.2	Techniken	286
10.2.3	Wiederverwendung im Software-Life-Cycle	292
10.2.4	Reverse Engineering	295
10.3	Prototyping	297
10.3.1	Einführung	297
10.3.2	Prototyping-Zyklus	300
o 10.3.3	Prototyping-Ansätze	303
10.3.4	Techniken zur Unterstützung des Prototyping	308
10.4	Operationale Spezifikationssprachen und Transformationssysteme	314
10.4.1	Problematik konventioneller Software-Entwicklung	314
10.4.2	Operationale Spezifikationssprachen	315
10.4.3	Spezifikationsinterpreter	319
10.4.4	Transformationssysteme	322
<b>11</b>	<b>Konzeption und Durchsetzung von CASE-Strategien</b>	<b>327</b>
11.1	Konzeption einer CASE-Strategie	327
11.2	Einführung und Einsatz	330
11.3	Resümee	331
	<b>Produkte</b>	<b>333</b>
	<b>Literatur</b>	<b>337</b>
	<b>Verzeichnis der Abbildungen</b>	<b>357</b>
	<b>Register</b>	<b>363</b>