

Inhalt

Einleitung 11

1 Von All-in-one-Software zu Webservices 17

- 1.1 Enterprise Application Integration 17
- 1.2 SAP-Technologie und -Komponenten 19
 - 1.2.1 SAP NetWeaver 20
 - 1.2.2 mySAP Business Suite 21
- 1.3 All-in-one versus Best-of-Breed 22

2 Middleware 27

- 2.0.1 Proxy-Dienste 27
- 2.0.2 Proxy-Dienste im Sicherheitsbereich 28
- 2.0.3 Message-Queues 29
- 2.0.4 Workflow 30
- 2.0.5 GUI-Server 30
- 2.0.6 3-Tier-Lösungen 31

3 Beispielszenarien und Voraussetzungen 35

- 3.1 Die nahe Zukunft 35
 - 3.1.1 XML-Datenbanken 35
 - 3.1.2 Entwicklung in mehreren Programmiersprachen 35
 - 3.1.3 Grid-Computing 36
- 3.2 Komponenten und Protokolle 37
- 3.3 Ein Beispiel einer heutigen Unternehmens-IT 38
 - 3.3.1 Modellsichten und Abstraktion 41
 - 3.3.2 Ansatz für eine zeitgemäße Lösung: Integration mit HTTP
und XML 44

4 SAP R/3 Remote Function Call 49

4.1	Was ist RFC?	49
4.2	R/3-RFC von einer anderen R/3-Instanz	51
4.2.1	Interne Destination NONE	52
4.3	Windows-zu-R/3-Connectivity mit DCOM	52
4.3.1	R/3-Logon mit VBA	54
4.3.2	Anatomie der RFC- und BAPI-Aufrufe von Windows	56
4.4	Helper-Klassen für den Zugriff auf R/3 via RFC	66
4.4.1	Class R3LogonObj	66
4.4.2	Class R3Credentials	67
4.4.3	Alternative R3Credentials-Class	68
4.5	R/3 Java Connectivity	70
4.5.1	SAP R/3 Java Connector JCo	70
4.5.2	SAP R/3-Logon mit Java	70
4.6	Das SAP RFC-Gateway	73
4.6.1	Registrierung eines RFC-Listeners beim SAP-Gateway	73
4.6.2	Anlegen einer RFC-Destination für einen registrierten Listener	74
4.6.3	Eigene RFC-Listener erstellen	74
4.7	Troubleshooting RFC	77
4.8	Aufruf von Remote Programs aus R/3 Via RFC	78
4.8.1	RFC via Remote Shell	78
4.8.2	RFC über Webserver	79
4.8.3	Aufruf eines Programms auf der Workstation mit RFC	80
4.8.4	Aufruf eines HTTP-Webserverns von SAP R/3	84
4.8.5	Proxy-Settings in ABAP	86
4.9	RFC von SAP R/3 auf Windows	87
4.9.1	Kommunikation mit Excel via OLE/2 und HTTP-Server	87
4.9.2	Zugriff auf ein Excel-Datenblatt von R/3	88
4.10	Business-Objekte und BAPI	95
4.10.1	Business-Objekte und Modellsichten	95
4.10.2	Technische Kriterien eines BAPI	99
4.10.3	Anwendungsbeispiel: Das Sales-Order-BAPI	100
4.10.4	Beispiel: BAPI_SALESORDER_GETLIST	101
4.10.5	VBA-Beispiel zu BAPI_SALESORDER_GETLIST	109
4.11	SAP Interface Repository	111

5 Frontend und GUIs 115

5.1	Casabac GUI-Server	116
5.1.1	Technologie	117
5.1.2	Beispiel zur Kalkulation der Mehrwertsteuer	119
5.2	Web Dynpro und Business Server Pages	123

6 Inhalt

6 Message-Queues und Message-Server 125

6.1	Anwendungen von Message Queues	125
6.1.1	Modelle von Message-Queues	125
6.1.2	Nutzen der Message-Queues	126
6.2	Message-Server Software	128
6.3	Kommunikation mit Messages-Queues	130

7 Internes Messaging in R/3 133

7.0.1	NAST Message Control	133
7.0.2	Entwicklung eigener NAST-Handler	134
7.0.3	Workflow-Events	138
7.0.4	Pro und Kontra Workflow versus NAST	138
7.1	Messaging mit R/3-IDocs	139
7.2	SAP R/3 Workflow programmieren	142
7.3	Workflow-Handler	146

8 Programmieren mit mehreren Programmiersprachen 147

8.1	Von Plattformen, Frameworks und Personalities	147
8.1.1	Plattformen	147
8.1.2	Frameworks und Personalities	149
8.2	Programmiersprachen und Entwicklungsumgebung	151
8.2.1	Die Entwicklungsumgebung	152
8.2.2	Die Compile-on-Demand-Runtime	152
8.2.3	Der Debugger	152
8.2.4	Einrichten einer Entwicklungsumgebung	153
8.2.5	Plattformscheidung: UNIX versus Windows NT	154
8.2.6	Framework: Web AS/ABAP, Microsoft.NET oder J2EE	154
8.2.7	Windows Scripting Host versus Visual Basic	156
8.2.8	Windows Classic versus Microsoft.NET	156
8.3	Die Wahl der richtigen Programmiersprache	157
8.3.1	Sprache für den Webserver	157
8.3.2	Java	158
8.3.3	J#	158
8.3.4	Visual Basic (VB) und Visual Basic for Applications (VBA)	158
8.3.5	Visual Basic Script (VBS) und ASP	158
8.3.6	C#	159
8.3.7	X#	159
8.3.8	Delphi	160
8.3.9	ADA	160
8.3.10	SmallTalk	161
8.3.11	LISP	161

8.4	Synopsis der Objektprogrammierung mit Java, VB und ABAP	162
8.4.1	Grundbegriffe	162
8.4.2	Constructor	169
8.4.3	Destructor	170
8.4.4	Property-Deklarationen	171
8.4.5	Methoden	172
8.4.6	String-Deklarationen	173
8.4.7	Array-Deklarationen	174

9 Webservice als Standard-Nachrichtenformat 179

9.1	Beispiel für Webservices mit Amazon.com	180
9.1.1	Einfache Beispiele für Webservices	180
9.1.2	Ein Beispiel mit einem winzigkleinen Fehler	181
9.1.3	Die WSDL zum Amazon.com-Service	185
9.1.4	Webservice als HTTP GET-Anfrage	188
9.1.5	Verwendung eines XSLT-Stylesheets	188
9.1.6	Type- und Namespace-Referenz	189

10 HTTP- und TCP/IP-Protokolle 191

10.1	TCP/IP und UDP/IP Network Protocol	191
10.2	Client-Server-Protokolle	193
10.2.1	Clientseite	193
10.2.2	Serverseite	194
10.2.3	URN, URI und URL – Resource Locator	194
10.2.4	CGI – Common Gateway Interface	196
10.3	HTTP-Kommunikation mit einem Webserver	196
10.3.1	Beispiel einer HTTP-Session	199
10.3.2	Beispiel für einen HTTP POST-Request	201
10.3.3	Anfragen direkt ins Internet	203

11 XML – Extended Markup Language 205

11.1	XML als Lingua franca des Internet	205
11.1.1	Kleine Anatomie eines XML-Dokuments	206
11.2	XSD-Schema und DTD	209
11.3	DOM – Document Object Model	213
11.3.1	DOM-Beispiele	214
11.4	XML-Parser	215
11.4.1	DOM oder SAX	216
11.4.2	Der Microsoft XML-Parser	216
11.4.3	XML-Parser Libraries for Java	217

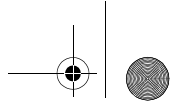
11.4.4	XML-Browser und Editoren	219
11.4.5	Namespaces – Namensräume	219
11.5	XSLT – eXtended Style Sheet Language Transformations	220
11.6	HTML-Seiten mit XSL-Stylesheets und XML	221
11.6.1	Inventurtabelle der Animal-Farm	221
11.6.2	XSL – Step by step	222
11.6.3	XSL-Templates	222
11.6.4	XSL auf die Animal-Farm angewandt	223
11.6.5	Anatomie des XSL-Stylesheets	224
11.6.6	Parsing des XML und Anwendung des XSL	227
11.7	SOAP	228
11.8	WSDL-Dokumente	231
11.8.1	WSDL-DOM	232
11.8.2	Beispiel eines WSDL-Dokuments für eine BAPI-Funktion	235

12 Entwicklung von dynamischen Webseiten 239

12.1	Server Pages und Scripting	239
12.1.1	Common Gateway Interface – CGI	240
12.1.2	Active Server Pages – ASP und ASP.NET	240
12.1.3	Java Server Pages – JSP	241
12.1.4	Java Servlets	243
12.1.5	Business Server Pages (BSP)	244
12.2	Web Pages mit Active Server Pages	245
12.2.1	ASP »Hello World«	245
12.2.2	Senden von HTML	246
12.2.3	Aufrufe beliebiger Windows-Objekte aus VBS	247
12.2.4	BAPI-Aufruf von ASP	247
12.2.5	Microsoft.NET Connector	250
12.2.6	Parameter aus der URI abfragen	250
12.2.7	ASP Application Variables	253
12.2.8	Global.asa – Die ASP-Autoexec-Datei	254
12.2.9	Extra-Beispiel: File-Liste mit ASP	255
12.3	Entwicklung von Business Server Pages für den SAP Web AS	256
12.3.1	»Hello World« für BSP	256
12.3.2	BAPI-Aufruf aus einer BSP	257
12.3.3	Online Text Repository	259

13 Webseiten von anderen Programmen abrufen 261

13.1	Abfragen einer URI mit Visual Basic	261
13.1.1	Das Windows-Webbrowser-Control	261
13.1.2	Anwendungsbeispiel für das Browser-Control	262
13.1.3	Das WINHTTP-Control	266



13.1.4	Anwendungsbeispiel für das WINHTTP-Control	267
13.2	Webentwicklung mit SAP R/3	271
13.2.1	SAP Function HTTP_GET	272
13.2.2	Proxy-Settings in ABAP	275
13.2.3	SAP und SOAP: Webservice Choreography Interface (WSCI) ...	276
13.3	SAP Internet Transaction Server	276

A Anhang 279

A.1	MIME-Typen	279
A.2	Beispiel für HTTP GET und HTTP POST	282
A.3	HTTP-Statuscodes	283
A.4	HTTP- und FTP-Kommandos	286
A.5	Java-Utilities	287
A.6	SAPLPD-Befehlszeilenparameter	290
A.7	Setzen des Suchpfads für SAPLOGON.INI	291
A.8	RFC-Connectors	292
A.9	Beispiele aus dem SAP Interface Repository	296

B Synopsis der Programmiersprachen ABAP, VB und Java 299

Index 309

