

Inhalt

Vorwort	11
1 Einleitung	13
1.1 An wen richtet sich dieses Buch?	13
1.2 Wie ist dieses Buch zu lesen?	13
1.3 Wie kann ich mit diesem Buch praktisch arbeiten?	16
1.3.1 Systemvoraussetzungen	16
1.3.2 Beispielprogramme	16
1.3.3 Übungsdaten	17
1.4 Syntaxkonventionen	19
2 Softwareentwicklung mit ABAP	21
2.1 Einleitung	21
2.2 Objektorientiertes oder klassisches ABAP?	22
2.2.1 Bessere Datenkapselung	23
2.2.2 Mehrfachinstanzierung	24
2.2.3 Bessere Wiederverwendbarkeit des Quelltextes	30
2.2.4 Separation von Schnittstelle und Implementierung	40
2.2.5 Unterstützung ereignisgesteuerter Programmierung	45
2.2.6 Keine »schwarze Magie« durch den Verzicht auf implizite Funktionalitäten	47
2.2.7 Verbesserte Syntax- und Semantikregeln	48
2.2.8 Einziger Weg zur Nutzung neuer ABAP-Technologien	49
2.2.9 ABAP Objects – just do it!	50
2.3 (Nicht) Wiederverwendbare Komponenten	53
2.3.1 Freibabestatus von Funktionsbausteinen und Klassen	54
2.3.2 Handlungsempfehlung	55
2.4 Entwicklungsrichtlinien in der ABAP-Welt	59
2.4.1 Namenskonventionen	60
2.4.2 Vermeidung von Literalen durch Verwendung von Konstanten	65
2.4.3 Dokumentation von funktionalen Einheiten	68
2.4.4 QS-Assessment	69
2.5 Robuste Programme entwickeln	72
2.5.1 Defensive Programmierung	72
2.5.2 Ausnahmebehandlung	73

2.5.3	Klassenbasierte Ausnahmebehandlung	81
2.5.4	Definition eigener Ausnahmen	85
2.5.5	Einbinden der traditionellen Ausnahmebehandlung in die klassenbasierte Ausnahmebehandlung	89
2.5.6	Historische Fehlerereignisse	92
3	Lösungskonzepte für die tägliche Programmierung mit ABAP	95
3.1	Dynamische und generische Programmierung	95
3.1.1	Dynamische Adressierung von Feldnamen	96
3.1.2	Dynamische Adressierung von Tabellennamen	98
3.1.3	Dynamische Angabe von Feldlisten und Klauseln bei SELECT	101
3.1.4	Dynamische Erzeugung eines Datenobjekts beliebigen Typs	105
3.1.5	Dynamische Adressierung von Strukturkomponenten	108
3.1.6	Sichere Sortierung einer internen Tabelle dynamisch durch den Benutzer erlauben	110
3.1.7	Dynamischer Aufruf von internen Unterroutinen	114
3.1.8	Dynamischer Aufruf von externen Unterroutinen mit Zugriff auf globale Felder des externen Programms	117
3.1.9	Dynamischer Aufruf von Funktionsbausteinen mit dynamischer Parameterübergabe	120
3.1.10	Dynamische Erzeugung von Klasseninstanzen und dynamischer Methodenaufruf mit dynamischer Parameterübergabe	126
3.1.11	Polymorphie über Interfaces	134
3.2	Generierung von Programmen	140
3.2.1	Erzeugung transienter Programme	141
3.2.2	Erzeugung persistenter Programme	152
3.2.3	Performancevergleich	156
3.3	Run-Time Type Identification	158
3.3.1	Arbeitsweise mit RTTI-Klassen	159
3.3.2	Geschachtelte Strukturen generisch auslesen	161
3.3.3	Export eines beliebigen Datenbanktabelleninhalts mit Überschriftenzeile	165
3.4	SQL-Performance und effiziente Datenbankkommunikation	169
3.4.1	Selektion der benötigten Felder	171
3.4.2	Vermeiden von SELECT...CHECK-Anweisungen	171
3.4.3	Aggregationsfunktionen	172
3.4.4	Lesen mehrerer Datenbanktabellen	173
3.4.5	Update mehrerer Datensätze	174
3.4.6	Kopieren von Datenbanktabellen in interne Tabellen	175
3.4.7	Abschalten der Änderungsprotokollierung für Tabellen	177
3.5	Zeichenkettenverarbeitung	182
3.5.1	Strings versus C-Felder	184
3.5.2	Zerlegen eines Strings in gleich große Blöcke	186
3.5.3	Einlesen einer Textdatei in eine interne Tabelle mit Zeilentyp STRING	187
3.5.4	Einfügen von Zeichenketten	190

3.6	Datumsfelder	191
3.6.1	Berechnung der Differenz zwischen zwei Daten	192
3.6.2	Letzter Tag eines Monats	193
3.6.3	Anzahl der Schaltjahre zwischen zwei Jahren	194
3.6.4	Bestimmung des nächsten Arbeitstages	195
3.6.5	Berechnung von Kalenderwochen und Wochentagen	197

4 GUI- und Desktop-Entwicklung 201

4.1	GUI-Controls und das Control Framework	201
4.1.1	GUI-Controls	202
4.1.2	Control Framework	203
4.1.3	Einbinden eines GUI-Controls	204
4.1.4	Mehrere GUI-Controls einsetzen	208
4.2	ALV-Grid-Control	211
4.2.1	Daten aus mehreren Quellen (mit eingeschränkter Spaltenselektion) darstellen	211
4.2.2	Einfaches Drop-Down-Listefeld im Grid darstellen	219
4.2.3	Drop-Down-Listefeld nur für bestimmte Zellen im Grid darstellen	226
4.2.4	Reaktion auf Hotspot-Clicks und Setzen eines Filters	232
4.3	Frontend-Kommunikation	240
4.3.1	Datenaustausch mit dem Clipboard	242
4.3.2	Mit dem Dateisystem des Frontends arbeiten	244
4.3.3	Export eines beliebigen Tabelleninhalts in eine durch Trennzeichen separierte Liste	250
4.3.4	Ausführen eines Programms auf dem Präsentationsserver	255
4.3.5	Arbeiten mit der Windows-Registry	258
4.3.6	Fortschrittsanzeige in der Statusleiste des SAP GUI	262
4.4	Dynpros und ABAP Objects	264

5 Kommunikation mit der Außenwelt 275

5.1	Kommunikation mit dem Applikationsserver	275
5.1.1	Interne Tabelle auf dem Applikationsserver speichern und einlesen	276
5.1.2	Kommandos auf dem Applikationsserver ausführen	281
5.1.3	Externe Kommandos ausführen	284
5.2	RFC	290
5.2.1	Erstellen eines externen RFC-Client-Programms	291
5.2.2	Externer RFC-Client mit dynamischer Tabellendefinition	303
5.2.3	Erstellen eines externen RFC-Servers	309
5.2.4	Externer RFC-Server als integrierte Applikation	315
5.3	FTP-Kommunikation	318
5.3.1	Kommunikation mit FTP-Servern	319

5.3.2	Ausführen von FTP-Kommandolisten zum Transfer von Dateien	324
5.3.3	FTP-Transfer von Dateien in interne Tabellen	330
5.4	Desktop-Integration	334
5.4.1	COM-Klassen	334
5.4.2	COM-Klasse aufrufen	336
5.4.3	Datenaustausch mit Word, Excel & Co.	345

6 SAP GUI Scripting 375

6.1	Einleitung und Überblick	375
6.1.1	Technologischer Hintergrund	376
6.1.2	Voraussetzungen für das GUI Scripting	380
6.1.3	Aufzeichnen und Abspielen eines Scripts	383
6.1.4	Konstanten für virtuelle Keys und Funktionstasten	390
6.2	Fortgeschrittene Scripting-Applikationen	391
6.2.1	Überblick	391
6.2.2	Verbindung mit dem SAP GUI-Prozess herstellen	392
6.2.3	Verbindung mit dem SAP-System herstellen	395
6.2.4	Ermitteln einer bestimmten SAP-Session	397
6.2.5	Anmelden am SAP-System per Script	399
6.2.6	Abmelden vom SAP-System per Script	401
6.2.7	Einsatz der Scripting-Support-Routinen	403
6.2.8	Exportieren einer beliebigen Liste	405

A Glossar 417

B Tabellen des Flugdatenmodells 425

C Konstanten für virtuelle Funktionstasten-Codes 429

D ABAP-Systemfelder 433

E Nützliche Eingaben im OK-Code-Feld 439

E.1	Direkter Aufruf von Transaktionen (Modusverwaltung)	439
E.2	Batchinput	440
E.3	ABAP-Debugging	440
E.4	Pufferkommandos	440

F	Die Autoren	441
	Index	443