

Oliver Vogel / Ingo Arnold / Arif Chughtai /
Edmund Ihler / Uwe Mehlig / Thomas Neumann /
Markus Völter / Uwe Zdun

Software-Architektur

Grundlagen – Konzepte – Praxis



Spektrum
AKADEMISCHER VERLAG

| Inhaltsverzeichnis

1 Einleitung	1
1.1 Ausgangslage und Zielsetzung des Buches	2
1.2 Was ist Software-Architektur?	8
1.3 Leser-Leitfaden.....	11
1.3.1 Buchaufbau	11
1.3.2 Zielpublikum	15
1.3.3 Kapitelüberblick.....	15
1.3.4 Kapitel im Detail	18
2 Architektonischer Ordnungsrahmen	23
2.1 Motivation.....	24
2.2 Ordnungsrahmen im Überblick	26
2.3 Architekturen und Architektur-Disziplinen (WAS).....	30
2.4 Architektur-Perspektiven (WO)	31
2.5 Architektur-Anforderungen (WARUM)	32
2.6 Architektur-Mittel (WOMIT).....	33
2.7 Organisationen und Individuen (WER).....	36
2.8 Architektur-Vorgehen (WIE)	37
3 Architekturen und Architektur-Disziplinen (WAS)	39
3.1 Klassische Architektur als Ausgangspunkt.....	40
3.2 Von der klassischen Architektur zur Software-Architektur	44
3.3 Architektur und der Systemgedanke	52
3.4 Architektur und die Bausteine eines Systems	57
4 Architektur-Perspektiven (WO)	65
4.1 Architektur-Ebenen.....	66
4.1.1 Organisationsebene	71
4.1.2 Systemebene	73
4.1.3 Bausteinebene im Bereich Makro-Architektur	73
4.1.4 Bausteinebene im Bereich Mikro-Architektur.....	74
4.2 Architektur-Sichten.....	75
4.2.1 Zachman-Framework	84
4.2.2 Reference Model for Open Distributed Processing	87
4.2.3 4+1-Sichten-Modell.....	88

5 Architektur-Anforderungen (WARUM)	91
5.1 Allgemeines	92
5.2 Anforderungen im Überblick	94
5.3 Anforderungen im Detail	99
5.3.1 Organisationsanforderungen	99
5.3.2 Systemanforderungen	100
5.3.3 Bausteinanforderungen	101
5.3.4 Laufzeitanforderungen.....	102
5.3.5 Entwicklungszeitanforderungen	103
5.3.6 Organisatorische Rahmenbedingungen.....	105
5.4 Anforderungen im Architektur-Kontext.....	106
6 Architektur-Mittel (WOMIT).....	111
6.1 Architektur-Prinzipien	112
6.1.1 Prinzip der losen Kopplung.....	114
6.1.2 Prinzip der hohen Kohäsion.....	117
6.1.3 Prinzip des Entwurfs für Veränderung	119
6.1.4 Separation-of-Concerns-Prinzip.....	121
6.1.5 Information-Hiding-Prinzip.....	124
6.1.6 Abstraktionsprinzipien	127
6.1.7 Modularitätsprinzip.....	129
6.1.8 Rückverfolgbarkeitsprinzip	133
6.1.9 Selbstdokumentationsprinzip.....	134
6.1.10 Inkrementalitätsprinzip	134
6.1.11 Weitere Architektur-Prinzipien.....	136
6.2 Grundlegende architektonische Konzepte	136
6.2.1 Prozedurale Ansätze.....	136
6.2.2 Objektorientierung	139
6.2.3 Komponentenorientierung	146
6.2.4 Meta-Architekturen und Reflection	149
6.2.5 Generative Erzeugung von Systembausteinen	151
6.2.6 Modellgetriebene Software-Entwicklung.....	153
6.2.7 Aspektorientierung.....	161
6.2.8 Wartung von Software-Architekturen	165
6.3 Architektur-Stile	169
6.4 Architektur-Muster	173
6.4.1 Was ist ein Software-Muster?	173
6.4.2 Beispiele für Muster	177
6.4.3 Mustersprachen	182
6.5 Referenzarchitekturen.....	186
6.5.1 Definition und Bestandteile.....	186
6.5.2 Einsatz und Vorteile von Referenzarchitekturen	188
6.5.3 Anforderungen an Referenzarchitekturen	188

6.5.4 Arten von Referenzarchitekturen	189
6.5.5 Beispiel für eine Referenzarchitektur	190
6.6 Architektur-Dokumentationsmittel	195
6.6.1 Architektur-Metamodelle	196
6.6.2 Unified Modeling Language (UML).....	197
6.6.3 Architecture Definition Languages (ADL)	201
6.6.4 Domain Specific Languages (DSL)	206
6.7 Architektur-Strukturen	208
6.7.1 Zentralisierung gegenüber Dezentralisierung.....	209
6.7.2 n-Tier-Architektur.....	211
6.7.3 Rich Client gegenüber Thin Client.....	213
6.7.4 Middleware-Architekturen	215
6.7.5 Komponentenarchitekturen.....	218
6.7.6 Serviceorientierte Architekturen	221
6.7.7 Enterprise Service Bus.....	222
6.7.8 P2P-Systeme	223
6.8 Aktuelle Technologien in Software-Architekturen	223
6.8.1 Middleware-Systeme	224
6.8.2 Datenbanken und Persistenz von Geschäftsobjekten	229
6.8.3 Datenaustausch und Datentransformation mit XML.....	233
6.8.4 Dynamische Web-Seiten und Web-Application-Server	236
6.8.5 Komponentenplattformen	237
6.8.6 Web Services	240
7 Organisationen und Individuen (WER).....	243
7.1 Allgemeines	244
7.2 Organisationen.....	247
7.3 Individuen.....	253
7.4 Individuen und Gruppen.....	255
7.5 Architektur und Entscheidungen	259
7.6 Architekt als zentrale Rolle	264
8 Architektur-Vorgehen (WIE).....	269
8.1 Architektonisches Vorgehen	270
8.1.1 Entwicklungsprozess	270
8.1.2 Architektonische Tätigkeiten	273
8.1.3 Erstellen des Business Case.....	276
8.1.4 Verstehen der Anforderungen	278
8.1.5 Entwerfen der Architektur	279
8.1.6 Umsetzen der Architektur.....	284
8.1.7 Kommunizieren der Architektur.....	287
8.2 Anwendungsszenario: Product Line Engineering.....	293
8.2.1 Produktlinien und Software-Systemfamilien	294

8.2.2 Realisierungstechnologien und Zusammenhang zu MDSD	297
8.2.3 Erstellen des Business Case	299
8.2.4 Verstehen der Anforderungen (Domänenanalyse)	299
8.2.5 Entwerfen der Architektur (Domänenendesign)	302
8.2.6 Umsetzen der Architektur (Domänenimplementierung)	303
8.2.7 Rollen und Aufgaben	304
8.3 Anwendungsszenario: Enterprise Application Integration	305
8.3.1 Erstellen des Business Case	306
8.3.2 Verstehen der Anforderungen	309
8.3.3 Entwerfen der Architektur	317
8.3.4 Kommunizieren und Umsetzen der Architektur	327
8.4 Anwendungsszenario: Anwendungsdienst	328
8.4.1 Erstellen des Business Case für das System	352
8.4.2 Verstehen der Anforderungen	363
8.4.3 Entwerfen der Architektur	373
8.4.4 Kommunizieren und Umsetzen der Architektur	384
9 Risikofallmanagementsystem	389
9.1 Zusammenfassung	390
9.2 Architektur-Anforderungen (WARUM)	391
9.2.1 Business Case	391
9.2.2 Organisationsanforderungen	392
9.2.3 Systemanforderungen	397
9.2.4 Bausteinanforderungen	401
9.3 Architekturen und Architektur-Disziplinen (WAS)	407
9.3.1 Disziplinen	407
9.3.2 Entscheidungen zur Software-Architektur	409
9.4 Architektur-Perspektiven (WO)	411
9.4.1 Systemebene	411
9.4.2 Bausteinebene	412
9.5 Architektur-Mittel (WOMIT)	416
9.5.1 Architektur-Prinzipien	416
9.5.2 Grundlegende architektonische Konzepte	418
9.5.3 Generative und generische Verfahren	419
9.6 Organisationen und Individuen (WER)	423
9.6.1 Organisation	424
9.6.2 Individuen	426
9.7 Architektur-Vorgehen (WIE)	427

10 CRM-Kundendatenrepository	429
10.1 Zusammenfassung	430
10.2 Architektur-Anforderungen (WARUM)	431
10.3 Organisationen und Individuen (WER)	433
10.4 Architekturen und Architektur-Disziplinen (WAS)	435
10.5 Architektur-Perspektiven (WO)	435
10.6 Architektur-Mittel (WOMIT)	436
10.7 Architektur-Vorgehen (WIE)	437
10.8 Ausgangssituation	437
10.9 Anforderungen	439
10.10 Architektonische Entscheidungen	448
10.11 Logische Sicht	452
10.12 Umsetzung ausgewählter Anforderungen	458
10.13 Was hat die Architektur genutzt?	468
11 Eingebettete Komponenteninfrastrukturen	471
11.1 Zusammenfassung	472
11.1.1 Architektur-Anforderungen (WARUM)	473
11.1.2 Organisation und Individuen (WER)	475
11.1.3 Architekturen und Architektur-Disziplinen (WAS)	475
11.1.4 Architektur-Perspektiven (WO)	476
11.1.5 Architektur-Mittel (WOMIT)	476
11.1.6 Architektur-Vorgehen (WIE)	477
11.2 Product Line Engineering	479
11.2.1 Domain Scoping	479
11.2.2 Variabilitätsanalyse und Domänenstrukturierung	480
11.2.3 Kommunikationsparadigmen	481
11.2.4 Container-Services	484
11.2.5 Domänenendesign	485
11.2.6 Produktionsprozess	487
11.3 Modelle	489
11.3.1 Definition von Interfaces	490
11.3.2 Definition von Komponenten und Ports	492
11.3.3 Definition eines Systems	493
11.3.4 Gesamtmodell	497
11.3.5 Generator - Überblick	497
11.4 Implementierung von Komponenten	498
11.4.1 Abbildung auf Java	499
11.4.2 Parsen und Zusammenführen des Gesamtmodells	500
11.4.3 Pseudodeklarative Metamodellimplementierung	504
11.5 Codegenerierung	506
11.5.1 Referenzen	506
11.5.2 Umsetzung der Protokollzustandsautomaten	510

11.5.3 Generierung des Build Files	511
11.6 Fachliche Kaskadierung.....	512
11.6.1 Modellierungssprache	512
11.6.2 Generierung.....	514
11.7 Erfahrungen aus unserem Unternehmen	514
 Literaturverzeichnis.....	517
 Abkürzungsverzeichnis	537
 Index.....	541