

Inhalt

Einleitung	17
------------------	----

TEIL I Grundlagen

1 Grundlagen der SAP-Schnittstellenprogrammierung	23
--	-----------

1.1 Die ABAP-Plattform	23
1.1.1 Anwendungsserver für SAP S/4HANA	23
1.1.2 SAP NetWeaver Application Server ABAP	25
1.2 Sicherheit	39
1.2.1 Sicherheit in heterogenen Landschaften	39
1.2.2 Authentifizierung und Autorisierung	43
1.2.3 Sichere Integration externer Systeme	45
1.3 Programmierung auf der ABAP-Plattform	51
1.3.1 ABAP Dictionary	52
1.3.2 Authentifizierung und Autorisierung	57
1.3.3 Nummernkreise	57
1.3.4 Funktionsbausteine	58
1.3.5 Verbuchungsbausteine	60
1.3.6 Anwendungsfunktionalität und Benutzeroberflächen	62
1.4 Überblick über die Schnittstellentechnologien von SAP	63
1.4.1 Dateischnittstelle	64
1.4.2 Remote Function Call (RFC)	66
1.4.3 Business Application Programming Interface (BAPI)	76
1.4.4 Application Link Enabling (ALE)	78
1.4.5 SOAP	84
1.4.6 OData	85
1.4.7 ABAP Channels	86

2

**Grundlagen der Schnittstellen-
programmierung auf der SAP Business
Technology Platform**

87

2.1	Einführung in die SAP Business Technology Platform	87
2.1.1	Planung für den Einsatz der SAP BTP	90
2.1.2	SAP-BTP-Bezahlmodell	93
2.2	Aufbau der Infrastruktur	97
2.2.1	Account-Konzept der SAP BTP	97
2.2.2	Verzeichnisse	100
2.2.3	Einführung in den Cloud Connector	104
2.3	Sicherheit	130
2.3.1	Benutzer	130
2.3.2	Application Roles und Role Collections	131
2.3.3	Identity Provider	134
2.3.4	Vertrauensbeziehung aufbauen	135
2.4	Entwicklungsinfrastruktur aufsetzen	141
2.4.1	SAP Business Application Studio als Entwicklungs- umgebung bereitstellen	141
2.4.2	Voraussetzung für die Verwendung des SAP Business Application Studios	142

TEIL II

**Schnittstellenprogrammierung auf
der ABAP-Plattform**

3

Remote Function Call mit ABAP

151

3.1	RFC-Funktionsbausteine in ABAP	151
3.1.1	Funktionsbausteine zum Lesen	151
3.1.2	Aufruf per sRFC	154
3.1.3	Funktionsbausteine zum Löschen und Ändern	157
3.2	Transaktionaler RFC	162
3.3	Queued RFC	169
3.3.1	qRFC mit Ausgangswarteschlange	171
3.3.2	qRFC mit Ausgangs- und Eingangswarteschlange	173

3.4	Background-RFC	176
3.4.1	Programmierung mit bgRFC	177
3.4.2	Konfiguration des bgRFC	180
3.5	Business-Objekte und BAPIs	184
3.5.1	Entwicklung von Business-Objekten	184
3.5.2	Entwicklung von BAPIs	186
3.5.3	Business-Objekt »Helpvalues«	208

4	Remote Function Call mit C/C++	213
----------	---------------------------------------	-----

4.1	SAP NetWeaver RFC SDK	213
4.1.1	Inhalte des SAP NetWeaver RFC SDK	213
4.1.2	Kompilieren und Linken von C- und C++-Programmen	215
4.2	Einfache SAP-NetWeaver-RFC-Clients	221
4.2.1	Aufbau eines SAP-NetWeaver-RFC-Clients	221
4.2.2	Einfache Parameter	229
4.2.3	Strukturierte Parameter	230
4.2.4	Tabellenparameter	232
4.3	Komplexe SAP-NetWeaver-RFC-Clients	235
4.3.1	Aufruf von BAPIs	235
4.3.2	Transaktionaler RFC-Client	238
4.3.3	Queued RFC-Client	249
4.4	SAP-NetWeaver-RFC-Server	250
4.4.1	Einfacher RFC-Server	251
4.4.2	Implementierung der Funktionen	255
4.4.3	Transaktionaler RFC-Server	259

5	Remote Function Call mit Java	269
----------	--------------------------------------	-----

5.1	SAP Java Connector	269
5.1.1	Installation	269
5.1.2	Architektur des SAP Java Connectors	270
5.1.3	Programmierung mit dem SAP Java Connector	272
5.1.4	Verarbeitung von Tabellen und Strukturen	282
5.1.5	Transaktionaler RFC	285

5.1.6	Queued RFC	286
5.1.7	Metadatenverarbeitung	289
5.2	RFC-Server	290
5.2.1	Serverseitiges und clientseitiges Repository	291
5.2.2	Programmierung eines einfachen JCo-Servers	293
5.2.3	Registrieren eines Funktions-Handlers	295
5.2.4	Verwaltung von Transaktionen	297
6	Remote Function Call mit C#	301
6.1	SAP Connector for Microsoft .NET	301
6.1.1	Installation des SAP Connectors für Microsoft .NET	302
6.1.2	Verwendung des SAP Connectors für Microsoft .NET in Visual-Studio-Projekten	302
6.2	Einfache RFC-Clients	304
6.2.1	Aufbau eines einfachen RFC-Clients	304
6.2.2	Destinationsmanagement	306
6.2.3	Repository und Metadaten	312
6.2.4	Einfache Parameter	314
6.2.5	Strukturierte Parameter	315
6.2.6	Tabellenparameter	318
6.2.7	Ausnahmebehandlung	320
6.3	Komplexe RFC-Clients	323
6.3.1	Aufruf von BAPIs und Session-Management	323
6.3.2	Transaktionaler RFC	327
6.3.3	Queued RFC	341
6.3.4	Background RFC	342
6.4	RFC-Server	345
6.4.1	Serverkonfiguration und Implementierung	345
6.4.2	Repository-Management	353
6.4.3	Sicherheitsmanagement	353
6.4.4	Session-Management	355
6.4.5	Transaktionaler RFC	357
6.4.6	Queued RFC	360
6.4.7	Background RFC	360
6.4.8	Zustand des Servers überwachen	361

7	IDocs und Application Link Enabling	363
7.1	IDocs	363
7.1.1	Entwicklung von IDocs	364
7.1.2	Erzeugung von IDocs	367
7.1.3	Eingangsverarbeitung von IDocs	371
7.2	Application Link Enabling	376
7.2.1	Konfiguration von ALE	376
7.2.2	Test und Monitoring	382
7.2.3	ALE-Schnittstelle für BAPIs	383
7.3	IDoc-Programmierung mit dem SAP NetWeaver RFC SDK	387
7.3.1	IDoc-Empfänger	388
7.3.2	IDoc-Sender	390
7.4	IDoc-Programmierung mit Java und dem SAP Java Connector	391
7.4.1	Vorbereitung zur Verwendung der IDoc-Bibliotheken	392
7.4.2	Client-Anwendung für IDocs	392
7.4.3	IDoc-Server	397
7.4.4	Konfiguration für den Versand von IDocs	400
7.5	IDoc-Programmierung mit dem SAP Connector for Microsoft .NET	401
7.5.1	IDoc-Empfänger	401
7.5.2	IDoc-Sender	402
8	SOAP-Webservices	405
8.1	Inside-Out-Webservices und -Webclients mit der ABAP-Plattform	405
8.1.1	SOAP-Webservice mit ABAP	406
8.1.2	SOAP-Webclient mit ABAP	415
8.2	Outside-In-Webservices und -Webclients mit der ABAP-Plattform	420
8.2.1	System Landscape Directory	420
8.2.2	Service-Interfaces	422
8.2.3	Das Backend Repository	425
8.2.4	SOAP-Proxys in ABAP	428

8.3	SOAP-Programmierung mit Java	432
8.3.1	Jakarta XML Web Services	432
8.3.2	Implementierung eines Webservice-Clients	435
8.3.3	Implementierung eines Webservice-Servers	439
8.4	SOAP-Programmierung mit C#	446

9	SAP Gateway und OData	451
----------	------------------------------------	------------

9.1	SAP Gateway	451
9.1.1	Embedded Deployment	452
9.1.2	Central Hub Deployment	453
9.2	OData-Protokoll	454
9.2.1	Representational State Transfer	454
9.2.2	OData-Datenmodell	456
9.3	OData-Services entwickeln	460
9.3.1	Beschreibung des Datenmodells	462
9.3.2	Generierung der Komponenten und Anlegen des OData-Service	465
9.3.3	Implementierung der Data-Provider-Klasse	468
9.3.4	Monitoring und Fehlersuche	482
9.4	OData-Services mit SAPUI5 konsumieren	484
9.4.1	Überblick über SAPUI5	484
9.4.2	Implementierung eines OData-Clients mit SAPUI5	487
9.5	OData-Services mit C# konsumieren	506
9.6	OData-Services mit Java konsumieren	514
9.6.1	Download und Installation von Apache Olingo	514
9.6.2	Implementierung eines OData-Clients mit Java	516

10	SAP HANA und ABAP Core Data Services	525
-----------	---	------------

10.1	Einführung in die Programmierung mit ABAP CDS	525
10.2	ABAP-CDS-Views implementieren	527
10.3	OData-Service registrieren und testen	538

11 ABAP RESTful Application Programming Model und OData 545

11.1 Einführung in das ABAP RESTful Application Programming Model 546

11.2 Design des virtuellen Datenmodells 548

11.3 Implementierung eines virtuellen Datenmodells 550

11.4 Business Services implementieren 556

 11.4.1 Implementierung einer Servicedefinition 557

 11.4.2 Implementierung eines Service-Bindings 558

 11.4.3 Implementierung von Verhalten 564

 11.4.4 Implementierung von Aktionen 568

11.5 OData-Services mit SAP Fiori Elements konsumieren 571

 11.5.1 SAP Fiori und SAP Fiori Elements im Überblick 571

 11.5.2 Implementierung eines List Reports mit Object Page 574

12 ABAP Channels 587

12.1 ABAP Push Channels 588

 12.1.1 Ein einfacher ABAP Push Channel 588

 12.1.2 Ein einfacher WebSocket-Client 591

12.2 ABAP Message Channels 593

 12.2.1 ABAP Message Channels als Repository-Objekte 594

 12.2.2 AMC-Sender 595

 12.2.3 AMC-Empfänger 596

12.3 Kollaborationsszenario mit ABAP Channels 599

 12.3.1 BOR-Ereignisse definieren 599

 12.3.2 AMC-Sender definieren 601

 12.3.3 APC-Handler-Klasse festlegen 603

 12.3.4 Szenario testen 604

12.4 ABAP-Plattform als Push-Channel-Client 605

 12.4.1 Ein einfacher ABAP-Push-Channel-Client 606

 12.4.2 ABAP-Plattform als Detached-Push-Channel-Client 609

TEIL III

SAP-Schnittstellenprogrammierung
in der Cloud

13

SAP Cloud Application Programming Model
und OData

615

13.1

Einführung in das SAP Cloud Application Programming
Model

615

13.1.1

Das SAP Cloud Application Programming Model
im Überblick

617

13.1.2

Dokumentation und Informationsquellen

619

13.2

Eine CAP-Anwendung mit OData-Service entwickeln

620

13.2.1

Entwicklung im SAP Business Application Studio

621

13.2.2

Implementierung eines OData-Service

630

13.2.3

Anlegen einer SAP-Fiori-Benutzeroberfläche

634

13.2.4

Schreibenden OData-Zugriff erlauben

642

13.3

Deployment einer CAP-Anwendung

643

13.3.1

SAP-BTP-Services aktivieren

644

13.3.2

Anpassungen im Projekt

646

13.3.3

Deployment ausführen

648

13.3.4

OData-Service testen

654

14

SAP Business Accelerator Hub

657

14.1

Einführung in den SAP Business Accelerator Hub

657

14.2

Mit dem SAP Business Accelerator Hub arbeiten

660

14.3

Services konsumieren

667

14.3.1

Anlegen eines CAP-Projekts

669

14.3.2

Generierung der Proxyklassen

671

14.3.3

Implementierung des Konsumenten

673

14.4

Nutzung des Service Centers im SAP Business Application
Studio

680

15	Echtzeitkommunikation mit Event-driven Architecture	693
15.1	SAP Event Mesh	694
15.2	Eine ereignisbasierte Architektur mit SAP Event Mesh einrichten	695
15.2.1	Event Mesh einrichten	696
15.2.2	Nachrichten an SAP Event Mesh senden	696
15.2.3	Nachrichten aus SAP Event Mesh empfangen	702
15.3	SAP Integration Suite	705
15.3.1	Open Connectors	707
15.3.2	Konnektor für einen SOAP-Webservice konfigurieren	708
15.4	Ausblick	715
	Literaturhinweise	717
	Die Autoren	719
	Index	721