

Inhalt

Einleitung	13
------------------	----

1 Grundlagen der SAP-Schnittstellenprogrammierung 17

1.1 SAP NetWeaver Application Server	17
1.1.1 SAP-Lösungen und SAP NetWeaver	18
1.1.2 SAP NetWeaver Application Server ABAP	20
1.1.3 SAP NetWeaver Application Server Java	34
1.2 Sicherheit	39
1.2.1 Sicherheit in heterogenen Landschaften	39
1.2.2 User Management Engine	44
1.2.3 KeyStores: Authentifizierung, Signierung, Verschlüsselung	46
1.3 Authentifizierung und Autorisierung	50
1.3.1 Single Sign-on mit dem SAP NetWeaver Portal	51
1.3.2 Sichere Integration externer Systeme	56
1.4 Programmierung des SAP NetWeaver Application Servers ABAP	68
1.4.1 ABAP Dictionary	69
1.4.2 Authentifizierung und Autorisierung	74
1.4.3 Nummernkreise	75
1.4.4 Funktionsbausteine	76
1.4.5 Verbuchungsbausteine	78
1.4.6 Anwendungsfunktionalität und Benutzeroberflächen	80
1.5 Überblick über die Schnittstellen technologien von SAP	81
1.5.1 Dateischnittstelle	82
1.5.2 Remote Function Call (RFC)	84
1.5.3 Business Application Programming Interface (BAPI)	95
1.5.4 Application Link Enabling (ALE)	98
1.5.5 SOAP	104
1.5.6 OData	105

2 Remote Function Call mit ABAP 107

2.1	RFC-Funktionsbausteine in ABAP	107
2.1.1	Funktionsbausteine zum Lesen	107
2.1.2	Aufruf per sRFC	110
2.1.3	Funktionsbausteine zum Löschen und Ändern	113
2.2	Transaktionaler RFC	119
2.3	Queued RFC	125
2.3.1	qRFC mit Ausgangswarteschlange	127
2.3.2	qRFC mit Ausgangs- und Eingangswarteschlange	129
2.4	Background-RFC	133
2.5	Business-Objekte und BAPIs	137
2.5.1	Entwicklung von Business-Objekten	137
2.5.2	Entwicklung von BAPIs	139
2.5.3	Business-Objekt »Helpvalues«	161

3 Remote Function Call mit C/C++ 167

3.1	SAP NetWeaver RFC Software Development Kit	167
3.1.1	Inhalte des SAP NetWeaver RFC Software Development Kits	167
3.1.2	Kompilieren und Linken von C- und C++-Programmen	169
3.2	Einfache SAP-NetWeaver-RFC-Clients	174
3.2.1	Aufbau eines SAP-NetWeaver-RFC-Clients	175
3.2.2	Einfache Parameter	182
3.2.3	Strukturierte Parameter	184
3.2.4	Tabellenparameter	186
3.3	Komplexe SAP-NetWeaver-RFC-Clients	189
3.3.1	Aufruf von BAPIs	189
3.3.2	Transaktionaler RFC-Client	191
3.3.3	Queued RFC-Server	203
3.4	SAP-NetWeaver-RFC-Server	204
3.4.1	Einfacher RFC-Server	205
3.4.2	Implementierung der Funktionen	209
3.4.3	Transaktionaler RFC-Server	213

4 Remote Function Call mit Java 223

4.1	SAP Java Connector	223
4.1.1	Installation	224

4.1.2	Architektur des SAP Java Connectors	225
4.1.3	Programmierung mit dem SAP Java Connector	227
4.1.4	Verarbeitung von Tabellen und Strukturen	237
4.1.5	Transaktionaler RFC	240
4.1.6	Queued RFC	241
4.1.7	Metadatenverarbeitung	244
4.2	SAP Enterprise Connector	245
4.2.1	Erzeugen von Proxy-Klassen	246
4.2.2	Programmierung des Clients	251
4.3	Nutzung generischer Backend-Modelle	253
4.3.1	Generische Proxy-Klassen	253
4.3.2	Klassenabhängigkeiten	255
4.3.3	Konfiguration der Destinationen	257
4.3.4	Implementierung	260
4.4	RFC-Server	269
4.4.1	Serverseitiges und clientseitiges Repository	269
4.4.2	Programmierung eines einfachen JCo-Servers	272
4.4.3	Registrieren eines Funktionshandlers	274
4.4.4	Verwaltung von Transaktionen	276
4.5	SAP NetWeaver Portal Connector Framework	278
4.5.1	Java Connector Architecture	278
4.5.2	System Landscape des Portals	279
4.5.3	Einführung in die Portalprogrammierung	281
4.5.4	Anwendungsbeispiel des Connector Frameworks	285

5 Remote Function Call mit C# 293

5.1	SAP .NET Connector	293
5.1.1	Installation des SAP .NET Connectors	294
5.1.2	Verwendung des SAP .NET Connectors in Visual-Studio-Projekten	296
5.2	Einfache RFC-Clients	299
5.2.1	Aufbau eines einfachen RFC-Clients	299
5.2.2	Destination-Management	301
5.2.3	Repository und Metadaten	308
5.2.4	Einfache Parameter	310
5.2.5	Strukturierte Parameter	311
5.2.6	Tabellenparameter	314

5.2.7	Ausnahmebehandlung	316
5.3	Komplexe RFC-Clients	320
5.3.1	Aufruf von BAPIs und Session-Management	320
5.3.2	Transaktionaler RFC	323
5.3.3	Queued RFC	338
5.3.4	Background RFC	338
5.4	RFC-Server	341
5.4.1	Serverkonfiguration und Implementierung	342
5.4.2	Repository-Management	349
5.4.3	Sicherheitsmanagement	349
5.4.4	Session-Management	351
5.4.5	Transaktionaler RFC	353
5.4.6	Queued RFC	356
5.4.7	Background RFC	357
5.4.8	Zustand des Servers überwachen	358

6 IDocs und Application Link Enabling 361

6.1	IDocs	361
6.1.1	Entwicklung von IDocs	362
6.1.2	Erzeugung von IDocs	366
6.1.3	Eingangsverarbeitung von IDocs	370
6.2	Application Link Enabling	375
6.2.1	Konfiguration von ALE	375
6.2.2	Test und Monitoring	381
6.2.3	ALE-Schnittstelle für BAPIs	382
6.3	IDoc-Programmierung mit dem SAP NetWeaver RFC SDK	387
6.3.1	IDoc-Empfänger	387
6.3.2	IDoc-Sender	389
6.4	IDoc-Programmierung mit Java und dem Java Connector	391
6.4.1	Vorbereitung zur Verwendung der IDoc-Bibliotheken	391
6.4.2	Client-Anwendung für IDocs	392
6.4.3	IDoc-Server	397
6.4.4	Konfiguration für den Versand von IDocs	399
6.5	IDoc-Programmierung mit dem SAP .NET Connector	400
6.5.1	IDoc-Empfänger	401
6.5.2	IDoc-Sender	402

7 SOAP 405

7.1	Inside-Out-Webservices und -Webclients mit dem AS ABAP	405
7.1.1	ABAP-SOAP-Webservice	406
7.1.2	ABAP-SOAP-Webclient	415
7.2	Outside-In-Webservices und -Webclients mit dem AS ABAP	420
7.2.1	System Landscape Directory	420
7.2.2	Service-Interfaces	421
7.2.3	ABAP-SOAP-Proxys	425
7.3	Webservices und Webclients mit dem AS Java	429
7.3.1	Webservice-Infrastruktur im AS Java	430
7.3.2	Webservice-Provider mit J2EE	434
7.3.3	Webservice-Clients	441
7.3.4	Adaptiver Webservice mit Web Dynpro	446
7.3.5	Webservice-Programmierung mit SAP JEE	447
7.4	SOAP-Programmierung mit Java	468
7.4.1	Java API for XML Web Services	468
7.4.2	Implementierung eines Webservice-Clients	470
7.4.3	Implementierung eines Webservice-Providers	471
7.5	SOAP-Programmierung mit C#	472

8 SAP NetWeaver Gateway und OData 477

8.1	SAP NetWeaver Gateway	477
8.1.1	Embedded Deployment	478
8.1.2	Central Hub Deployment	479
8.2	OData-Protokoll	480
8.2.1	Atom-Syndication-Format	481
8.2.2	Representational State Transfer	481
8.2.3	OData-Datenmodell	482
8.3	OData-Services entwickeln	487
8.3.1	Beschreibung des Datenmodells	489
8.3.2	Generierung der Komponenten und Anlegen des OData-Services	492
8.3.3	Implementierung der Data-Provider-Klasse	496
8.3.4	Monitoring und Fehlersuche	509
8.4	OData-Services mit SAPUI5 konsumieren	511
8.4.1	Von Phoenix zu SAPUI5	512

8.4.2 Konsumieren eines OData-Services	515
8.5 OData-Services mit C# konsumieren	522
Anhang	531
A Literaturhinweise	531
B Die Autoren	533
Index	535