

Software-Entwicklungswerkzeuge: Methodische Grundlagen .

Von *DT-*, rer. pol. Frank Schönthaler
PROM ATIS'Informatik, Straubenhardt
und Dipl.-Wi.-Ing. Tibor iNjemeth ..
PROMATIS Informatik, Straubenhardt

2., überarbeitete Auflage



B. G. Teubner Stuttgart 1992

Inhalt

1	Einleitung				13
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	Anforderungsanalyse	•	•	•	22
1.4.3	System- und Software-Entwurf	•	•	•	23
1.4.4	Implementation und Komponententest,-	•	•	•	24
1.4.5	Systemtest	•	•	•	27
1.4.6	Betrieb und Wartung	•	•	•	27
1.5	Überblick	•	•	•	28
2	Universell einsetzbare Techniken				33
2.1	Dekompositionsdigramme	•	•	•	33
2.1.1	Graphische Repräsentation	•	•	•	33
2.1.2	Gebräuchliche Erweiterungen	•	•	•	36
2.1.3	Praktische Anwendung	•	•	•	41
2.2	Entscheidungstabellen und -bäume	•	•	•	43
2.2.1	Grundlagen der Entscheidungstabellen-Technik	•	•	•	43
2.2.2	Erweiterungen	•	•	•	47
2.2.3	Entscheidungsbäume	•	•	•	51
2.3	Endliche Automaten	•	•	•	53

Inhalt

Methoden und Sprachen für die Strukturierte Analyse	58
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 \ddot{o}	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
Datenmodellierung	86
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

		Inhalt	.9
5	• Spezifikation des Systemverhaltens	/ ...,-•	135
5.1	Vorüberlegungen	• -,"urL ,	135 "i
5.2	Structured Analysis/Real-Time	fy\''	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	•	^
5.3.1	Bedingungs/Ereignis-Netze	•* J, ••*•	• 158
5.3.2	Stelle/Transitions-Netze	- - •	161
5.3.3	Prädikat/TränsitidnstNetze"	* ' '	164
5.3.4	Spezifikation weiterer ⁵ Systemaspekte	" "	169
5.3.5	Anmerkungen zum Einsatz von Petri-Netzen	- -	171
5.3.6	Analyse	. . .	^
6	EntAvurf von Benutzerschnittstellen	178	
6.1	Begriffe	""	179
6.1.1	Dialogklassen	', "?	180
6.1.2	Dialogunabhängigkeit	'	183
6.2	Werkzeugunterstützung	, . . . • ' "jV.	185
6.2.1	User-interface Toolkits	' "l	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 Bildschirmentwürfe	. -	193
6.3.5	Wissensbasierte Repräsentation	. . . • r	195
6.4	Zustandsdiagramme nach Wassermari	:	196
7	, Funktionsorientierter Programmentwurf...	... 200	
7.1	Composite/Structured Design	" " - - ' •" "	201
7.1.1	Modularisierung	• - " : • " • . ' •:	201
7.1.2	Structüre 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

Datenorientierter Programmumentwurf	226
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 Programmumentwurf 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 Programmumentwurf	252
Objektorientierter Entwurf	255
9.1 Vorüberlegungen und Begriffe	255
9.1.1 Objekt	256
9.1.2 Vererbung	259
9.2 Programmumentwurf 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 Programmumentwurf	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

	Inhalt	11
10 Alternative Ansätze zur Software-Entwicklung	283	
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 Spezifikationsinterpretierer	319	
10.4.4 Transformationssysteme	322	
11 Konzeption und Durchsetzung von CASE-Strategien	327	
11.1 Konzeption einer CASE-Strategie	327	
11.2 Einführung und Einsatz	330	
11.3 Resümee	331	
Produkte	333	
Literatur	337	
Verzeichnis der Abbildungen	357	
Register	363	