Inhaltsverzeichnis

1	Einfü	ihrung ir	n die Softwareevolution	1		
	1.1	Wartu	ng und Evolution – eine Begriffsbestimmung	. 1		
		1.1.1	Zum Ursprung des Begriffes »Maintenance«	. 2		
		1.1.2	Zum Unterschied zwischen Erhaltung und Entwicklung	. 3		
		1.1.3	Zum Unterschied zwischen Erhaltung und Evolution	. 4		
		1.1.4	Zum Unterschied zwischen Änderung und Erweiterung	. 8		
		1.1.5	Zum Unterschied zwischen Korrektur und Sanierung	. 8		
	1.2	Iterativ	ve und evolutionäre Softwareentwicklung	. 9		
	1.3	Softwa	areevolution und agile Softwareentwicklung	10		
	1.4	Wartu	ng und Evolution in einer serviceorientierten IT-Welt	13		
	1.5	Strukt	ur und Inhalt der folgenden Kapitel	16		
2	Wirtschaftlichkeit der Softwareevolution					
	2.1	Zur W	Verterhaltung von Softwarekapitalgütern	17		
	2.2	Software als verpacktes Wissen				
	2.3	Wertgetriebene Softwareevolution				
	2.4	Einflüsse auf die Evolutionskosten				
		2.4.1	Der Einfluss der Größe auf die Softwareerhaltung	26		
		2.4.2	Der Einfluss der Komplexität auf die Softwareerhaltung	28		
		2.4.3	Der Einfluss der Qualität auf die Softwareerhaltung	29		
	2.5	Schätz	ung der Evolutionskosten	30		
	2.6	Ermitt	lung vom Evolutionsnutzen	33		
		2.6.1	Zum Nutzen der korrektiven Aufträge	34		
		2.6.2	Zum Nutzen der adaptiven Aufträge	35		
		2.6.3	Zum Nutzen der perfektiven Aufträge	35		
		2.6.4	Zum Nutzen der enhansiven Aufträge	36		

xii Inhaltsverzeichnis

	2.7	Beispie	eispiel einer Kosten-Nutzen-Rechnung		
		2.7.1	Kalkulation des ROI für ein betriebliches		
			Informationssystem		
		2.7.2	Kalkulation des ROI für ein Testwerkzeug	38	
	2.8	Schlus	sfolgerungen aus der Wirtschaftlichkeitsbetrachtung	39	
3	Die G	esetze d	ler Softwareevolution	41	
	3.1	Die Pie	onierleistung von Lehman und Belady	41	
	3.2	Lehma	ns Kategorisierung der Softwaresystemtypen	42	
		3.2.1	S-Systeme	43	
		3.2.2	P-Systeme	44	
		3.2.3	E-Systeme	46	
	3.3	Die für	nf Gesetze der Evolution	48	
		3.3.1	Gesetz der fortdauernden Änderung	48	
		3.3.2	Gesetz der zunehmenden Komplexität	48	
		3.3.3	Gesetz der abnehmenden Qualität	49	
		3.3.4	Gesetz der sinkenden Produktivität	49	
		3.3.5	Gesetz des begrenzten Wachstums	49	
	3.4	Zur G	ültigkeit der Evolutionsgesetze	49	
	3.5	Konse	quenzen aus den Gesetzen der Softwareevolution	50	
		3.5.1	Releasegrößen müssen begrenzt werden	51	
		3.5.2	Die Dokumentation muss mit jedem Release		
			aktualisiert werden	52	
		3.5.3	Der Code muss in regelmäßigen Abständen		
		2 - 4	saniert werden	52	
		3.5.4	Die Qualität der Software soll laufend überwacht werden	52	
		3.5.5	werden		
	3.6		n Software doch noch stirbt		
	3.6				
		3.6.1	Veralterung der Implementierungstechnologie		
		3.6.2	Ungeeignete fachliche Lösung		
		3.6.3	Admangigken von Schlüsseipersonen	33	
4	Der E	volution	nsprozess	57	
	4.1	Der So	oftwareevolutionsprozess aus historischer Perspektive	58	
	4.2	Das Ro	eleasekonzept	63	
	4.3	System	nanalyse	67	

Inhaltsverzeichnis xiii

	4.4	Releas	eplanung	69
	4.5	Fortsc	hreibung des Systems	71
		4.5.1	Korrekturaufgaben	71
		4.5.2	Änderungsaufgaben	72
		4.5.3	Weiterentwicklungsaufgaben	72
		4.5.4	Integrationsaufgaben	72
		4.5.5	Sanierungsaufgaben	72
		4.5.6	Optimierungsaufgaben	73
	4.6	System	nregressionstest	73
		4.6.1	Regressionstestplanung	73
		4.6.2	Regressionstestspezifizierung	74
		4.6.3	Regressionstestaufbau	74
		4.6.4	Regressionstestausführung	74
		4.6.5	Regressionstestvalidierung	75
		4.6.6	Regressionstestevaluierung	75
	4.7	System	ndokumentation	76
	4.8	System	nauslieferung	76
5	Softv	varesyst	emanalyse	79
5	Softv 5.1	•	emanalyse stände der Systemanalyse	79
5		Gegen	•	
5	5.1	Gegen	stände der Systemanalyse	80
5	5.1	Gegen Metho	stände der Systemanalyse	80 81
5	5.1	Gegen Metho 5.2.1	stände der Systemanalyse	80 81 81
5	5.1	Gegen Metho 5.2.1 5.2.2	stände der Systemanalyse	80 81 81 83
5	5.1	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4	stände der Systemanalyse	80 81 81 83 85
5	5.1 5.2	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4	stände der Systemanalyse	80 81 81 83 85 87
5	5.1 5.2	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr	stände der Systemanalyse	80 81 81 83 85 87 91
5	5.1 5.2	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr 5.3.1	stände der Systemanalyse	80 81 83 85 87 91
5	5.1 5.2	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr 5.3.1 5.3.2	stände der Systemanalyse	80 81 81 83 85 87 91 91
5	5.1 5.2	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr 5.3.1 5.3.2 5.3.3 5.3.4	stände der Systemanalyse	80 81 81 83 85 87 91 91 92 94
6	5.15.25.3	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr 5.3.1 5.3.2 5.3.3 5.3.4 System	stände der Systemanalyse	80 81 81 83 85 87 91 92 94
	5.15.25.3	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr 5.3.1 5.3.2 5.3.3 5.3.4 System	stände der Systemanalyse	80 81 81 83 85 87 91 92 94 95 95
	5.1 5.2 5.3 5.4 Softv	Gegen Metho 5.2.1 5.2.2 5.2.3 5.2.4 Ergebr 5.3.1 5.3.2 5.3.3 5.3.4 System	stände der Systemanalyse	80 81 81 83 85 87 91 92 94 95 95

xiv Inhaltsverzeichnis

	6.2	Die Schätzung eines neuen Release	02
		6.2.1 Messung der gegenwärtigen Systemgröße	03
		6.2.2 Hochrechnung der Systempflegekosten 10	05
		6.2.3 Schätzung der Weiterentwicklungskosten	09
		6.2.4 Kalkulation der Gesamtkosten des nächsten Release 11	10
	6.3	Die Identifikation und Zuweisung der Evolutionsaufgaben 13	12
		6.3.1 Evolutionsaufgaben	13
		6.3.2 Zuweisung der Evolutionsaufgaben	14
		6.3.3 Terminierung der Evolutionsaufgaben	15
		6.3.4 Management by Contract	16
	6.4	Werkzeuge für die Releaseplanung	17
7	Fehle	erbehebungen 1°	19
	7.1	Die Fehlermeldung	19
	7.2	Die Fehleranalyse	20
	7.3	Fehlerursachenforschung	23
	7.4	Fehlerbeseitigung	25
	7.5	Fehlerkorrekturtest	26
	7.6	Fehlerkorrekturdokumentation	28
	7.7	Fehlerstatistik	29
8	Ände	erungen 1:	33
	8.1	Der Änderungsprozess	34
	8.2	Der Änderungsantrag	
	8.3	Änderungsanalyse	
	8.4	Lokalisierung der zu ändernden Stellen	
		8.4.1 Analyse der Textdokumente	
		8.4.2 Analyse der Entwurfsdokumente	
		8.4.3 Analyse des Codes	
		8.4.4 Analyse der Datenstrukturen 14	43
		8.4.5 Analyse der Testfälle	
	8.5	Änderungsfolge	43
	8.6	Änderungsabschottung	
	8.7	Änderungsdurchführung	46

Inhaltsverzeichnis xv

	8.8	Änderu	ıngsvalida	tion	147
		8.8.1	Aufbau c	ler Integrationstestumgebung	148
		8.8.2		ung der Integrationstestdaten	149
		8.8.3		ntierung des geänderten Codes	149
		8.8.4	Ausführu	ing des Integrationstests	149
		8.8.5		ing der Testergebnisse	149
	8.9	Änderu	ıngsdokun	nentation	150
9	Sanie	runa			153
_	9.1	_	ng als Vor	aussetzung der Sanierung	154
	9.2		-		156
	9.3			iren	157
	,.0	9.3.1		ngsstrategie	159
		9.3.2		neitsstrategie	159
	9.4			ahmen	160
	· · ·	9.4.1	U	er-Sprachen	160
		9.4.2		rale Sprachen	161
		∕.π.∠	9.4.2.1	Reformatierung des Codes	163
			9.4.2.2	Bereinigung des Codes	163
			9.4.2.3	Umbenennung nicht sprechender	103
). I. <u>2.</u> 3	Datennamen	164
			9.4.2.4	Beseitigung inkompatibler Datentypen	164
			9.4.2.5	Entfernung festverdrahteter Daten	164
			9.4.2.6	Auslagerung der IO-Schnittstellen	165
			9.4.2.7	Restrukturierung der Ablauflogik	165
			9.4.2.8	Verflachung der Ablaufstruktur	166
			9.4.2.9	Zerlegung des Codes in kleinere Bausteine	167
		9.4.3	Objektor	rientierte Sprachen	167
			9.4.3.1	Verflachung der Methodenlogik	169
			9.4.3.2	Verflachung der Klassenhierarchie	170
			9.4.3.3	Eliminierung redundanter Methoden	170
			9.4.3.4	Ablösung komplexer Auswahlstrukturen	170
			9.4.3.5	Verlagerung gemeinsamer Methoden und	4 = 0
			0.426	Attribute	170
			9.4.3.6	Spaltung zu groß gewordener Klassen	171
			9.4.3.7	Vereinfachung der Schnittstellen	173
			9.4.3.8	Entfernung festverdrahteter Daten	173
			9.4.3.9	Entfernung des toten Codes	174
			9.4.3.10	Umbenennung der Bezeichner	174

xvi Inhaltsverzeichnis

			9.4.3.11	Ergänzung der Kommentierung	175
			9.4.3.12	Vereinheitlichung des Codeformats	175
		9.4.4	Skriptspra	achen	175
	9.5	Sanieru	ngsrevisio	n	176
10	Softw	areweite	rentwicklu	ıng	179
	10.1	Beauftra	agung eine	er Systemerweiterung	183
		10.1.1	Anforder	ungsgetriebene Evolution	183
		10.1.2	Modellge	triebene Evolution	184
		10.1.3	Change-R	Request-getriebene Evolution	185
	10.2	Analyse	eines Erv	veiterungsantrages	186
		10.2.1	Bestätigu	ng des angegebenen Nutzens	187
		10.2.2	Analyse d	ler Funktionsspezifikation	188
		10.2.3	Auswirku	ingsanalyse der beantragten Erweiterung	188
	10.3	Aufwan	ıdsschätzu	ing einer Erweiterung	189
		10.3.1		sschätzung in der anforderungsgetriebenen	
					190
		10.3.2		sschätzung in der modellgetriebenen	101
		10 2 2		CDindex CDindex	
	10.4			sschätzung in der CR-getriebenen Evolution	
				er Erweiterung	
				rweiterungsprojekts	
				Erweiterung	
				r Erweiterung	
				ung	
	10.9	Dokum	entation c	ler Erweiterung	197
11	Syste	mregress	ionstest		199
	11.1	Die Pro	blematik (der Testfallselektion	199
	11.2	Notwer	ıdigkeit ei	nes unabhängigen Testteams	201
	11.3	Regress	ionstestwo	erkzeuge	203
		11.3.1	Werkzeug	ge für die statische Analyse	204
			11.3.1.1	Werkzeuge zur Prüfung der Anforderungs-	
				spezifikation	204
			11.3.1.2	Werkzeuge zur Prüfung des Codes und des	205
			11 2 1 2	Entwurfsmodells	205
			11.3.1.3	Werkzeuge zur Prüfung der Testdokumentation	206
				1 Cotta OKulli Cilitati Oli	200

Inhaltsverzeichnis xvii

	11.3.2	Werkzeuge für die Verwaltung und Fortschreibung	
		der Testfälle	207
	11.3.3	Werkzeuge für die Testfallauswahl	208
	11.3.4	Werkzeuge für die Testdatengenerierung	209
	11.3.5	Werkzeuge für die Regressionstestausführung	210
	11.3.6	Werkzeuge für die Verfolgung der Testabläufe	211
	11.3.7	Werkzeuge für die Vermessung der Testüberdeckung	212
	11.3.8	Werkzeuge für den Abgleich der Testergebnisse	213
	11.3.9	Werkzeuge für die Fehlermeldung und	
		Fehlerverfolgung	214
11.4	Regress	sionstestverfahren	216
	11.4.1	Regressionstestplanung	216
	11.4.2	Regressionstestvorbereitung	218
	11.4.3	Regressionstestausführung	218
	11.4.4	Regressionstestauswertung	219
11.5	Regress	sionstestergebnisse	220
12 Fortla	aufende	Dokumentation	223
12.1	Die Re	chtfertigung der Systemdokumentation	223
		atisierte Nachdokumentation	225
	12.2.1	Analyse der Anforderungsdokumente	227
		Analyse des Entwurfsmodells	227
		Analyse des Codes	228
		12.2.3.1 Prozedurale Sprachen	229
		12.2.3.2 Objektorientierte Sprachen	230
		12.2.3.3 Gemeinsame Strukturen aller Sprachen	230
12.3	Multip	le Sichten auf ein Softwaresystem	232
	12 2 1	0 1 1 0 1 (1 1 1	222
	12.3.1	Statische Sichten auf die Programmlogik	232
			232
		Visualisierungstechniken	
		Visualisierungstechniken	234
		Visualisierungstechniken	234 235
		Visualisierungstechniken	234 235 235
		Visualisierungstechniken	234 235 235 236
	12.3.2	Visualisierungstechniken	234 235 235 236 237

xviii Inhaltsverzeichnis

	12.4	Aufbau	und Nutzung eines System-Repository	241
		12.4.1	Aufbau eines Repository	242
		12.4.2	Besichtigung der Repository-Inhalte	243
		12.4.3	Abfragen der Repository-Inhalte	245
		12.4.4	Generierung von Dokumenten	246
	12.5	Fortsch	reibung der Dokumentation	246
		12.5.1	Abgestimmte Evolution aller Softwareschichten	247
		12.5.2	Software = Code + Dokumentation + Test	248
Αı	nhan	g		
Α	Gloss	ar		253
В	Abkü	rzungen		259
c	Litera	iturverze	eichnis	261
	Index	(281