

Inhaltsübersicht

1	Einleitung	19
1.1	Softwarearchitektur als Disziplin im Software Engineering	20
1.2	iSAQB® – International Software Architecture Qualification Board	22
1.3	Certified Professional for Software Architecture – Foundation und Advanced Level	23
1.4	Zielsetzung des Buches	25
1.5	Voraussetzungen	26
1.6	Leitfaden für den Leser	27
1.7	Zielpublikum	28
1.8	Danksagungen	28
2	Grundlagen von Softwarearchitekturen	29
2.1	Einbettung in den iSAQB®-Lehrplan	30
2.2	Softwareintensive Systeme und Softwarearchitekturen	31
2.3	Grundlegende Konzepte von Softwarearchitekturen	37
2.4	Der Softwarearchitekturentwurf aus der Vogelperspektive	55
2.5	Lernkontrolle	63
3	Anforderungen und Randbedingungen	67
3.1	Einbettung in den iSAQB®-Lehrplan	67
3.2	Arbeit mit Randbedingungen und äußeren Einflussfaktoren	68
3.3	Anforderungen an Qualitäten formulieren	70
3.4	Szenarien	77
3.5	Lernkontrolle	80
4	Entwurf von Softwarearchitekturen	83
4.1	Einbettung in den iSAQB®-Lehrplan	84
4.2	Überblick über das Vorgehen beim Architekturentwurf	85
4.3	Entwurfsprinzipien und Heuristiken	91

4.4	Architekturzentrierte Entwicklungsansätze	98
4.5	Techniken für einen guten Entwurf	108
4.6	Architekturmuster	115
4.7	EXKURS: Entwurfsmuster	127
4.8	Daten und Datenmodelle	136
4.9	Deployment und Betrieb	139
4.10	Lernkontrolle	143
5	Beschreibung und Kommunikation von Softwarearchitekturen	149
5.1	Einbettung in den iSAQB®-Lehrplan	149
5.2	Das CoCoME-Beispiel	150
5.3	Sichten und Schablonen	153
5.4	Technische oder querschnittliche Konzepte in Software- architekturen	180
5.5	Architektur und Implementierung	182
5.6	Übliche Dokumenttypen für Softwarearchitekturen	184
5.7	Praxisregeln zur Dokumentation	189
5.8	Beispiele weiterer Architektur-Frameworks	192
5.9	Lernkontrolle	194
6	Analyse und Bewertung von Softwarearchitekturen	197
6.1	Einbettung in den iSAQB®-Lehrplan	198
6.2	Bewertung von Softwarearchitekturen	198
6.3	EXKURS: Technischer Durchstich und Prototyp	202
6.4	Architekturanalyse	204
6.5	Lernkontrolle	209
7	EXKURS: Werkzeuge für Softwarearchitekten	211
7.1	Allgemeine Hinweise zu Werkzeugen	211
7.2	Werkzeuge zum Anforderungsmanagement	212
7.3	Werkzeuge zur Modellierung	214
7.4	Werkzeuge zur statischen Codeanalyse	215
7.5	Werkzeuge zur dynamischen Analyse	216
7.6	Werkzeuge zum Konfigurations- und Versionsmanagement	217
7.7	Werkzeuge zum Codemanagement	218
7.8	Werkzeuge zum Testen	219
7.9	Werkzeuge zur Dokumentation	220

Anhang	223
A Beispielfragen	225
A.1 Auszüge aus der Prüfungsordnung	225
A.2 Beispielfragen	227
B Abkürzungsverzeichnis	231
C Glossar	233
D Literaturverzeichnis	247
Index	253

Inhaltsverzeichnis

1	Einleitung	19
1.1	Softwarearchitektur als Disziplin im Software Engineering	20
1.2	iSAQB® – International Software Architecture Qualification Board	22
1.3	Certified Professional for Software Architecture – Foundation und Advanced Level	23
1.4	Zielsetzung des Buches	25
1.5	Voraussetzungen	26
1.6	Leitfaden für den Leser	27
1.7	Zielpublikum	28
1.8	Danksagungen	28
2	Grundlagen von Softwarearchitekturen	29
2.1	Einbettung in den iSAQB®-Lehrplan	30
2.1.1	Lernziele	30
2.2	Softwareintensive Systeme und Softwarearchitekturen	31
2.2.1	Was ist ein softwareintensives System?	31
2.2.2	EXKURS: Ausprägungen von softwareintensiven Systemen	32
2.2.3	Bedeutung der Softwarearchitektur für ein softwareintensives System	36
2.3	Grundlegende Konzepte von Softwarearchitekturen	37
2.3.1	Was ist eine Softwarearchitektur?	38
2.3.2	Bausteine, Schnittstellen und Konfigurationen	39
2.3.3	Konzepte der Beschreibung von Softwarearchitekturen	46
2.3.4	Architekturbeschreibung und Architekturebenen	50
2.3.5	Wechselwirkungen zwischen Softwarearchitektur und Umgebung	52
2.3.6	Qualität und Nutzen der Softwarearchitektur	53

2.4	Der Softwarearchitekturentwurf aus der Vogelperspektive	55
2.4.1	Ziele und Aufgaben des Softwarearchitekturentwurfs	56
2.4.2	Der Softwarearchitekturentwurf im Überblick	57
2.4.3	Wechselspiel der Tätigkeiten und Abstraktionsstufen im Entwurf	60
2.4.4	Aufgaben des Softwarearchitekten und Bezug zu anderen Rollen	62
2.5	Lernkontrolle	63
3	Anforderungen und Randbedingungen	67
3.1	Einbettung in den iSAQB®-Lehrplan	67
3.1.1	Lernziele	67
3.2	Arbeit mit Randbedingungen und äußeren Einflussfaktoren	68
3.2.1	Anliegen der Stakeholder	68
3.2.2	Arten von Einflussfaktoren	68
3.3	Anforderungen an Qualitäten formulieren	70
3.3.1	Qualitätsmodelle und Qualitätsmerkmale	70
3.3.1.1	Qualitätsmodell DIN ISO/IEC 25010:2023	71
3.3.1.2	Qualitätsmerkmale	71
3.3.1.3	Auswirkungen bestimmter Qualitätsmerkmale	74
3.3.1.4	Taktiken und Praktiken	74
3.3.2	Messbarkeit von Qualitäten	76
3.4	Szenarien	77
3.4.1	Arten von Szenarien	77
3.4.2	Bestandteile von Szenarien	78
3.4.3	Beispiele für Szenarien	79
3.5	Lernkontrolle	80
4	Entwurf von Softwarearchitekturen	83
4.1	Einbettung in den iSAQB®-Lehrplan	84
4.1.1	Lernziele	84
4.2	Überblick über das Vorgehen beim Architekturentwurf	85
4.3	Entwurfsprinzipien und Heuristiken	91
4.3.1	Top-down und bottom-up	91
4.3.2	Hierarchische (De-)Komposition	92
4.3.2.1	Divide et impera	92
4.3.2.2	Prinzipien bei der Zerlegung	93
4.3.2.3	So-einfach-wie-möglich-Prinzip	93
4.3.2.4	Trennung von Verantwortlichkeiten	94

4.3.3	Konzeptionelle Integrität	95
4.3.4	Erwarte Fehler	95
4.3.4.1	Postel's Law	96
4.3.5	Schmale Schnittstellen und Information Hiding	96
4.3.5.1	Information Hiding	96
4.3.5.2	Verwendung von Schnittstellen	96
4.3.6	Regelmäßiges Refactoring und Redesign	97
4.4	Architekturzentrierte Entwicklungsansätze	98
4.4.1	EXKURS: Domain-Driven Design	98
4.4.1.1	Fachmodelle als Basis	98
4.4.1.2	Systematische Verwaltung der Domänen- objekte	99
4.4.1.3	Strukturierung der Fachdomäne	100
4.4.1.4	Arten von Domänen	100
4.4.1.5	Integration von Domänen	101
4.4.2	EXKURS: Globale Analyse	101
4.4.3	EXKURS: Evolutionäre Architektur	101
4.4.3.1	Prinzipien	102
4.4.3.2	Fitnessfunktionen	103
4.4.4	EXKURS: Modellgetriebene Architektur	104
4.4.5	Referenzarchitekturen	105
4.4.5.1	Generative Erzeugung von Systembausteinen	105
4.4.5.2	Aspektorientierung	106
4.4.5.3	Objektorientierung	107
4.4.5.4	Prozedurale Ansätze	108
4.5	Techniken für einen guten Entwurf	108
4.5.1	Ausgangssituation und Motivation: degeneriertes Design	108
4.5.2	Lose Kopplung	109
4.5.3	Hohe Kohäsion	110
4.5.4	Single-Responsibility-Prinzip	111
4.5.5	Offen-geschlossen-Prinzip	111
4.5.6	Umkehr der Abhängigkeiten	112
4.5.7	Abtrennung von Schnittstellen	113
4.5.8	Zyklische Abhängigkeiten auflösen	113
4.5.9	Liskov'sches Substitutionsprinzip	114
4.6	Architekturmuster	115
4.6.1	Adaptierbare Systeme	115
4.6.1.1	Dependency Injection	115

4.6.2	Interaktive Systeme	116
4.6.2.1	Model View Controller	116
4.6.2.2	Model View Presenter	117
4.6.2.3	Presentation Abstraction Control	118
4.6.3	Vom Chaos zur Struktur	119
4.6.3.1	Schichtenarchitektur	120
4.6.3.2	Pipes and Filters	121
4.6.3.3	Blackboard	122
4.6.4	Verteilte Systeme	122
4.6.4.1	Herausforderungen verteilter Systeme	122
4.6.4.2	Broker	124
4.6.4.3	EXKURS: Serviceorientierung	125
4.6.4.4	Modularisierung	126
4.6.4.5	Microservices	127
4.7	EXKURS: Entwurfsmuster	127
4.7.1	Adapter	128
4.7.2	Observer	128
4.7.3	Decorator	129
4.7.4	Proxy	130
4.7.5	Fassade	130
4.7.6	Brücke	131
4.7.7	State	132
4.7.8	Mediator	133
4.7.9	Fabrik	133
4.7.10	Interpreter	134
4.7.11	Plug-in	135
4.7.12	Kombinator	136
4.8	Daten und Datenmodelle	136
4.8.1	Datenmodelle	136
4.8.2	Arten von Daten	137
4.8.3	Auswirkungen von Daten auf Architektur- entscheidungen	138
4.9	Deployment und Betrieb	139
4.9.1	Deployment	140
4.9.2	Betrieb	141
4.9.3	EXKURS: DevOps	142
4.10	Lernkontrolle	143

5	Beschreibung und Kommunikation von Softwarearchitekturen	149
5.1	Einbettung in den iSAQB®-Lehrplan	149
5.1.1	Lernziele	150
5.2	Das CoCoME-Beispiel	150
5.2.1	Anwendungsfälle im CoCoME-System	151
5.2.2	Übersicht über den strukturellen Aufbau des CoCoME-Systems	152
5.3	Sichten und Schablonen	153
5.3.1	Bewährte Sichten nach iSAQB®	154
5.3.2	UML-Diagramme als Notationsmittel in Sichtenbeschreibungen	155
5.3.3	Sichtenbeschreibung – Grobaufbau und Einführungsbeispiel	158
5.3.3.1	Grobaufbau – schablonenartige Sichtenbeschreibung	158
5.3.3.2	Beispiel: Auszug aus einer Sichtenbeschreibung für eine Bausteinsicht	160
5.3.4	Kontextsicht oder Kontextabgrenzung	162
5.3.5	Bausteinsicht	166
5.3.6	Laufzeitsicht	169
5.3.7	Verteilungssicht bzw. Infrastruktursicht	173
5.3.8	Wechselwirkungen zwischen Architektursichten	176
5.3.9	Hierarchische Verfeinerung von Architektursichten	177
5.4	Technische oder querschnittliche Konzepte in Softwarearchitekturen	180
5.4.1	Technische bzw. querschnittliche Konzepte: Beispieldimensionen	180
5.4.2	Beispiel: Fehlerbehandlung	181
5.4.3	Beispiel: Sicherheit	182
5.5	Architektur und Implementierung	182
5.5.1	Beispiel: Implementierung	183
5.6	Übliche Dokumenttypen für Softwarearchitekturen	184
5.6.1	Zentrale Architekturbeschreibung	184
5.6.2	Architekturüberblick	185
5.6.3	Dokumentübersicht	186
5.6.4	Übersichtspräsentation	186
5.6.5	»Architekturtapete«	186

5.6.6	Handbuch zur Dokumentation	186
5.6.7	Architecture Decision Record	186
5.6.8	Technische Informationen	187
5.6.9	Dokumentation von externen Schnittstellen	188
5.6.10	Template	188
5.7	Praxisregeln zur Dokumentation	189
5.7.1	Regel 1: »Schreiben aus der Sicht der Leser«	189
5.7.2	Regel 2: »Unnötige Wiederholung vermeiden«	189
5.7.3	Regel 3: »Mehrdeutigkeit vermeiden«	189
5.7.4	Regel 4: »Standardisierte Organisationsstruktur bzw. Schablonen«	190
5.7.5	Regel 5: »Begründen Sie wesentliche Entscheidungen schriftlich«	190
5.7.6	Regel 6: »Überprüfung auf Gebrauchstauglichkeit«	190
5.7.7	Regel 7: »Übersichtliche Diagramme«	191
5.7.8	Regel 8: »Regelmäßige Aktualisierungen«	191
5.7.9	EXKURS: Regel 9: »Passen Sie die Änderbarkeit der Dokumentation an die Architektur an«	191
5.8	Beispiele weiterer Architektur-Frameworks	192
5.8.1	4+1-Framework	192
5.8.2	SAGA	193
5.9	Lernkontrolle	194
6	Analyse und Bewertung von Softwarearchitekturen	197
6.1	Einbettung in den iSAQB®-Lehrplan	198
6.1.1	Lernziele	198
6.2	Bewertung von Softwarearchitekturen	198
6.2.1	Qualitative Bewertung	199
6.2.2	Quantitative Bewertung	199
6.2.2.1	Überprüfung von Architekturregeln	200
6.2.2.2	Metriken	200
6.2.2.3	Zyklomatische Komplexität	201
6.2.2.4	Goodharts Gesetz	202
6.3	EXKURS: Technischer Durchstich und Prototyp	202
6.3.1	Technischer Durchstich	202
6.3.2	Prototyp	203
6.3.2.1	Einsatz von Softwareprototypen	203
6.3.2.2	Arten von Softwareprototypen	203

6.4	Architekturanalyse	204
6.4.1	EXKURS: ATAM-Methode	204
6.4.1.1	Vorgehen bei der Bewertung	205
6.5	Lernkontrolle	209
7	EXKURS: Werkzeuge für Softwarearchitekten	211
7.1	Allgemeine Hinweise zu Werkzeugen	211
7.1.1	Kosten von Werkzeugen	211
7.1.2	Lizenzen und Lizenzbedingungen	212
7.2	Werkzeuge zum Anforderungsmanagement	212
7.2.1	Anforderungen und Entscheidungskriterien	213
7.2.2	Herausforderungen von Werkzeugen für das Anforderungsmanagement	213
7.2.3	Beispielhafte Vertreter	213
7.3	Werkzeuge zur Modellierung	214
7.3.1	Anforderungen und Entscheidungskriterien	214
7.3.2	Herausforderungen von Werkzeugen für die Modellierung	215
7.3.3	Beispielhafte Vertreter	215
7.4	Werkzeuge zur statischen Codeanalyse	215
7.4.1	Anforderungen und Entscheidungskriterien	215
7.4.2	Herausforderungen von Werkzeugen zur statischen Codeanalyse	216
7.4.3	Beispielhafte Vertreter	216
7.5	Werkzeuge zur dynamischen Analyse	216
7.5.1	Anforderungen und Entscheidungskriterien	217
7.5.2	Herausforderungen von Werkzeugen zur dynamischen Analyse	217
7.5.3	Beispielhafte Vertreter	217
7.6	Werkzeuge zum Konfigurations- und Versionsmanagement	217
7.6.1	Anforderungen und Entscheidungskriterien	217
7.6.2	Herausforderungen von Werkzeugen zum Konfigurations- und Versionsmanagement	218
7.6.3	Beispielhafte Vertreter	218
7.7	Werkzeuge zum Codemanagement	218
7.7.1	Herausforderungen von Werkzeugen zum Codemanagement	219
7.7.2	Beispielhafte Vertreter	219

7.8	Werkzeuge zum Testen	219
7.8.1	Anforderungen und Entscheidungskriterien	220
7.8.2	Herausforderungen von Testwerkzeugen	220
7.8.3	Beispielhafte Vertreter	220
7.9	Werkzeuge zur Dokumentation	220
7.9.1	Anforderungen und Entscheidungskriterien	221
7.9.2	Herausforderungen von Dokumentationswerkzeugen ...	221
7.9.3	Beispielhafte Vertreter	221

Anhang	223
---------------	------------

A	Beispielfragen	225
A.1	Auszüge aus der Prüfungsordnung	225
A.2	Beispielfragen	227
B	Abkürzungsverzeichnis	231
C	Glossar	233
D	Literaturverzeichnis	247
	Index	253