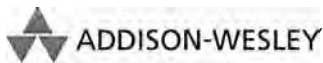


Adam Bien, Rainer Sawitzki

J2EE Hotspots

Professionelle Lösungen für die Java-Entwicklung



An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England
Don Mills, Ontario • Sydney • Mexico City
Madrid • Amsterdam

Inhalt

Vorwort 1	11
Vorwort 2	13
I Der Client	15
1.1 Installation einer EJB-Anwendung auf dem Client-Rechner	15
1.1.1 Problemstellung	15
1.1.2 Technischer Hintergrund	15
1.1.3 Lösung	16
1.1.4 Performance	17
1.2 Application Clients	18
1.2.1 Problemstellung	18
1.2.2 Technischer Hintergrund	18
1.2.3 Lösung	20
1.2.4 Praxis	21
1.2.5 Performance	25
1.3 Authentifizierung beim Applikationsserver mit dem JAAS	26
1.3.1 Problemstellung	26
1.3.2 Technischer Hintergrund	26
1.3.3 Praxis	30
1.4 Credentials und Anmeldung bei weiteren Ressourcen	33
1.4.1 Problemstellung	33
1.4.2 Technischer Hintergrund	34
1.4.3 Lösung	37
1.4.4 Praxis	41
1.5 Authentifizierung und Anpassung des JAAS	42
1.5.1 Problemstellung	42
1.5.2 Technischer Hintergrund	42
1.5.3 Praxis	47
1.6 Single Sign-On	49
1.6.1 Problemstellung	49
1.6.2 Technischer Hintergrund	49
1.6.3 Lösung	50
1.6.4 Praxis	50

1.7	Applets als J2EE-Komponenten	53
1.7.1	Problemstellung	53
1.7.2	Technischer Hintergrund	53
1.7.3	Lösung	54
1.7.4	Praxis	58
1.8	Deployment von J2EE-Clients mit Java Web Start	61
1.8.1	Problemstellung	61
1.8.2	Technischer Hintergrund	61
1.8.3	Lösung	63
1.8.4	Praxis	65
1.9	Ein Business Delegate pro Client	66
1.9.1	Problemstellung	66
1.9.2	Technischer Hintergrund	66
1.9.3	Lösung	68
1.9.4	Praxis	70
1.10	J2EE als Vorbild für ein Client-API	73
1.10.1	Problemstellung	73
1.10.2	Technischer Hintergrund	73
1.10.3	Lösung	75
1.10.4	Praxis	81
1.11	Visual Basic und Java	84
1.11.1	Problemstellung	84
1.11.2	Technischer Hintergrund	84
1.11.3	Lösung	86
1.11.4	Praxis	87
1.12	Microsoft Office und Java	94
1.12.1	Problemstellung	94
1.12.2	Technischer Hintergrund	94
1.12.3	Lösung	95
1.12.4	Praxis	95
1.13	Java und Microsoft Office	98
1.13.1	Problemstellung	98
1.13.2	Technischer Hintergrund	99
1.13.3	Lösung	99
1.13.4	Praxis	100
1.13.5	Performance	106
1.14	J2ME und die J2EE	107
1.14.1	Problemstellung	107
1.14.2	Technischer Hintergrund	107
1.14.3	Lösung	109
1.14.4	Praxis	111
2	Die Präsentationsschicht	119
2.1	Anbindung von EnterpriseBeans	119
2.1.1	Problemstellung	119
2.1.2	Technischer Hintergrund	119
2.1.3	Lösung	119
2.1.4	Performance	120

2.2	Ablage von Objektreferenzen	121
2.2.1	Problemstellung	121
2.2.2	Technischer Hintergrund	121
2.2.3	Lösung	122
2.2.4	Performance	126
2.3	Deaktivierte Cookies	127
2.3.1	Problemstellung	127
2.3.2	Technischer Hintergrund	127
2.3.3	Lösung	127
2.4	Die Performance der JavaServer Pages Standard Tag Libraries	128
2.4.1	Problemstellung	128
2.4.2	Technischer Hintergrund	128
2.4.3	Performance	129
2.5	Auswirkungen des SingleThreadModel-Interfaces	130
2.5.1	Problemstellung	130
2.5.2	Technischer Hintergrund	130
2.5.3	Performance	130
2.6	JavaServer Pages zur Konfiguration	131
2.6.1	Problemstellung	131
2.6.2	Technischer Hintergrund	131
2.6.3	Lösung	131
2.6.4	Praxis	131
2.7	JavaServer Pages zur Erzeugung beliebiger Formate	132
2.7.1	Problemstellung	132
2.7.2	Technischer Hintergrund	132
2.7.3	Praxis	133
2.8	Realms	136
2.8.1	Problemstellung	136
2.8.2	Technischer Hintergrund	136
3	Geschäftslogikschicht	141
3.1	Optimistische Locks mit EJB 2.0 und der Container Managed Persistence (CMP)?	141
3.1.1	Problemstellung	141
3.1.2	Technischer Hintergrund	143
3.1.3	Lösung	143
3.1.4	Praxis	144
3.2	Was »kosten« unnötige Transaktionen?	153
3.2.1	Problemstellung	153
3.2.2	Technischer Hintergrund	155
3.2.3	Lösung	157
3.2.4	Praxis	157
3.2.5	Performance	158
3.3	Wie kritisch sind die Stateful SessionBeans für die Performance wirklich?	162
3.3.1	Problemstellung	162
3.3.2	Technischer Hintergrund	163
3.3.3	Performance	165
3.3.4	Fazit	175

3.4	Benutzung von WebServices für den Zugriff auf eine Stateless SessionBean, statt eines konventionellen Zugriffs?	175
3.4.1	Problemstellung	175
3.4.2	Technischer Hintergrund	175
3.4.3	Lösung	176
3.4.4	Praxis	177
3.4.5	Performance	178
3.4.6	Fazit	182
3.5	Wie kann der Client mitteilen, dass die gerade benutzte EntityBean nicht mehr benötigt wird?	182
3.5.1	Problemstellung	182
3.5.2	Technischer Hintergrund	183
3.5.3	Lösung	184
3.6	Ab wann lohnt es sich, die EJB-Technologie in Projekten einzusetzen?	185
3.6.1	Problemstellung	186
3.6.2	Technischer Hintergrund	186
3.6.3	Antwort	187
3.7	Ist BMP schneller als die CMP 2.0 Persistenz?	188
3.7.1	Problemstellung	188
3.7.2	Praxis	202
3.7.3	Performance	204
3.7.4	Bewertung der Ergebnisse	212
3.7.5	Fazit	215
3.8	Wann wird eine EntityBean mit der Datenbank synchronisiert – oder was sind »commit-options«?	216
3.8.1	Problemstellung	216
3.8.2	Technischer Hintergrund	217
3.8.3	Performance	218
3.8.4	Fazit	223
3.9	Was passiert, wenn einer EntityBean die Daten aus der Datenbank zur Laufzeit geändert werden?	223
3.9.1	Problemstellung	224
3.9.2	Technischer Hintergrund	224
3.9.3	Lösung	225
3.9.4	Praxis	225
3.10	Was bedeutet <i>reentrant</i> bei den Deployment-Einstellungen der EntityBeans?	226
3.10.1	Problemstellung	227
3.10.2	Empfehlung	228
3.11	Was bringt die Benutzung von Local-Interfaces für die Performance der Anwendung?	229
3.11.1	Problemstellung	229
3.11.2	Performance	231
3.11.3	Lösung	238
3.11.4	Fazit	239
3.12	Wo soll der Zustand des Benutzers abgelegt werden (Präsentations- oder Geschäftslogikschicht)?	239
3.12.1	Problemstellung	240
3.12.2	Technischer Hintergrund	240

3.12.3	Lösung	242
3.12.4	Praxis	244
3.13	Was ist in einer J2EE-Umgebung nicht erlaubt?	245
3.13.1	Problemstellung	245
3.13.2	Restriktionen bei der Entwicklung von EJBs (EJB 2.1 Spezifikation)	245
3.13.3	Die Security-Einstellungen des Applikationsservers	251
3.14	Was muss bei der Implementierung eines Logging/Tracing-Mechanismus berücksichtigt werden?	251
3.14.1	Problemstellung	251
3.14.2	Technischer Hintergrund	252
3.14.3	Lösung	252
3.14.4	Praxis	254
3.14.5	Fazit	260
3.15	Warum gibt die ejbCreate Methode einer EntityBean ein Primary Key zurück?	261
3.15.1	Problemstellung	261
3.15.2	Technischer Hintergrund	262
3.15.3	Lösung	264
3.16	Was muss in der Design- und Implementierungsphase berücksichtigt werden?	267
3.16.1	Problemstellung	267
3.16.2	Lösung	269
3.16.3	Fazit	275
3.17	Berechtigungen auf dem Applikationsserver	275
3.17.1	Problemstellung	275
3.17.2	Technischer Hintergrund	275
3.17.3	Lösung	278
3.17.4	Praxis	280
3.18	Zugriff auf RMI-Server	282
3.18.1	Problemstellung	282
3.18.2	Technischer Hintergrund	282
3.18.3	Lösung	282
3.18.4	Praxis	282
4	Die Integrationsschicht	285
4.1	MQSeries als JMS-Provider	285
4.1.1	Problemstellung	285
4.1.2	Technischer Hintergrund	285
4.1.3	Lösung	285
4.2	Ansprechen nativer Methoden aus einer J2EE-Komponente	290
4.2.1	Problemstellung	290
4.2.2	Technischer Hintergrund	290
4.2.3	Lösung	290
4.2.4	Praxis	292