

# Inhaltsverzeichnis

<b>Vorwort</b>	<b>9</b>
<b>1 Java EE im Überblick</b>	<b>11</b>
1.1 Aufgabenstellung	11
1.2 Architekturmodell	11
1.3 Anwendungsbestandteile und Formate	12
1.4 Profile	15
1.5 Plattformen	16
<b>2 CDI</b>	<b>17</b>
2.1 Was ist das?	17
2.2 Wozu braucht man das?	17
2.3 Bereitstellung und Injektion von Beans	20
2.4 Lifecycle Callbacks	25
2.5 Qualifier	26
2.6 Alternatives	28
2.7 Nutzung der Java-EE-Umgebung	30
2.8 Producer und Disposer	31
2.9 Kontexte und Scopes	35
2.10 Interceptors	40
2.11 Decorators	43
2.12 Stereotypes	45
2.13 Eventverarbeitung	47
2.14 Programmgesteuerter Zugriff auf CDI Beans	49
2.15 Integration von JPA, EJB und JSF	52
2.16 Plattformen und Ergänzungen	53
<b>3 Java Persistence</b>	<b>57</b>
3.1 Worum geht's?	57
3.1.1 Lösungsansätze	58
3.1.2 Anforderungen an O/R-Mapper	59
3.1.3 Entwicklung des Standards	60
3.1.4 Architektur von Anwendungen auf Basis von JPA	61
<b>Java EE 6</b>	<b>5</b>

3.2	Die Basics	62
3.2.1	Entity-Klassen	62
3.2.2	Konfiguration der Persistence Unit	64
3.2.3	CRUD	66
3.2.4	Detached Objects	68
3.2.5	Entity-Lebenszyklus	69
3.2.6	Mapping-Annotationen für einfache Objekte	70
3.2.7	Generierte IDs	76
3.2.8	Objektgleichheit	79
3.3	Objektrelationen	82
3.3.1	Unidirektionale n:1-Relationen	83
3.3.2	Unidirektionale 1:n-Relationen	86
3.3.3	Bidirektionale 1:n-Relationen	87
3.3.4	Uni- und bidirektionale 1:1-Relationen	91
3.3.5	Uni- und bidirektionale n:m-Relationen	93
3.3.6	Eager und Lazy Loading	94
3.3.7	Kaskadieren	96
3.3.8	Orphan Removal	98
3.3.9	Anordnung von Relationselementen	99
3.4	Queries	100
3.4.1	JPQL	100
3.4.2	Native Queries	112
3.4.3	Criteria Queries	114
3.5	Vererbungsbeziehungen	122
3.5.1	Mapping-Strategie „SINGLE_TABLE“	123
3.5.2	Mapping-Strategie „TABLE_PER_CLASS“	125
3.5.3	Mapping-Strategie „JOINED“	126
3.5.4	Non-Entity-Basisklassen	127
3.6	Dies und das	128
3.6.1	Secondary Tables	128
3.6.2	Zusammengesetzte IDs	129
3.6.3	Dependent IDs	130
3.6.4	Locking	133
3.6.5	Callback-Methoden und Listener	137
3.6.6	Bulk Update/Delete	139
3.7	Caching	140

3.8 Erweiterte Entity Manager	143
3.9 Java Persistence in SE-Anwendungen	149
3.9.1 Konfiguration der Persistence Unit im SE-Umfeld	149
3.9.2 Erzeugung eines Entity Managers in SE-Anwendungen	150
3.9.3 Transaktionssteuerung in Java-SE-Anwendungen	152
<b>4 Bean Validation</b>	<b>153</b>
4.1 Aufgabenstellung	153
4.2 Plattformen und benötigte Bibliotheken	154
4.3 Validation Constraints	155
4.4 Objektprüfung	161
4.5 Internationalisierung der Validierungsmeldungen	162
4.6 Validierungsgruppen	163
4.7 Integration in JPA und JSF	164
4.8 Bean Validation in SE-Umgebungen	166
<b>5 JavaServer Faces</b>	<b>167</b>
5.1 Einsatzzweck von JSF	167
5.2 Die Basis: Java-Webanwendungen	167
5.2.1 Grundlegender Aufbau	167
5.2.2 Servlets	168
5.2.3 JavaServer Pages	170
5.3 JSF im Überblick	171
5.3.1 Model View Controller	171
5.3.2 Facelets	173
5.3.3 Request-Verarbeitung	173
5.4 Konfiguration der Webanwendung	175
5.5 Benötigte Bibliotheken und Plattformen	177
5.6 Programmierung der Views	177
5.6.1 JSF Tag Libraries	178
5.7 Managed Beans	184
5.8 Unified Expression Language	186
5.9 Navigation	191
5.10 Scopes	193
5.11 Verarbeitung tabellarischer Daten	194
5.12 Internationalisierung	196

5.13 Ressourcenverwaltung	199
5.14 GET Support	201
5.15 Event-Verarbeitung	203
5.16 Konvertierung	206
5.17 Validierung	210
5.18 Immediate-Komponenten	218
5.19 Ajax	218
5.20 Templating mit Facelets	222
5.21 Eigene JSF-Komponenten	226
5.22 Komponentenbibliotheken	233
5.23 Security	234
<b>6 Enterprise JavaBeans</b>	<b>237</b>
6.1 Aufgabenstellung	237
6.2 Aufbau von Enterprise JavaBeans	237
6.3 EJB Deployment	240
6.4 Lokaler Zugriff auf Session Beans	241
6.5 Remote-Zugriff	243
6.6 Transaktionssteuerung	246
6.7 Asynchrone Methoden	249
6.8 Timer	250
6.9 Security	252
<b>7 Ein „Real World“-Projekt</b>	<b>255</b>
7.1 Aufgabenstellung	255
7.2 Anwendungsarchitektur	257
7.3 Persistenz	259
7.4 Views	270
7.5 Fachliche Injektion	274
<b>Stichwortverzeichnis</b>	<b>277</b>