



Ulrich Mende

Moderne Workflow- Programmierung mit ABAP[®] Objects

Handbuch für Entwickler

dpunkt.verlag



Dr. Ulrich Mende ist seit 12 Jahren als selbstständiger R/3-Berater, Entwickler und Trainer im Bereich ABAP-Workbench und SAP Business Workflow tätig. Er hat in den letzten Jahren für so namhafte Firmen wie die Deutsche Börse AG, die E.ON AG, die RWE AG, die Berliner GASAG sowie die Energie und Wasser Potsdam GmbH komplexe Workflow-Anwendungen entwickelt und bis zur Produktivsetzung begleitet. Das vorliegende Buch entstand direkt aus seiner Entwicklertätigkeit in diesen Projekten.

Ulrich Mende

Moderne Workflow- Programmierung mit ABAP® Objects

Handbuch für Entwickler



dpunkt.verlag

Ulrich Mende
ulrich.mende@web.de

Lektorat: Dr. Michael Barabas
Copy-Editing: Ursula Zimpfer, Herrenberg
Satz: Josef Hegele, Birgit Bäuerlein
Herstellung: Birgit Bäuerlein
Umschlaggestaltung: Helmut Kraus, www.exclam.de
Druck und Bindung: M.P. Media-Print Informationstechnologie GmbH, 33100 Paderborn

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;
detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN:
Buch 978-3-86490-013-6
PDF 978-3-86491-213-9
ePub 978-3-86491-214-6

1. Auflage 2012
Copyright © 2012 [dpunkt.verlag](http://dpunkt.verlag.com) GmbH
Ringstraße 19 B
69115 Heidelberg

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung des Verlags urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

Es wird darauf hingewiesen, dass die im Buch verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

In dieser Publikation wird auf Produkte der SAP AG, Dietmar-Hopp-Allee 16, 69190 Walldorf/Deutschland, Bezug genommen. Bei den Bezeichnungen dieser Produkte handelt es sich um eingetragene und/oder nicht eingetragene Marken der SAP AG. Die SAP AG ist weder Autor noch Verleger dieses Buches und ist für seinen Inhalt nicht verantwortlich.

Alle Angaben und Programme in diesem Buch wurden mit größter Sorgfalt kontrolliert. Weder Autor noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

5 4 3 2 1 0

❖❖❖ *Für Kerstina* ❖❖❖

Inhaltsverzeichnis

Einleitung	1
Wer hat das Buch geschrieben?	1
Dank	1
Worum geht es in diesem Buch?	2
An wen richtet sich das Buch?	3
Hinweise zu den einzelnen Kapiteln	4
1 SAP Business Workflow im Überblick	11
1.1 Die Architektur des SAP Business Workflow	13
1.1.1 Definitionszeit	13
1.1.2 Workflows und Workitems zur Ausführungszeit	16
1.1.3 Bearbeiterfindung	16
1.1.4 Workflow als Reaktion auf Anwendungssignale	17
1.1.5 Vorkonfigurierte Workflow-Definitionen	17
1.1.6 Der typische Workflow-Benutzer	18
1.2 Ein typisches Workflow-Szenario	19
1.3 Dokumentation und Tutorial	20
1.4 Business Workplace	20
1.4.1 Mails im Business Workplace	21
1.4.2 Workitems mit Benutzerdialog in der Worklist	21
2 Workflow-Entwicklung – Tools	23
2.1 Workflow aus Entwicklersicht	23
2.1.1 Workflow als Framework	23
2.1.2 Schichtenmodell einer Workflow-Anwendung	23
2.1.3 Paketorientierte Workflow-Entwicklung in der SE80 ...	24
2.1.4 Das Workflow-Bereichsmenü SWLD	25

2.2	Workflow-Beispiellösungen der SAP	26
2.2.1	Tutorials	26
2.2.2	Workflow-Anwendung »Urlaubsantrag«	27
2.2.3	Workflows der Aufgabengruppe WFUNIT	27
2.2.4	Workflows der Aufgabengruppe WF_Verify	28
2.3	SAP Service Marketplace	29
2.3.1	SAP Support Portal mit Hinweissystem	29
2.3.2	SDN-Community	32
2.3.3	SAP Help online und offline	33
3	ABAP Objects – Grundlagen	35
3.1	Objektorientierung als Abbild der realen Welt	35
3.2	Klassen und ihre Komponenten	36
3.2.1	PUBLIC-, PROTECTED- und PRIVATE-Komponenten .	36
3.2.2	Statische und Instanzkomponenten	37
3.2.3	Abstrakte und finale Methoden und Klassen	37
3.2.4	Lokale und globale Klassen	38
3.2.5	Objekte und Referenzen	40
3.2.6	Zugriff auf Klassen- und Instanzkomponenten	42
3.3	Vererbung und Casting von Referenzen	44
3.3.1	Klasse CL_VATER	46
3.3.2	Klasse CL_SOHN	48
3.3.3	Klasse CL_ENKEL	49
3.3.4	Beispielprogramm zur Vererbung	50
3.3.5	Upcasting und Downcasting von Referenzen	52
3.4	Events	55
3.4.1	Prinzip Publish and Subscribe	55
3.4.2	Event Handling	56
3.4.3	Events im Workflow-Kontext	57
3.5	Interfaces	57
3.5.1	Definition von Interfaces	59
3.5.2	Interfaceimplementierung und -referenzen	59
3.6	Polymorphie	61
3.6.1	Allgemeines	61
3.6.2	Polymorphie durch Vererbung	63
3.6.3	Polymorphie über Interfaces	64
3.7	Ausnahmen und Ausnahmeklassen	64
3.7.1	Klassische Ausnahmen ohne Ausnahmeklassen	64
3.7.2	Ausnahmeklassen	66
3.7.3	Ausnahmetexte aus dem OTR	75

3.8	Zugriff auf die Klassendefinition im Repository	76
3.9	Dynamischer Aufruf von Klassenmethoden	78
3.9.1	Parameter vom Typ REF TO dynamisch übergeben	82
3.10	Run Time Type Services	86
3.10.1	Run Time Type Identification (RTTI)	86
3.10.2	Run Time Type Creation (RTTC)	89
3.11	Übernehmen lokaler Klassen in das Repository	91
4	Durchgängiges Beispiel des Buches: Kreditverträge	93
4.1	Aufgabenstellung	93
4.2	Tabelle ZVERTRAG	93
4.3	Vertragstransaktion	94
4.4	Statusdiagramm	94
4.5	Implementierung (DB und ABAP)	95
4.6	Quelltext der Vertragstransaktion	98
4.7	Quelltext zur Berechnung des Ratenplans	102
5	Die neue Welt: ABAP Objects im Workflow	107
5.1	Problemstellung	107
5.2	IF_WORKFLOW	108
5.3	Vertragsbearbeitung im Workflow – Klasse ZCL_VERTRAG ...	112
5.3.1	Attribute MV_VERNR und MS_VERTRAG	112
5.3.2	Methoden ANZEIGEN, AENDERN, GENEHMIGEN .	113
5.3.3	Ereignisse CREATED, CHANGED und CANCELLED .	115
5.3.4	Erste Tests der Klasse ohne Interface IF_WORKFLOW	116
5.4	Einfachste Ausprägung von IF_WORKFLOW	116
5.5	Erweiterte Ausprägung IF_WORKFLOW	119
6	Workflow-Container und Workflow-Ereignisse	125
6.1	Konzept	125
6.2	Workflow-Container	127
6.2.1	Die Klasse CL_SWF_CNT_CONTAINER	127
6.2.2	Datenflüsse zwischen Containern	133
6.2.3	Containerpersistenz auf der Datenbank	136
6.3	Ereigniserzeugung	141
6.3.1	Universelle Ereigniserzeugung in ABAP-Programmen ..	141
6.3.2	Erzeugung von BOR-Events mit SWE_EVENT_CREATE	145
6.3.3	Ereigniserzeugung über Änderungsbelege	145
6.3.4	Weitere Methoden der Ereigniserzeugung	150

6.4	Ereignis-Verbraucher-Kopplungen	150
6.4.1	Typkopplung mit Funktionsbaustein oder Methode	151
6.4.2	Verbrauchertyp	153
6.4.3	Verbraucher-Funktionsbaustein	153
6.4.4	Check-Funktionsbaustein	153
6.4.5	Verbrauchertyp-Funktionsbaustein	154
6.4.6	Typkopplung aktiv, Ereignis-Queue zulassen	155
6.4.7	Instanzkopplung	155
6.4.8	Einträge in den Koppeltabellen durch Workflow	156
6.4.9	Workflow-Startbedingungen	157
6.5	Ereignisverarbeitung	159
6.5.1	Ablauf der Ereignisverarbeitung	159
6.5.2	Erzeuger- und Verbraucherkontext	159
6.5.3	Ereignisfluss bei Vererbung	160
6.6	Laufzeitwerkzeuge für Ereignisse	161
6.6.1	Ereignis auslösen	161
6.6.2	Ereignis simulieren	162
6.6.3	Ereignis-Trace	163
6.6.4	Ereignis-Queue	164
7	Organisationsmanagement und Regeln	167
7.1	Konzept	167
7.2	Bestandteile des Organisationsmanagements	168
7.2.1	Organisationseinheiten	168
7.2.2	Besetzungsplan	168
7.2.3	Aufgabenprofil	169
7.3	Einfache Pflege der Aufbauorganisation	170
7.4	Infotypen	171
7.4.1	OM-Typeebene und OM-Objektebene	171
7.4.2	OM-Objekttypen (Infotyp 1000)	173
7.4.3	Verknüpfungen zwischen Objekttypen (Infotyp 1001) ..	173
7.4.4	Weitere Infotypen	175
7.4.5	Zuständigkeiten	175
7.5	Regeln und Regelauflösung	176
7.5.1	Konzept	176
7.5.2	Regeldefinition	177
7.5.3	Regeln mit Funktionsbausteinen	178
7.5.4	Regeln mit Zuständigkeiten	179

7.6	Dynamische Eingangskörbe	182
7.6.1	Aufgabenstellung	182
7.6.2	Das Prinzip der dynamischen Eingangskörbe	183
7.6.3	Pflege der Bestandteile dynamischer Eingangskörbe ...	185
8	Standardaufgaben anlegen und verwenden	189
8.1	Konzept Standardaufgaben	189
8.2	Standardaufgaben anlegen	190
8.2.1	Objektyp und Methode	190
8.2.2	Hintergrundaufgabe und Dialogaufgabe	192
8.2.3	Verarbeitungsende bestätigen	193
8.2.4	Container und Initialwerte	194
8.2.5	Aufgabe mit Rückgabewert	196
8.2.6	Workitem-Text	199
8.2.7	Beschreibung	200
8.2.8	Auslösende Ereignisse	201
8.2.9	Beendende Ereignisse	202
8.2.10	Mögliche Bearbeiter	203
8.2.11	Zuständige Bearbeiter – Defaultregeln	205
8.3	Standardaufgaben starten	206
8.3.1	Start aus Workflow (Standardfall)	206
8.3.2	Start über Ereignis	206
8.3.3	Start aus Programm	206
9	Workflow-Muster anlegen und verwenden	209
9.1	Konzept Workflow-Muster	209
9.2	Workflow-Interface	210
9.2.1	Workflow-Container	211
9.2.2	Initialwerte	213
9.2.3	Aufgabentext	213
9.2.4	Aufgabenbeschreibung	213
9.2.5	Auslösende Ereignisse	214
9.3	Workflow-Definition	215
9.3.1	Die Arbeit mit dem grafischen Editor	215
9.3.2	Der Bedingungseditor	217
9.3.3	Übersicht Schritttypen	218
9.3.4	Schritttyp Aktivität	220
9.3.5	Schritttyp Webaktivität	223
9.3.6	Schritttyp Mailversand	223
9.3.7	Schritttyp Formular	224
9.3.8	Schritttyp Benutzerentscheidung	225

9.3.9	Schritttyp Warten auf Ereignis	226
9.3.10	Schritttyp Bedingung	227
9.3.11	Schritttyp Mehrfachbedingung	228
9.3.12	Schritttyp Paralleler Abschnitt	229
9.3.13	Schritttyp UNTIL-Schleife	230
9.3.14	Schritttyp WHILE-Schleife	232
9.3.15	Schritttyp Ablaufsteuerung	233
9.3.16	Schritttyp Containeroperation	234
9.3.17	Schritttyp Ereigniserzeuger	235
9.3.18	Schritttyp Ad-hoc-Anker	235
9.3.19	Schritttyp Block	235
9.3.20	Schritttyp Lokaler Workflow	236
9.4	Container in der Workflow-Definition	236
9.5	Workflow-Grunddaten	237
9.5.1	Versionsunabhängig (Aufgabe)	238
9.5.2	Versionsabhängig (aktuelle Workflow-Version)	240
9.6	Einfacher Vertrags-Workflow	243
9.6.1	Technische Workflow-Beschreibung und Pseudocode	243
9.6.2	Workflow-Muster zao_main	244
9.7	Bearbeiterfindung im Workflow	249
9.7.1	Allgemeines	249
9.7.2	Definition der möglichen Bearbeiter	249
9.7.3	Zuständige und ausgeschlossene Bearbeiter	250
9.7.4	Ermittlung der Empfänger	251
9.8	Arbeiten mit dem Business Workflow Explorer	252
9.8.1	Zweck des Business Workflow Explorer	252
9.8.2	Auswahl des Suchbereiches	253
9.8.3	Anzeige und Bearbeitung von Objekten	254
9.9	Lesen der Workflow-Definitionen	254
9.10	Starten von Workflows	257
9.10.1	Start von Workflows über ABAP OO	257
9.10.2	Start von Workflows über Ereignisse	258
10	Klassenbasierte Techniken im Workflow	259
10.1	SAP-Klassen im Workflow-Umfeld	259
10.2	Iterator-Klassen, Factory-Klassen und Utility-Klassen	260
10.2.1	Iterator-Klassen	260
10.2.2	Factory-Klassen und Factory-Methoden	263
10.2.3	Utility-Klassen	265

10.3	Terminüberwachung	266
10.3.1	Problem und Lösungsprinzip	266
10.3.2	Standardfall mit Deadline-Workitems	266
10.3.3	Berechnete Termine mit Deadline-Workitem	268
10.3.4	Obsolet setzen eines Workitems	269
10.4	Nebenmethoden	270
10.4.1	Problem und Lösungsprinzip	270
10.4.2	Methode zur Vertragsanzeige	271
10.4.3	Eintrag als Nebenmethode in die Aktivität	271
10.4.4	Amodaler Aufruf der Nebenmethode	272
10.5	Objektreferenzen in die Workitem-Vorschau einstellen	273
10.5.1	Referenzen auf bestehende Workflow-Klassen	273
10.5.2	Referenzen auf spezielle Anzeigeklassen	274
10.6	Programmier-Exits für Workitems	279
10.6.1	Das Interface IF_SWF_IFS_WORKITEM_EXIT	279
10.6.2	Logging mit CHECK-POINT-Gruppe	279
10.6.3	Automatisches Zurücklegen eines Workitems	282
10.7	Fortgeschrittene Workflow-Definition	284
10.7.1	Lokale Ereignisse und Ausnahmen	284
10.7.2	Lokale Ereignisse und komplexe Bedingungen	285
10.7.3	TRY/CATCH im Workflow: Blöcke und Ausnahmen ..	287
10.7.4	Lokale Workflows und lokale Ereignisse	287
10.8	Dynamische Blöcke	288
10.8.1	Dynamisch sequenzieller Block = Workflow-Iterator ...	289
10.8.2	Dynamisch paralleler Block	289
10.8.3	Beispiel: Bewertungs-Workflow für Vertrag	290
10.9	Ad-hoc-Funktionen im Workflow	294
10.9.1	Ad-hoc-Aufgabenauswahl	294
10.9.2	Ad-hoc-Bearbeiterauswahl	296
10.9.3	Ad-hoc-Workflow einbinden	298
10.10	Reaktion eines Workflows auf externe Ereignisse	300
10.10.1	Standardreaktionen über Grunddaten des Workflows ..	301
10.10.2	Modellierte Reaktionen über Ereignisempfänger	302
10.11	Klassenverwendung im Workflow ohne Standardaufgaben	304
10.11.1	Konzept	304
10.11.2	%-%-Zugriff ohne IF_WORKFLOW	305
10.11.3	&&-Zugriff über ungebundene Containerreferenzen ..	306
10.11.4	&&-Zugriff über gebundene Containerreferenzen	307
10.11.5	Geschachtelter Zugriff mit %-% und &&	309
10.11.6	Methoden mit mehreren Exportparametern	310
10.11.7	Objektreferenzen on-the-Flight	311

11	Klassenbasierte Eigenentwicklungen im Workflow	313
11.1	Utility-Klasse ZCL_DATE: Datumsberechnung auf Kalendern ...	313
11.1.1	Verwendung von Datumsberechnungen	313
11.1.2	Implementierung von ZCL_DATE	313
11.1.3	Verwendung von ZCL_DATE in Ausdrücken	316
11.2	Utility-Klasse ZCL_COND: komplexe Bedingungen	317
11.2.1	Konzept komplexer Bedingungen im Workflow	317
11.2.2	Tabelle ZCONDITION	318
11.2.3	Implementierung von ZCL_COND	318
11.2.4	Verwendung von ZCL_COND in ZCL_VERTRAG ...	319
11.2.5	Setzen und Prüfen von Vertragsbedingungen	320
11.2.6	Aufruf von ZCL_COND in Ausdrücken	325
11.3	Utility-Klasse ZCL_CONST: Datenbankkonstanten	325
11.3.1	Konzept »variabler« Konstanten	325
11.3.2	Tabelle der Konstanten ZCONST	326
11.3.3	Aufruf der Konstanten über ZCL_CONST	326
11.3.4	Implementierung von ZCL_CONST	327
11.3.5	Einführen neuer Konstanten	329
11.3.6	Verwendung der Konstanten im Workflow	329
11.4	Utility-Klasse ZCL_SYSTEM: generische Systemmethoden	331
11.4.1	Inhalt von ZCL_SYSTEM	331
11.4.2	Erzeugen von Klassenreferenzen im Workflow	332
11.4.3	GENERICINSTANTIATE von CL- und BOR-Referenzen	334
11.4.4	MAKE_CL_REF für den Aufruf in Ausdrücken	335
11.5	Kommunikation zwischen parallelen Subworkflows	336
11.5.1	Kommunikation über Ereignisse	337
11.5.2	ZCL_SYSTEM: Lesen fremder Workflow-Container ...	338
11.6	Mails im Workflow-Umfeld	341
11.6.1	Mail senden mit Schritttyp Mailversand	341
11.6.2	ZCL_SYSTEM: Mail mit eigenem Absender	342
11.6.3	ZCL_SYSTEM: Mail mit SAP – Text und Containerbezug	344
11.6.4	HTML-Mails mit Rückantwort	349
12	Business Workplace als Userinterface	355
12.1	Allgemeines zum Business Workplace	355
12.2	Fenster im Business Workplace	357
12.2.1	Das Mappenfenster (1)	357
12.2.2	Das Inboxfenster (2)	360
12.2.3	Das Vorschaufenster (3)	361

12.3	Funktionen im Workplace	362
12.3.1	Funktionen der Workflow-Inbox	362
12.3.2	Persönliche Workflow-Einstellungen	367
12.3.3	Workflow-Vertretungen	367
12.3.4	Vertretungen in Tabelle HRUS_D2	370
12.4	Die Anpassung der Inbox	370
12.4.1	Layoutfunktionen des ALV-Grids	370
12.4.2	Verwendung dynamischer Anwenderspalten	373
13	Webservices im Workflow	377
13.1	Webservices im Workflow	377
13.2	Das Webservice-Framework	378
13.2.1	Erstellen eigener Webservices	379
13.2.2	Einbinden fremder Webservices	380
13.3	Beispiel: Einbinden eines Webservice	381
13.3.1	SOAP-Webservice für Wetterdienste	381
13.3.2	Einbinden eines Webservice in SAP	382
13.3.3	Generierte Klasse ZCO_GLOBAL_WEATHER_SOAP ..	385
13.3.4	Anlegen logischer Ports	387
13.4	Aufruf des Webservice	389
13.4.1	Aufruf des Service GlobalWeather in einem Programm ..	389
13.4.2	Einbinden des Webservice in den Workflow	391
14	Das Workflow-Laufzeitsystem	393
14.1	Konzept	393
14.2	Die Tabellen des Laufzeitsystems	393
14.2.1	Die Tabelle SWWWIHEAD der Workitem-Köpfe	394
14.2.2	Die Tabelle SWW_WI2OBJ für Objektreferenzen	396
14.2.3	Die Tabelle SWWORTASK	397
14.2.4	Die Tabelle SWWUSERWI der Benutzerzuordnung ...	397
14.2.5	Views	397
14.2.6	Programmierter Zugriff auf die Laufzeittabellen	397
14.3	Wichtige Funktionsbausteine und Klassen	400
14.3.1	SAP_WAPI-Bausteine	400
14.3.2	Klassen des Laufzeitsystems	401
14.4	Die Arbeitsweise des Laufzeitsystems	402
14.4.1	Datenfluss über Container	406
14.4.2	Aktionen auf Workitems	406
14.4.3	Termin- und Fehlerüberwachung	407
14.4.4	Business Workplace als Sicht auf die Laufzeittabellen ..	407

15	Tests und Fehlersuche	409
15.1	Workflow-Instanzen suchen	409
15.1.1	Workitem-Selektion mit ihren Anzeigemöglichkeiten . . .	409
15.1.2	Workflow-Instanzen zu Objekttyp	411
15.1.3	Workflow-Instanzen zu Objektinstanz	412
15.2	Workflow-Protokoll mit seinen diversen Sichten	413
15.2.1	ActiveX-Protokoll und seine Sichten	414
15.2.2	Technisches Workflow-Protokoll und seine Sichten	416
15.2.3	Grafisches Workflow-Protokoll	417
15.2.4	Anzeigen, Ändern und Beenden von Workitems	417
15.2.5	Testen der Fristverletzung an Ereignisempfängern	418
15.3	Debug-Report für Batch-Workitems	420
15.3.1	Notwendigkeit	420
15.3.2	Anwendung	421
15.3.3	Quelltext des Reports ZAO_DEBUGWI	422
15.4	Hintergrundschrte im Status STARTED	424
15.4.1	Dump-Analyse	424
15.4.2	Wiedernstarten nach Fehlerkorrektur	426
15.5	Nützliche Hilfsprogramme	426
15.5.1	Container-Update	426
15.5.2	SBE fortsetzen mit Simulation der Auswahl	427
15.5.3	Benutzer an Dialog-Workitems ändern	428
15.5.4	Ultima Ratio nach 22 Uhr	430
15.6	Up- und Download von Workflow-Mustern	434
15.6.1	Problematik	434
15.6.2	Noch kein SAPlink-Plug-in für Workflows	435
15.6.3	Up- und Download von Klassen im XML-Format	436
15.6.4	Up- und Download von Workflow-Mustern als XML . . .	436
15.7	Workflow-Transporte	438
16	Workflow-Administration	439
16.1	Aufgabe und Festlegung der Administratoren	439
16.1.1	Allgemeine Aufgaben des Workflow-Administrators . . .	439
16.1.2	Customizing verifizieren	440
16.1.3	Workflow-Administrator global einstellen	441
16.1.4	Workflow-Administrator pro Workflow-Definition	441
16.2	Batchjobs konfigurieren und einplanen	442
16.2.1	Terminüberwachung	442
16.2.2	Workitem-Fehlerüberwachung	443
16.2.3	Bedingungsauswertung	443

16.3	E-Mail-Notification für SAP-Gelegenheitsbenutzer	444
16.3.1	Pflege der E-Mail-Adresse für das Weiterleiten	444
16.3.2	Pflege der persönlichen E-Mail-Adresse in den Benutzerdaten	444
16.3.3	Erzeugen von E-Mails zu neuen Workitemen	445
16.3.4	Versenden der Internetmails über SAPconnect	447
16.4	Reaktion auf Fehlermeldungen des Laufzeitsystems	448
16.5	Typische Workflow-Fehler und ihre Behebung	449
16.5.1	Workitemen ohne Bearbeiter	449
16.5.2	Workitemen mit Terminüberschreitungen	450
16.5.3	Diagnose fehlerhafter Workitemen	450
16.5.4	Laufzeitpuffer synchronisieren	451
16.6	Restarten unterbrochener Workflows	452
16.6.1	Restart fehlerhafter Workflows	452
16.6.2	Restart nach Systemabsturz oder Schrittdump	452
16.7	Analysen	453
16.7.1	Workitemen pro Aufgabe	453
16.7.2	Workitemen nach Bearbeitungsdauer	454
16.7.3	Workload-Analyse	455
16.8	Reorganisation der Laufzeitdaten	457
16.8.1	Allgemeines	457
16.8.2	Auslastung der Workflow-Laufzeittabellen	457
16.8.3	Gezieltes Löschen von einzelnen Workflows	458
16.8.4	Löschen von Workflows im Batch	458
16.8.5	Archivieren von Workflows im Batch	462
16.9	Transportieren von Workflows	465

Anhang

A	Die alte Welt – BOR-Objekte im Workflow	467
A.1	Historisches	467
A.2	Anlegen eines neuen Objekttyps	468
A.2.1	Die Objektliste mit ihren Bestandteilen	469
A.2.2	Programm des Objekttyps als Subroutinenpool	471
A.2.3	Interface	472
A.2.4	Freigeben von Objekttyp und Komponenten	473
A.3	Attribute	474
A.3.1	Datenbankattribute mit GET_TABLE_PROPERTY ...	474
A.3.2	Objektreferenzen und Makros für den Objektzugriff ..	477
A.3.3	Implementierung und Aufruf virtueller Attribute	479

A.4	Implementierung und Aufruf von Methoden	484
A.4.1	Synchrone Anzeigemethode als Defaultmethode	487
A.4.2	Synchrone Dialogmethode mit Ergebnisparameter	488
A.4.3	Synchrone Hintergrundmethode mit Parametern	491
A.4.4	Asynchrone Dialogmethode ohne alles	492
A.4.5	Methoden mit Ausnahmen	493
A.4.6	Überdefinieren von Methoden	495
A.5	Ereignisse	495
A.6	Prüfen und Testen einer Objektimplementierung	496
A.7	Vererbung und Delegation	498
A.7.1	Vererbung und Delegation im BOR	498
A.7.2	Vererbung und keine Delegation für ABAP-Klassen	501
A.7.3	Ereignisbehandlung bei Vererbung und Delegation	501
A.8	Standardisierte Schnittstellen – Container	502
A.8.1	Container in Programmen – Include <CNTN01>	503
A.8.2	Persistente Instanzen von Containern	507
A.8.3	Container im Workflow	508
A.9	Spezielle Objekttypen	509
	Literatur	511
	Inhalt des Downloadbereiches	513
	Abkürzungsverzeichnis	517
	Index	519

Einleitung

Wer hat das Buch geschrieben?

Autor des Buches ist Dr. Ulrich Mende, der auch bereits das Vorgängerbuch »Workflow und ArchiveLink mit SAP« [2] beim dpunkt.verlag geschrieben hat. Er arbeitet seit 13 Jahren selbstständig als Entwickler für SAP Business Workflow, in den letzten Jahren hauptsächlich im IS-U-(Industry Solution Utilities) Umfeld für große Energieversorger. Parallel dazu führt er Workshops für Workflow-Entwickler durch.

ulrich.mende@web.de

Dank

Der ganz besondere Dank des Autors gilt der Fa. affinis consulting GmbH in Hamburg. Ohne deren großzügige Unterstützung durch die Bereitstellung eines SAP-Entwicklungssystems wäre dieses Buch nicht zustande gekommen.

Insbesondere hat Herr Marc Trümer (affinis intelligent solutions GmbH & Co. KG) nicht nur organisatorisch, sondern auch fachlich (Kap. 13) vielfältig zum Gelingen des Buches beigetragen.

Weiterhin bedankt sich der Autor herzlich bei seinen Fachkolleginnen und Kollegen, Frau Eva Schleißheimer, Herrn Stefan Porges, Herrn Ayhan Toraman, Herrn Hüdayi Mengus und Herrn Mario Gratz, für thematische Diskussionen, Hinweise und Korrekturen zum Buch.

Dem dpunkt.verlag sei vielmals gedankt für die jederzeit konstruktive und vertrauensvolle Zusammenarbeit, insbesondere dem Lektor Herrn Dr. Barabas sowie der Korrektorin Frau Ursula Zimpfer.

Worum geht es in diesem Buch?

Längst hat sich SAP Business Workflow als modulübergreifende Querschnittstechnologie zur Abbildung und automatisierten Bearbeitung komplexer, arbeitsteiliger Prozesse im SAP-Umfeld etabliert. Viele Unternehmen haben erkannt, welches enorme Rationalisierungspotenzial diese Technologie für die Prozessbearbeitung in ihrem Unternehmen enthält.

Stellvertretend seien hier die Workflow-Prozesse im IS-U (Industry Solution Utilities) genannt, die vermutlich die komplexesten Prozesse im SAP-Umfeld überhaupt darstellen. So hat ein Netzbetreiber-Workflow im Lieferantenwechsel ca. 100 Schritte und kommuniziert via Marktkommunikation (IDOCs, B2B) mit Workflows, die parallel auf den SAP-Systemen fremder Unternehmen (Lieferanten und Messstellenbetreiber) laufen.

Entsprechend hoch ist der Aufwand, den die Unternehmen inzwischen für die Entwicklung und Administration von Workflows betreiben. Viele Entwickler, die sich früher ausschließlich mit ABAP (Advanced Business Application Programming) befasst haben, müssen sich nun in der einen oder anderen Form mit der Workflow-Entwicklung vertraut machen.

Inzwischen gibt es zum Thema Workflow-Entwicklung mehrere Bücher, in denen die Technologie aus Entwicklersicht von Anfang an detailliert besprochen wird [1], [2], [3]. Die Grundlagen der Workflow-Entwicklung sind daher relativ gut beschrieben und deshalb nicht Hauptgegenstand dieses Buches.

Dieses Buch konzentriert sich dagegen auf die erheblichen Veränderungen in der Workflow-Entwicklung, die durch ABAP Objects möglich bzw. nötig werden.

Mit dem Entstehen der Workflow-Technologie wurden die sogenannten BOR-Objekttypen (BOR = Business Object Repository) als quasi objektorientierte Verschalung aller ABAP-Anwendungen eingeführt. BOR-Objekttypen (BOR-OT) waren von Anfang an *die* Workflow-Objekttypen. Für ABAP-Entwickler war die stark makroorientierte Programmierung von und mit BOR-Objekttypen oft sehr gewöhnungsbedürftig, was vermutlich auch ein Grund dafür ist, dass es immer noch wenig Workflow-Entwickler gibt.

Um das Jahr 2000 erschien ABAP Objects als echter objektorientierter Sprachbestandteil von ABAP. Viele Anwendungen nutzten sofort die stark verbesserten Möglichkeiten der Programmierung, die sich damit boten. Workflow-Entwickler mussten allerdings bis zum Basisrelease 6.20 warten, bis sie neben BOR-OT auch ABAP-Klassen in Workflows benutzen konnten. (Dabei muss man den Zeitverzug von

einigen Jahren beachten, der zwischen der Freigabe eines Release bei SAP und dessen flächendeckender Einführung beim Anwender besteht.) Dies führte in vielen Fällen dazu, dass die BOR-OT nur die Workflow-Verschaltung waren, in deren Inneren sofort Klassenmethoden aufgerufen wurden.

Inzwischen kann ein Releasestand von SAP Basis von 6.20 und höher durchaus vorausgesetzt werden. Der Nutzung von ABAP Objects anstelle von BOR-OT steht also nichts mehr im Wege. Dass sich viele Entwickler dennoch vor diesem Schritt scheuen, mag an der jahrelangen Gewöhnung an das BOR und dessen Business-Intelligenz (in Form vieler Millionen Zeilen Code) liegen.

Insofern ist es beruhigend, zu wissen, dass hier kein »Entweder-oder« notwendig ist. BOR-OT und ABAP Objects können in friedlicher Koexistenz in ein und demselben Workflow leben. Die alten, bewährten, aber unverzichtbaren Dinge erledigen BOR-Objekte; neue Funktionalitäten werden über ABAP Objects realisiert.

Oft hört man von Entwicklern folgende Meinung: Wozu eigentlich Objektorientierung? Das verkompliziert alles nur; ein einfacher Report tut es auch.

In diesem Sinne stellt SAP Business Workflow eine echte »Daseinsberechtigung« für ABAP Objects dar. Letztlich ist der Workflow nämlich ein komplexes Framework, das mit beliebigen, unbekannten Objekten gleichartig hantieren soll. In diesem Umfeld kommen alle Konzepte der Objektorientierung wie Kapselung, Vererbung, Interfaces, Polymorphie, Events und Ausnahmeklassen auf völlig natürliche Art zum Tragen.

Dies zu vermitteln ist das hauptsächliche Ziel des Buches. Für die »Praktiker« unter den Entwicklern kann es den Zugang zur Objektorientierung erleichtern.

An wen richtet sich das Buch?

Das Buch richtet sich an EDV-Spezialisten, die auf die eine oder andere Weise mit SAP Business Workflow zu tun haben. Das können sein:

- Studenten der Informatik
- Entwickler bei SAP-Anwendern
- Systemverantwortliche bei SAP-Anwendern (Administratoren, Entscheider)
- Angestellte oder freie SAP-Berater

Das Buch richtet sich zunächst an alle Workflow-Entwickler, die es leid sind, immer wieder Container mit SWC_GET_ELEMENT auszuropa-

cken und am Ende der Methode mit SWC_SET_ELEMENT wieder einzupacken.

An solche also, die die modernen Methoden von ABAP Objects, die sie u.U. bereits in der reinen ABAP-Programmierung verwenden, nun auch direkt, ohne Bruch in der Workflow-Entwicklung benutzen möchten.

Für diejenigen unter den Lesern, die mit der Objektorientierung noch nicht vertraut sind, ist eine Einführung in ABAP Objects enthalten. Hier werden Begriffe wie Kapselung, Vererbung, Redefinition und Polymorphie praxis- und Workflow-bezogen dargestellt.

Und dann ist das Buch für solche Entwickler gedacht, die den Einstieg in die Workflow-Entwicklung wagen wollen bzw. sollen. Ihnen ist zu raten, von Anfang an mit der neuen Technologie zu arbeiten.

Aber auch Entwickler, denen der übliche Zugang zu ABAP Objects zu theoretisch ist, können hier u.U. einen besseren, weil praktischen Zugang zur Objektorientierung finden.

Workflow-Berater, die bereits über gute Kenntnisse in ABAP Objects und Workflow verfügen, können in diesem Buch viele konkrete Hinweise für ihre Projekte bekommen.

Workflow-Administratoren sollten auf jeden Fall das einführende Kapitel sowie die Kapitel zum Laufzeitsystem und zur Administration lesen.

Entscheider (Teamleiter), die sich einen Überblick über SAP Business Workflow verschaffen möchten, sollten das einführende Kapitel lesen. Außerdem gibt es in fast jedem Kapitel einen Punkt »Konzept«, der die jeweilige Grundidee erläutert.

Hinweise zu den einzelnen Kapiteln

Kapitel 1 – SAP Business Workflow im Überblick

Hier wird die grundlegende Architektur des SAP Business Workflow mit ihren wichtigsten Bestandteilen und Begriffen einführend dargestellt.

■ Neue Begriffe

Laufzeit, Definitionszeit, Bearbeiter, Workplace

Kapitel 2 – Workflow – Entwicklung Tools

Hier werden Hinweise zu den Entwicklungstools gegeben. Ausführlich wird das Workflow-Bereichsmenü SWLD vorgestellt. Empfohlen wird die paketorientierte Arbeit im Object Navigator (SE80). Weiterhin

werden zwei große Gruppen von Beispiel-Workflows von SAP beschrieben. Hinweise zum Thema Workflow im SAP Service Marketplace und in der SDN-Community (SDN = SAP Developer Network) runden dieses Kapitel ab.

■ Neue Begriffe

Schichtenmodell der Workflow-Entwicklung, Bereichsmenü SWLD, WFUNIT, WF_Verify, Online Help, SDN

Kapitel 3 – ABAP Objects – Grundlagen

In diesem Kapitel werden überblicksartig die notwendigen Grundlagen von ABAP Objects zusammengefasst. Es erspart dem Leser ohne Kenntnisse von ABAP Objects (ABAP OO) das Nachschlagen der Grundbegriffe in anderen Büchern. Wo es möglich ist, wird bereits Bezug auf die spätere Verwendung im Workflow-Umfeld genommen. Der erfahrene ABAP-OO-Entwickler kann dieses Kapitel ohne Nachteile überspringen.

■ Neue Begriffe:

Klassen, Komponenten, Attribute, Methoden, Sichtbarkeit, Ereignisse, Vererbung, Redefinition, Instanzen, Referenzen, Interfaces, Polymorphie, Ausnahmeklassen

Kapitel 4 – Durchgängiges Beispiel des Buches: Kreditverträge

Im Buch wird ein durchgehendes Beispiel zur Vertragsbearbeitung benutzt. In diesem Kapitel werden die Tabellen und Transaktionen kurz dargestellt, die dann in den Workflow-Klassen benutzt werden.

Kapitel 5 – Die neue Welt: ABAP Objects im Workflow

Hier werden die Erkenntnisse aus Kapitel 3 in den Workflow-Klassen verwendet. Ausführlich wird das Interface IF_WORKFLOW besprochen. Es wird gezeigt, wie die Ereignisse von ABAP Objects als Workflow-Ereignisse systemweit sichtbar gemacht werden.

■ Neue Begriffe:

Workflow-Interface IF_WORKFLOW, persistente Objektreferenz, Workflow-Ereignisse

Kapitel 6 – Workflow-Container und Workflow-Ereignisse

In diesem Kapitel werden Definition und Verwendung von Containern und Ereignissen im Workflow-Umfeld besprochen.

Für Container werden Beispielprogramme zu Definition, Datenflüssen, persistenter XML-Speicherung diskutiert.

Danach werden die Erzeugung und Verarbeitung von systemweiten Ereignissen detailliert besprochen. Neben den verschiedenen direkten und indirekten Methoden der Ereigniserzeugung liegt der Schwerpunkt auf der Ereignisverarbeitung. Letztere stellt die Reaktion auf Statusänderungen an Objektinstanzen dar. Es werden Beispiele für die Implementierung eigener Koppelbausteine diskutiert.

■ Neue Begriffe:

Containerklasse CL_SWF_CNT_CONTAINER, Strukturpersistenz, XML-Persistenz, Eventklasse CL_SWF_EVT_EVENT, Ereigniskopplung, Ereignisverbraucher, Check-Funktionsbaustein, Typ-Funktionsbaustein, Eventqueue, Ereigniscontainer, Datenfluss

Kapitel 7 – Organisationsmanagement und Regeln

Dieses Kapitel erläutert die Möglichkeiten zur Abbildung des Organisationsmanagements. Hier wird auf die sogenannten Infotypen, insbesondere auf die Objekte des Organisationsmanagements sowie die zwischen ihnen möglichen Beziehungen, eingegangen. Ziel ist die Nutzung der hinterlegten Informationen zur automatischen Bearbeiterfindung in Workflows.

■ Neue Begriffe:

Organisationseinheit, Stelle, Planstelle, Person, User, Regel, Zuständigkeit, Auswertungsweg, Bearbeiterfindung

Kapitel 8 – Standardaufgaben anlegen und verwenden

Es wird gezeigt, wie Einzelschrittaufgaben mit Bezug auf eine Klassenmethode angelegt werden.

■ Neue Begriffe:

Einzelschrittaufgabe, Hintergrundaufgabe, Aufgabencontainer, führendes Objekt, Datenfluss, mögliche Bearbeiter, generelle Aufgabe, Workitem-Text, Ausdruck, Defaultregel

Kapitel 9 – Workflow-Muster anlegen und verwenden

Es wird gezeigt, wie Workflows unter Benutzung von Klassen, Events und Einzelschrittaufgaben und Regeln angelegt werden.

■ Neue Begriffe:

Workflow, Workflow-Interface, führendes Objekt, Import-, Export- und lokale Containervariablen, startendes Ereignis, Workflow-

Definition, Aktivität, ausgewählte Bearbeiter, Alternative, Mehrfachverzweigung, paralleler Abschnitt, Schleife, Containeroperation, Steuerschritt, Workflow-Version, Transport

Kapitel 10 – Klassenbasierte Techniken im Workflow

Hier werden weitergehende klassenbasierte Techniken der Workflow-Definition und der Laufzeitsteuerung besprochen.

■ Neue Begriffe/Stichwörter:

Terminüberwachung, Nebenmethoden, Workitem-Vorschau, Workitem-Exit, dynamische Blöcke, Startbedingungen, Ad-hoc-Funktionen

Kapitel 11 – Klassenbasierte Eigenentwicklungen im Workflow

Hier werden einige in der Praxis der Workflow-Entwicklung entstandene, Workflow-taugliche Utility-Klassen besprochen. ZCL_DATE bietet Methoden zur kompakten Berechnung von Datums- und Zeitwerten auf Fabrikkalendern an.

Die Klasse ZCL_SYSTEM stellt Methoden zur generischen Erzeugung von Klassenreferenzen und zum Mailversand bereit.

Beide Klassen können im Downloadbereich zum Buch in XML-Form heruntergeladen und mit dem im Buch vorgestellten Report direkt ins System eingespielt werden.

■ Neue Begriffe/Stichwörter:

GENERICINstantiate, Datumsklasse, Konstantenklasse, Kommunikation zwischen Subflows, generischer Mailversand

Kapitel 12 – Business Workplace als Userinterface

In diesem Kapitel wird der Business Workplace als Userinterface für den Workflow-Benutzer besprochen. Es werden alle Funktionen auf Workitems erläutert sowie Möglichkeiten zur persönlichen Konfiguration der Workflow-Inbox dargestellt.

■ Neue Begriffe:

Worklist, Vorschau, Status, Statusdiagramm, Workitem anzeigen, ~ ausführen, ~ annehmen, ~ zurücklegen, ~ wiedervorlegen, ~ weiterleiten

Kapitel 13 – Webservices im Workflow

Es wird demonstriert, wie SOAP-Webservices angelegt und aus dem Workflow heraus aufgerufen werden können.

■ Neue Begriffe:

Webservice, SOAP-Protokoll, CL_PROXY_CLIENT, WSDL, logischer Port

Kapitel 14 – Das Workflow-Laufzeitsystem

In diesem Kapitel wird das Laufzeitsystem des SAP Business Workflow besprochen. Die hier dargestellte technische Sicht (Tabellen und Funktionsbausteine) kann das Verständnis für den Workflow bei Informatikern erfahrungsgemäß erheblich fördern.

■ Neue Begriffe:

Workitem, Top-Level-Workitem, Laufzeittabellen, WAPI-Bausteine

Kapitel 15 – Tests und Fehlersuche

In diesem Kapitel werden Verfahren zum Test und zur Fehlersuche vorgestellt. Dazu gehören die unterschiedlichen Transaktionen zum Suchen nach Workflows bzw. Workitems sowie die Arbeit mit den diversen Protokollen. Besonders interessant dürfte ein Report sein, der es ermöglicht, B-Workitems nachträglich im Dialog zu debuggen. Hinweise zur Bearbeitung von Kurzdumps sowie diverse nützliche Hilfsprogramme runden das Kapitel ab.

■ Neue Begriffe:

CHECKPOINT-GROUPS, Workflow-Protokolle, Debug-Report, Dump-Analyse

Kapitel 16 – Workflow-Administration

Dieses Kapitel beschreibt die Aufgabenstellungen und Lösungsansätze in der Arbeit eines Workflow-Administrators. Zunächst werden Fragen des Customizings besprochen. Die für die Administration von Workflows wichtigen Werkzeuge (Analysetools, Diagnosetools) werden erläutert. Es werden Möglichkeiten zur Reorganisation der Laufzeitdaten dargestellt.

■ Neue Begriffe:

Workflow-Customizing, E-Mail-Notification über neue Workitems, SAPconnect, Terminüberwachung, Fehlerüberwachung, Fehleritem, Diagnosereport, Workitem-Analyse, Workflow-Protokoll, Workload-Analyse, Archivierung von Workitems, SARA

Anhang – Die alte Welt: BOR-Objekte im Workflow

Die Objekttypen des Business Object Repository (BOR) werden als »alte« objektorientierte Verschalung der Anwendungssoftware erklärt. Dieser Anhang trägt der Tatsache Rechnung, dass viele Millionen Zeilen ABAP-Code in den vorhandenen BOR-Objekttypen ein großes Erfahrungspotenzial darstellen.

■ Neue Begriffe:

Objekttyp, Schlüssel, Attribut, Methode, synchron/asynchron, Parameter, Container, Ereignisse, Delegation, Vererbung, Interface

1 SAP Business Workflow im Überblick

Der *SAP Business Workflow* ermöglicht die rationelle, rechnergestützte Bearbeitung von komplexen, betriebswirtschaftlichen Abläufen. Der *SAP Business Workflow* unterstützt besonders solche Geschäftsprozesse, die immer wieder in gleichen oder ähnlichen Formen auftreten, die eine Reihe von Aktivitäten umfassen und die mehrere Personen oder Abteilungen involvieren. Aber auch einfache »Zwei-Schritt-Szenarios«, bei denen ein Mitarbeiter eine Änderung im System vornimmt, von der ein anderer Mitarbeiter benachrichtigt werden soll oder bei der eine weitere Aktion durch das System erforderlich ist, lassen sich mit dem *SAP Business Workflow* abbilden und automatisieren.

*Workflow für komplex
strukturierte, arbeitsteilige
Prozesse*

Der *SAP Business Workflow* ist dazu – genau wie die ABAP Workbench – anwendungsübergreifend in das SAP-Basisystem integriert und ergänzt die vorhandenen, umfassenden betriebswirtschaftlichen Funktionen des SAP-Systems optimal. Zusätzliche Lizenzkosten entstehen beim Einsatz von SAP Business Workflow nicht. Er umfasst insbesondere:

*Keine zusätzlichen
Lizenzkosten für SAP
Business Workflow*

- eine grafische Definitionsumgebung für Workflows
- das Laufzeitsystem zur Steuerung und Überwachung
- mehrere Workflow-Muster für verschiedene Szenarios zur Wiederverwendung

Im weiteren Sinne gehören zum *SAP Business Workflow* auch:

- in die Basis verlagerte Teile der Anwendungskomponente *Human Resources* zur Abbildung der Aufbauorganisation eines Unternehmens
- das *Business Object Repository* zur Definition und Implementierung von Objekten als zentrale Bestandteile eines jeden Workflows
- Interfaces für die Einbindung von ABAP-Klassen in den Workflow

Auch ohne den Einsatz von Workflows lassen sich über die prozessorientierte Integration aller Anwendungskomponenten komplexe Geschäftsprozesse im SAP-System abbilden. Der *SAP Business Workflow* ist darüber hinaus in der Lage, Ihre unternehmensspezifischen Besonderheiten in den Geschäftsprozessen zu berücksichtigen.

*Workitem,
Business Workplace*

Ein Workflow stellt alle Arbeitsschritte (*Workitems*) mit den dazu benötigten Daten, Dokumenten und Anlagen zur vorgegebenen Zeit in die Inbox des Business Workplace der jeweils zur Ausführung berechtigten Benutzer. Dabei können *Regeln* zur dynamischen Mitarbeiterfindung genutzt werden. Per Doppelklick werden die *Workitems* ausgeführt und mit den passenden Daten aus dem Workflow versorgt. Die Ausführung von *Workitems* führt oftmals wieder auf die bekannten Transaktionen oder Reports des SAP-Systems. Allerdings entfällt die Suche in tiefen Menüstrukturen, was insbesondere gelegentlichen Benutzern des Systems sehr entgegenkommt.

Workflow-Definition

Schließt ein Bearbeiter einen Arbeitsschritt ab, so ermittelt das Workflow-System aus der Workflow-Definition auf der Grundlage der aktuellen Daten und evtl. getroffener Entscheidungen sofort den oder die Nachfolgeschritte, besorgt die dafür notwendigen Daten, ermittelt die zuständigen Bearbeiter und stellt diesen die Arbeitsschritte in ihren Eingangskorb. Die Pflicht, im System nach vorhandenen Aufgaben zu suchen, gibt es für die Bearbeiter nicht mehr.

Terminüberschreitung

Terminüberschreitungen kann das System rechtzeitig feststellen und bei Bedarf verantwortliche Personen benachrichtigen oder vorgesehene Eskalationsverfahren einleiten. Auf diese Weise lassen sich Durchlaufzeiten oft erheblich verkürzen.

Ein besonderer Vorteil des *SAP Business Workflow* besteht darin, dass die modellierten Geschäftsprozesse mit wenig Aufwand an veränderte Situationen angepasst werden können, ohne dass dazu mit Programmierung verbundene Änderungen in der Software selbst erforderlich wären. Durch eine Versionierung von Workflow-Definitionen kann zu vorgegebenen Zeitpunkten von einem Modell eines Geschäftsprozesses auf ein anderes umgeschaltet werden.

Aufbauorganisation

Änderungen in der Aufbauorganisation können durch geeignet gewählte, personenunabhängige Zuordnungen von Schritten eines Workflows zu Organisationseinheiten, Stellen oder Planstellen sogar ohne Änderungen einer Workflow-Definition abgebildet werden.

1.1 Die Architektur des SAP Business Workflow

In diesem Kapitel werden die grundlegenden Begriffe und die wichtigsten Architekturmerkmale des Workflow-Systems im Zusammenhang vorgestellt.

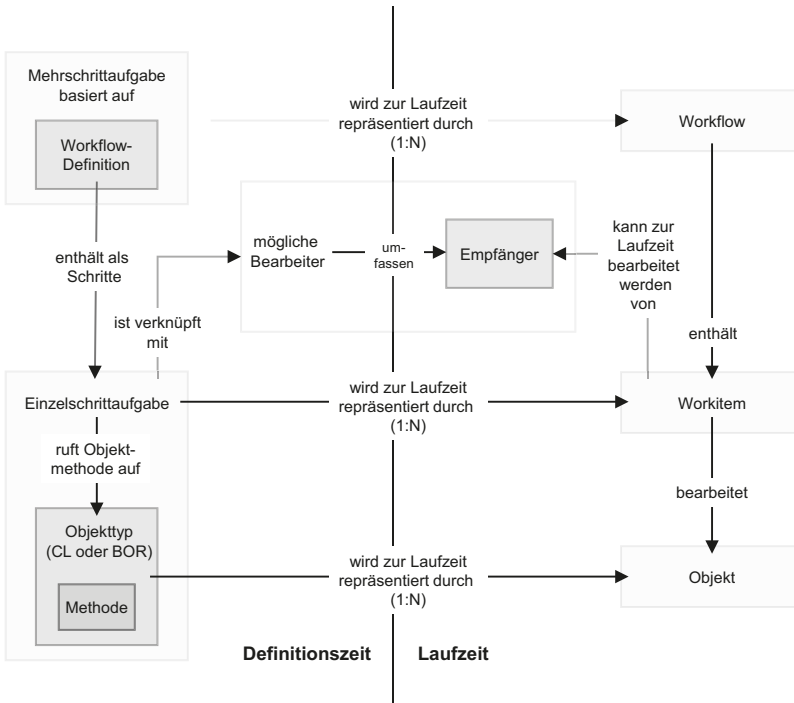


Abb. 1–1

Architektur des SAP
Business Workflows mit
allen Komponenten

Der *SAP Business Workflow* ist formal aufgebaut und trennt deutlich zwischen der *Definitionszeit* und der *Ausführungszeit* (auch *Laufzeit* genannt), wobei jeweils die »Objektypen« der Definitionszeit durch »Objekte« der Ausführungszeit repräsentiert werden.

1.1.1 Definitionszeit

Im *SAP Business Workflow* müssen zur Definitionszeit *Workflow-Definitionen*, *Einzelschrittaufgaben* und *Objektypen* festgelegt werden. Zur Ausführungszeit nimmt das System dann in den entsprechenden Laufzeitrepräsentationen, wie *Workflows*, *Workitems* und *Objekten*, Bezug auf diese Definitionen. Änderungen der jeweiligen Definition haben dann auch immer ein geändertes Laufzeitverhalten zur Folge.

Eine *Einzelschrittaufgabe* stellt das zentrale Element in der Architektur des *SAP Business Workflow* dar. In einer *Einzelschrittaufgabe* sind der *funktionale* und der *organisatorische* Aspekt einer betriebs-

Einzelschrittaufgabe

wirtschaftlichen Tätigkeit zusammengefasst. Um beiden Aspekten gerecht zu werden, ist in der Definition einer Einzelschrittaufgabe festgelegt, *was* diese Einzelschrittaufgabe leistet und *wer* für ihre Ausführung berechtigt und befugt ist.

Tab. 1-1
Zwei wesentliche Aspekte
der Einzelschrittaufgabe

Funktionaler Umfang – was? Objektmethode	Organisatorische Berechtigung – wer? Aufbauorganisation
Die Funktionalität einer Einzelschrittaufgabe wird durch genau eine <i>Objektmethode</i> ausgedrückt. Dazu sind, unabhängig von der Workflow-Funktionalität, die betriebswirtschaftlich relevanten Business-Objekte des SAP-Systems mit ihren Attributen und Methoden definiert und implementiert.	Die organisatorische Berechtigung für die Bearbeitung einer Einzelschrittaufgabe orientiert sich an der <i>Aufbauorganisation</i> des Unternehmens. Änderungen in der Unternehmensstruktur ändern somit auch Zuständigkeiten und Verantwortlichkeiten von Einzelschrittaufgaben im Workflow. Änderungen an der Workflow-Definition selbst sind nicht erforderlich.

Es gibt auch Einzelschrittaufgaben, die ausschließlich vom System im Hintergrund ohne Dialog erledigt werden (können). In diesem Fall braucht kein Benutzer etwas zu tun und die Festlegung einer organisatorischen Berechtigung ist nicht erforderlich.

Mehrschrittaufgabe =
Workflow-Definition =
Workflow-Muster

In einer *Workflow-Definition*, die immer in einer Mehrschrittaufgabe gekapselt ist, wird der gesamte betriebswirtschaftliche Prozess allgemeingültig, vollständig und unter Berücksichtigung aller Eventualitäten und Entscheidungsmöglichkeiten beschrieben. Die »Bausteine« für alle *ausführbaren* Schritte einer Workflow-Definition sind die eben beschriebenen Einzelschrittaufgaben.

Workflows werden im Unternehmen genau passend für dessen Anforderungen definiert. Dabei kann man auf Workflow-Muster, die die SAP AG ausliefert, zurückgreifen. Diese Workflow-Muster übernimmt man entweder unverändert oder verwendet sie als Kopiervorlagen und Ausgangsbasis für eigene Entwicklungen.

Schritttypen

Neben den funktionalen, auf Einzelschrittaufgaben basierenden Schritten gibt es weitere *Schritttypen*, die vom Workflow-System zur Ablaufkontrolle und Ablaufsteuerung interpretiert und ausgeführt werden.

Schritttyp Aktivität	Schritttyp Benutzerentscheidung	Weitere Schritttypen zur Verzweigung, Schleife oder Ablaufsteuerung
Schritttyp, der sich auf eine <i>Einzelschrittaufgabe</i> bezieht und von einem Benutzer (oder vom System, abhängig von der Definition der Einzelschrittaufgabe) ausgeführt wird.	Schritttyp, der sich auf eine spezielle Einzelschrittaufgabe, die sogenannte <i>Entscheidungsaufgabe</i> , bezieht und einem Benutzer verschiedene Verzweigungsmöglichkeiten anbietet.	Schritttypen, die als Modellierungselemente einer Workflow-Definition zur Verfügung stehen und die zusammen mit den Aktivitäten die Definition von umfassenden und vollständigen Workflows ermöglichen.
Die Schritte können in Folge oder in paralleler Verarbeitung angeordnet sein. In einer Workflow-Definition werden für alle möglichen Ergebnisse eines Schrittes, die zur Ausführungszeit auftreten können, die Folgeschritte modelliert.		

Tab. 1–2

Übersicht über die Schritttypen in einer Workflow-Definition

Als moderne Schnittstelle zwischen dem Workflow-System und den Anwendungen fungieren die Klassen aus ABAP Objects (CL-Objekttypen). Einen älteren objektorientierten Ansatz stellt das *Business Object Repository* mit den BOR-Objekttypen dar.

Business Object Repository

Die vorhandenen Funktionen des SAP-Systems und natürlich auch neu zu entwickelnde Funktionen sind als *Objektmethode* dieser *Objekttypen* definiert. Daten oder Informationen aus einer Anwendung können als *Attribute* dieser Objekte »erfragt« und in Workflows beispielsweise verwendet werden, um entsprechend zu verzweigen. Funktionen und Daten des SAP-Systems werden über die Objektschnittstelle für den *SAP Business Workflow* in einheitlicher Weise angesprochen, ohne die tatsächliche Implementierung der Methodenausführung und der Attributzugriffe im Detail zu kennen.

Objekttyp, Objektmethode, Objektattribut

Das Buch konzentriert sich auf die Klassenobjekttypen (CL-Objekttypen = CL-OT). Detaillierte Aussagen zum Business Object Repository finden sich im Anhang.

Man kann, auch wenn das auf den ersten Blick vielleicht naheliegender wäre, bei der Definition eines Workflows niemals direkt eine Objektmethode angeben. Man kann immer nur eine Einzelschrittaufgabe festlegen, über die dann indirekt auch die Methode bekannt ist.

Dieses Vorgehen wurde gewählt, um den Unterschied zwischen funktionalem Aspekt (Methode) und organisatorischem Aspekt (Einzelschrittaufgabe) deutlich zu machen.

Wenn neue Funktionen bereitzustellen sind, die in einem Workflow genutzt werden sollen, dann müssen entsprechende Methoden bei einem Objekttyp *definiert und implementiert* werden. In seltenen Fällen kann es auch erforderlich werden, sogar einen ganz neuen Objekttyp selbst zu definieren. In diesem Fall sind Programmierkenntnisse notwendig.

1.1.2 Workflows und Workitems zur Ausführungszeit

Workflow als Ausführung
einer Workflow-Definition

Jede Workflow-Definition kann mehrfach, auch zur selben Zeit, ausgeführt werden. Die jeweiligen Ausführungen einer Workflow-Definition (gern auch »Repräsentationen einer Workflow-Definition« genannt) werden als *Workflows* bezeichnet. Jeder Workflow bildet eine eigene, von allen anderen Workflows unabhängige Instanz.

Ein Workflow ist zur Ausführungszeit die gemeinsame »Hülle« für alle Workitems, die den Prozess bilden. Der Workflow übernimmt die Ablaufsteuerung, verwaltet die globalen, prozessspezifischen Daten und stellt das Workflow-Protokoll zur Verfügung.

Workitem als Schritt eines
Workflows

Die einzelnen Schritte innerhalb eines Workflows, die von einem Benutzer zu bearbeiten sind, repräsentiert das Workflow-System durch *Workitems* und bietet sie ihren Empfängern in deren *Business Workplace* zur Bearbeitung an.



Ein Workitem kann zwar mehrere Empfänger haben, also bei mehreren Personen in der Inbox erscheinen. Es hat aber letztlich nur einen tatsächlichen Bearbeiter.

1.1.3 Bearbeiterfindung

Ein Workflow kann dazu verwendet werden, Funktionen zu automatisieren und sie ohne Benutzerinteraktion vom System ausführen zu lassen. Die typischen Workflow-Szenarios spielen sich allerdings in aller Regel zwischen zwei oder mehreren Benutzern ab. In diesem Fall kann man zu Recht vom Workflow-System erwarten, dass es jeweils die »richtigen« *Empfänger* für die Bearbeitung eines Workitems ermittelt.

Organisatorische
Zuordnung

Ein wesentliches Element der automatischen Bearbeiterfindung wurde bereits besprochen: Das Workflow-System »weiß« für jede Einzelschrittaufgabe, wer ihre *möglichen Bearbeiter* sind, da jede Einzelschrittaufgabe mit ihrer entsprechenden organisatorischen Zuordnung definiert sein muss.

Regel

Eine nicht nur aufgaben-, sondern prozess- und objektspezifische Bearbeiterfindung wird durch die Verwendung von *Regeln* ermöglicht. Jeder Mitarbeiter kann, über seine unmittelbare organisatorische Einordnung hinausgehend, eine oder mehrere Rollen oder Berechtigungen innehaben, die jeweils besondere Zuständigkeiten, Befähigungen oder Eigenschaften beschreiben.

Zum Zweck einer dynamischen, erst zur Ausführungszeit festgelegten Bearbeiterzuordnung kann das Workflow-System angewiesen werden, die Empfänger eines Workitems über ihre Regel, die sie in Bezug auf einen bestimmten Geschäftsprozess oder in Bezug auf ein

bestimmtes Objekt (z.B. »Sachbearbeiter für Kunden M-P«, »Prüfer von Eingangsrechnungen«) besitzen, zu finden.

1.1.4 Workflow als Reaktion auf Anwendungssignale

Workflows können aus der Entwicklungsumgebung im Dialog und ohne Anwendungsbezug gestartet werden. Workflows können auch direkt aus einer beliebigen Anwendung heraus aufgerufen werden. Beide Möglichkeiten sind im Hinblick auf individuelle Erweiterungsmöglichkeiten wenig flexibel, haben aber ihre Daseinsberechtigung.

Ein (externes) Ereignis dagegen, mit dem eine beliebige Anwendungskomponente eine erfolgte Zustandsänderung oder ein erreichtes Ergebnis systemweit mitteilt, bietet eine flexible »Andockstelle« für die Ausführung von Workflow-Definitionen.

Ereignis

Wenn das Ereignis als *auslösendes Ereignis* einer Workflow-Definition eingetragen ist, wird diese Workflow-Definition ausgeführt (als Workflow instanziiert), sobald das Ereignis eintritt. Wenn das Ereignis nicht als auslösendes Ereignis irgendwo vorgesehen ist, passiert auch nichts, wenn es eintritt.

Jedes Ereignis gibt Informationen über das ereigniserzeugende Objekt an den Workflow weiter, den es startet.

Ereignisse überbrücken damit die »Grenze« zwischen der vorhandenen Anwendungsfunktionalität und neu entwickelten, eigenen Workflows. Um die Workflow-Funktionalität nutzen zu können, kann nahezu jede Anwendung flexibel, spezifisch und einfach(!) um die Erzeugung der benötigten Ereignisse erweitert werden. Im Kapitel 6 wird das eingehend besprochen.

1.1.5 Vorkonfigurierte Workflow-Definitionen

Für die schnelle Einführung liefert die SAP AG für viele betriebswirtschaftliche Szenarios bereits *vorkonfigurierte Workflow-Definitionen*, die sogenannten *Workflow-Muster*, aus. Sie können diese Workflow-Muster in vielen Situationen mit minimalem Anpassungsaufwand für Ihre betrieblichen Prozesse verwenden. Die Workflow-Muster verkürzen die Einarbeitungszeit ganz wesentlich und sind auf die jeweiligen Anwendungsfunktionen optimal abgestimmt.

Workflow-Muster

Workflow-Muster können Sie für eigene Entwicklungen als Vorlage verwenden. Kopieren Sie dazu das Workflow-Muster unter einem neuen Namen und verändern Sie es dann.

Bei der Suche nach vorhandenen Workflow-Mustern hilft Ihnen der Bestandsnachweis von Workflow-Mustern, den Sie als Teil der

Aufgabenanalyse in der Entwicklungsumgebung für den *SAP Business Workflow* finden.

1.1.6 Der typische Workflow-Benutzer

Den »typischen« Workflow-Benutzer gibt es eigentlich gar nicht. Vielmehr lassen sich folgende Benutzertypen voneinander unterscheiden, die sich mit unterschiedlichen Kenntnissen und Fähigkeiten einbringen:

Sachbearbeiter

Der *Sachbearbeiter* ist der eigentliche Nutzer und Nutznießer eines ablauffähigen, produktiven Workflows. Er verwendet den Business Workplace, über den ihm seine tägliche Arbeit gebracht wird. Er liest die Beschreibung und die Anlagen der ihm zugeteilten Workitems und führt sie aus. Zur optimalen Gestaltung seines Arbeitsumfeldes konfiguriert der Sachbearbeiter seinen Eingangskorb und nutzt die bereitgestellten Funktionen, wie *Weiterleiten*, *Anfrage senden*, *Zurücklegen* usw. Der Sachbearbeiter findet die für ihn wichtigen Informationen im Kapitel 12.

Prozessmodellierer

Der *Prozessmodellierer* hat seine Arbeit im Großen und Ganzen getan, bevor die ersten Workflows produktiv gehen. Er weiß, wo und wie die Workflow-Funktionalität genutzt werden soll, und kennt die betriebswirtschaftlichen Prozesse sehr genau, weil er sie entsprechend optimiert hat. Er analysiert die von der SAP AG ausgelieferten Workflow-Muster im Hinblick auf ihre Verwendbarkeit und nutzt ggf. den Workflow-Editor, um eigene Workflows zu definieren oder vorhandene Workflow-Definitionen anzupassen. Fehlende Einzelschrittaufgaben legt er unter Verwendung vorhandener Objektmethoden an. Falls der Workflow auf Änderungen in einer Anwendung reagieren soll, sorgt der Prozessmodellierer für die Erzeugung der entsprechenden Ereignisse. Der Prozessmodellierer findet die für ihn wichtigen Informationen in den Kapiteln 8, 9, 10 und 11.

Programmierer

Der *Programmierer* arbeitet eng mit dem Prozessmodellierer zusammen. Er erweitert, wenn nötig, die von der SAP AG ausgelieferten Objekttypdefinitionen (BOR-Objekttypen oder Workflow-fähige Klassen) um weitere Methoden und Attribute. Er implementiert im Pro-

gramm des Objekttyps, wie eine Methode ausgeführt und wie der Wert eines Attributes gelesen oder berechnet wird. Dazu muss er die Funktionen und Tabellen »seiner« Anwendung kennen.

Verantwortlicher für Aufbauorganisation

Der *Verantwortliche für Aufbauorganisationen* erstellt eine Aufbauorganisation, die zumindest die Mitarbeiter umfasst, die als Sachbearbeiter (s.o.) mit dem Workflow in Berührung kommen. Er nimmt in Abstimmung mit dem Prozessmodellierer die entsprechenden Zuordnungen zu den Einzelschrittaufgaben vor. Der *Verantwortliche für Aufbauorganisationen* verwendet eine abstrakte Beschreibungsweise auf der Grundlage von Stellen, Planstellen und Organisationseinheiten, die es ermöglicht, dass Änderungen an der Aufbauorganisation keine Änderungen der Workflow- oder Aufgabendefinitionen nach sich ziehen.

Workflow-Administrator

Der *Workflow-Administrator* führt das Customizing und die erforderlichen Transporte durch. Er ist der Adressat für Benachrichtigungen, die das Workflow-System bei fehlerhaften Workflows verschickt. Er kennt die verschiedenen Protokollierungs- und Reporting-Möglichkeiten und kann Fehler beheben.

Während der zuerst genannte Sachbearbeiter als Benutzertyp relativ klar abgegrenzt ist, sind die Grenzen zwischen den letztgenannten Benutzertypen fließend. In der Regel müssen sich einzelne Mitarbeiter in mehr als einem Bereich auskennen und arbeiten ohnehin in einem Team zusammen.

1.2 Ein typisches Workflow-Szenario

Abläufe, mit denen ein Workflow-System optimal zurechtkommt, sind Genehmigungs- und Freigabeverfahren. Hier sind alle Aspekte eines typischen Workflows vertreten:

Bei einem Genehmigungs- oder Freigabeverfahren geht es üblicherweise um ein Objekt, wie z.B. um einen Urlaubsantrag, eine Materialbestellung oder eine Rechnung. Dieses Objekt wird mit Methoden wie *Anlegen*, *Freigeben* oder *Überarbeiten* bearbeitet.

Genehmigungs- bzw. Freigabeverfahren sind ein Wechselspiel zwischen zwei Mitarbeitern, die eine besondere, in der Regel durch die Aufbauorganisation festgelegte Beziehung zueinander haben. Der Erfasser des Antrages oder des Dokumentes kann ein beliebiger Mitar-

*Genehmigungs- und
Freigabeverfahren*

beiter sein; der Mitarbeiter mit der Berechtigung zur Genehmigung oder Freigabe ist dann oft der Vorgesetzte (Regel!) des Antragstellers, den das Workflow-System selbstständig aus der Aufbauorganisation ermitteln muss.

Die Ablauflogik der Workflow-Definition muss u. a. folgende Fälle abdecken:

- Der Antrag oder das Dokument wird sofort genehmigt oder freigegeben. Der Antragsteller wird benachrichtigt.
- Der Antrag oder das Dokument wird vom Genehmigenden mit Rückfrage oder Kommentar zur Überarbeitung an den Antragsteller zurückgesandt. Nach der Überarbeitung beginnt das Verfahren erneut.
- Der Erfasser oder der Genehmigende sagen das Verfahren endgültig ab.
- Der Genehmigende verzögert (oder vergisst) die Bearbeitung.
- Der Genehmigende ist verhindert.

1.3 Dokumentation und Tutorial

In der Dokumentation zum *SAP Business Workflow* finden Sie zu allen Themen dieses Buches vertiefende Erklärungen, Vorgehensweisen und Beispiele. Die Dokumentation ist Bestandteil der R/3-Bibliothek, die mit Ihrem System ausgeliefert wurde. Sie finden die Dokumentation in der Anwendungskomponente *Basis – Business Management*, weitere Hinweise dazu im Kapitel 2.



Als Teil der Dokumentation erklärt Ihnen ein Tutorial schrittweise die Definition und Ausführung eines Workflows am Beispiel der Genehmigung eines Urlaubsantrags. Dieses Tutorial sollten Sie unbedingt durcharbeiten.

1.4 Business Workplace

Mitarbeiter, die in ihrer täglichen Arbeit mit Geschäftsprozessen zu tun haben, die durch Workflows gesteuert werden, nutzen den *Business Workplace* als zentrales Arbeitsumfeld. Von hier aus haben sie Zugriff auf ihren persönlichen Arbeitsvorrat (ihre *Worklist*) mit den *Workitems*, die sie aufgrund ihrer derzeitigen organisatorischen Einordnung und Zuständigkeit zu erledigen haben. Darüber hinaus enthält der Business Workplace auch ihre SAP-Mails.

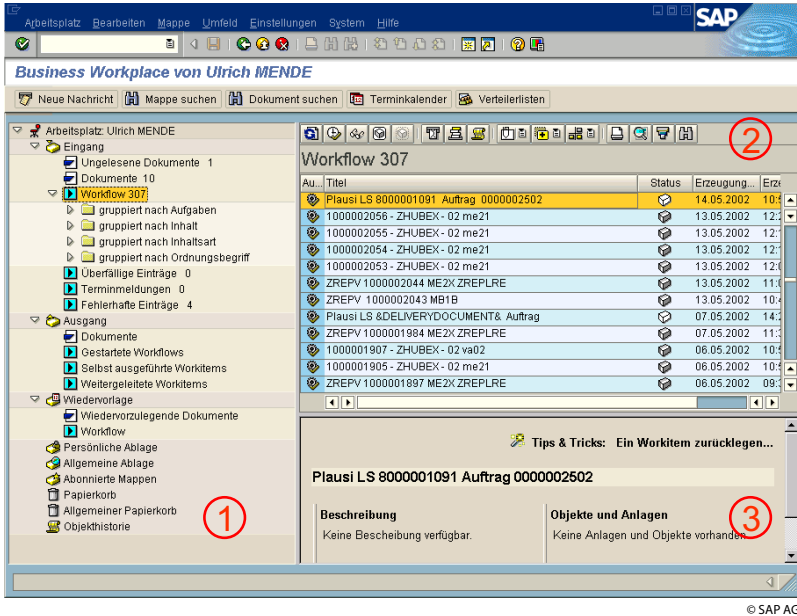


Abb. 1-2

Business Workplace

1.4.1 Mails im Business Workplace

Über ihre *Mails* sind Mitarbeiter an innerbetrieblichen Kommunikations- und Entscheidungsprozessen beteiligt. Unter Mails versteht man in diesem Kontext ganz allgemein die Dokumente der SAP Office-Eingangsliste. Der Begriff ist weit gefasst und meint einen einfachen SAPScript-Text im RAW-Format ebenso wie eine Microsoft PowerPoint Präsentation oder ein mit Microsoft Word erstelltes DOC-File. Auch Kombinationen sind denkbar (z.B. SAPScript-Text mit PowerPoint-Präsentation als Anlage).

Dokumente

Die Erfahrung zeigt jedoch, dass die SAP-Mails praktisch nicht genutzt werden, sondern dass Mails ausschließlich über entsprechende Mailsysteme (Outlook, Lotus Notes) versendet und empfangen werden. Der Grund dafür ist sicher die einfache Erreichbarkeit über das Internet.

1.4.2 Workitems mit Benutzerdialog in der Worklist

Die *Workitems*, die Mitarbeiter in ihrem Business Workplace unter der Sammelbezeichnung Workflow sehen, sind die *Schritte* aus beliebigen *Workflows*, die zu ihrer Bearbeitung den *Dialog* mit ihnen erfordern. Jedes Workitem repräsentiert eine bestimmte, allgemein formulierte *Aufgabenstellung* (Beispiel: Einzelschrittaufgabe *Bestellanforderung*).

Worklist

genehmigen). Diese allgemeine Aufgabenstellung wird beim Workitem konkretisiert durch das *Objekt*, das mit dem Workitem bearbeitet wird, und durch die *Empfänger*, die das Workitem ausführen dürfen (Beispiel: Workitem *Bestellanforderung 98/123 genehmigen*, Empfänger *Mitarbeiter mit Freigabecode 01*).

Die Workitems im Business Workplace bilden die persönliche *Worklist* eines Mitarbeiters, die das Workflow-System für ihn in Abhängigkeit von seiner organisatorischen Zuordnung und von seinem aktuellen betrieblichen Zuständigkeitsprofil beim Aufruf des Business Workplace aufbaut.

Der Mitarbeiter führt dann ein Workitem aus, um damit das zugeordnete Objekt zu bearbeiten und so die vorgesehene Einzelschrittaufgabe zu erledigen.

Eine detaillierte Beschreibung der Funktionen des Business Workplace findet sich im Kapitel 12.

2 Workflow-Entwicklung – Tools

2.1 Workflow aus Entwicklersicht

2.1.1 Workflow als Framework

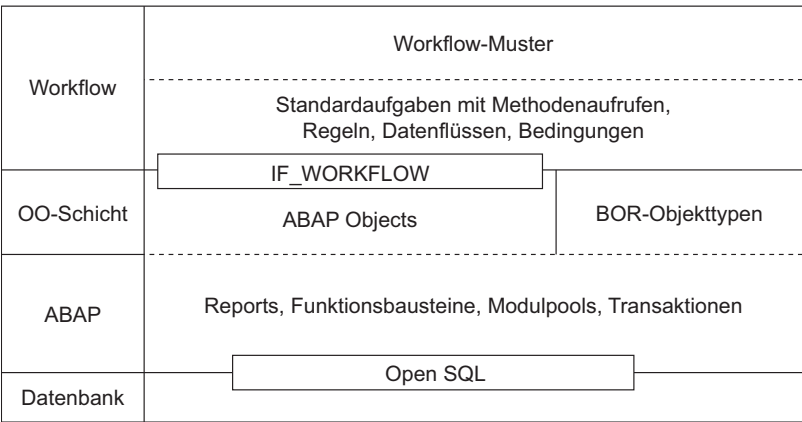
Aus Sicht eines Workflow-Entwicklers stellt das Workflow-System ein komplexes, generisches Framework dar, in dem er seine konkrete Anwendung, also seine zu erstellenden Workflow-Muster mit ihren Komponenten, implementieren muss. Dies wird insbesondere bei der klassenbasierten Entwicklung deutlich, in der entsprechende Methoden (Vererbung, Interfaces) zum Einsatz kommen.

2.1.2 Schichtenmodell einer Workflow-Anwendung

Eine Workflow-Anwendung ist eine komplexe, schichtenweise aufgebaute Softwarekonstruktion. Die untere Schicht ist die ABAP-DB-Schicht, in der die Anwendungen in ABAP codiert werden und ihre Daten persistent in der Datenbank speichern. Die oberste Schicht ist die Workflow-Schicht, in der die komplexen Prozesse über Aufgaben, Regeln zur Bearbeiterfindung und Datenflüsse abgebildet werden. Eine besondere Rolle – die auch Hauptgegenstand dieses Buches ist – spielt die mittlere Schicht (OO-Schicht). Hier handelt es sich um eine objekt-orientierte Verschalung der ABAP-DB-Schicht, die den Workflow letztlich unabhängig von den konkreten Anwendungen macht und es ihm als generisches Framework gestattet, die unterschiedlichsten Business-Objekte (Eingangsrechnungen im FI, Wechsel- und Sperrbelege im IS-U, Materialstämme im MM, Serviceaufträge im PM¹) völlig gleichartig zu behandeln.

1. SAP-Module: FI = Finanzwesen, MM = Materialwirtschaft, PM = Instandhaltung, IS-U = Industry Solution Utilities

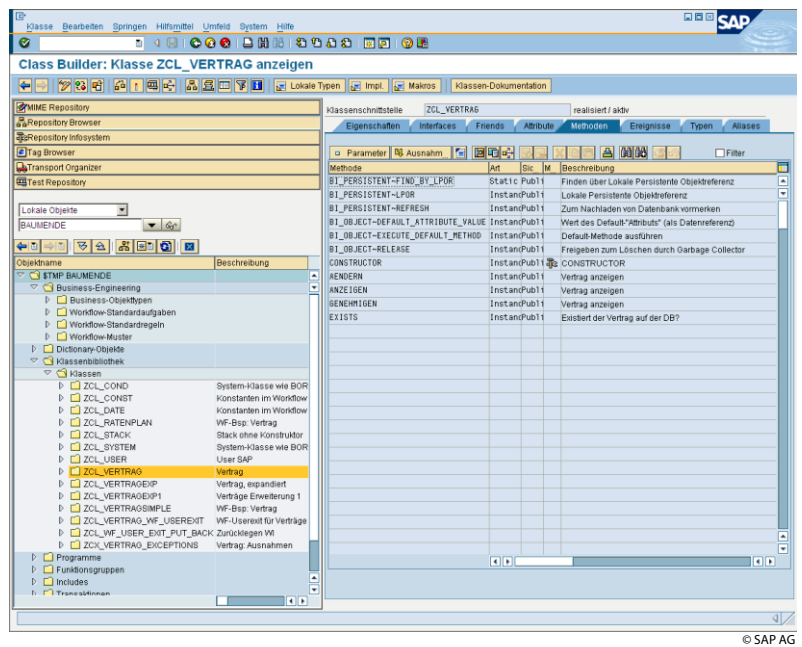
Abb. 2–1
Schichtenmodell einer
Workflow-Anwendung



2.1.3 Paketorientierte Workflow-Entwicklung in der SE80

Die meisten Objekte der Workflow-Entwicklung sind vollständig in den Object Navigator (SE80) integriert. Es ist unbedingt zu empfehlen, dies auch zu nutzen. So hat man immer den Bezug zu dem Paket, in dem die Workflow-Anwendung entwickelt wird, und findet auch in der Klassenbibliothek die im Workflow benutzten Klassen, die Funktionsbausteine der Klassenmethoden, die Datenstrukturen der Parameterdefinitionen usw.

Abb. 2–2
Alle Entwicklungsobjekte
einer Workflow-
Anwendung in der SE80



Lediglich für Aufgabengruppen (Taskgruppen – TG) muss man den Zugang über SWLD benutzen (PFTC_*), obwohl Aufgabengruppen ebenfalls eindeutig einem Paket zugeordnet sind. Man erkennt, dass die Business-Objekttypen (BOR) direkt zum Business Engineering gehören, während die Klassen (ABAP Objects) viel breiter genutzt werden.

*Aufgabengruppen nicht
über SE80*

2.1.4 Das Workflow-Bereichsmenü SWLD

Neben den paketerorientierten Entwicklungstools gibt es eine große Anzahl von übergreifenden Tools, die alle im Bereichsmenü SWLD zusammengefasst sind.

Ein Workflow-Entwickler kann diesen Menünamen (ohne Vorsatz »/n« für neue Transaktion) in das Transaktionsfeld eingeben und erhält unter SAP-Menü die folgenden sechs Hauptpunkte von SWLD:

- Definitionswerkzeuge
- Laufzeitwerkzeuge
- Reporting
- Hilfsmittel
- Umfeld
- Administration

Bereichsmenü SWLD

Unter diesen Punkten werden ca. 120 Transaktionen angeboten, von denen aber erfahrungsgemäß nur wenige häufig benutzt werden. Viele Workflow-Entwickler ziehen sich die wichtigsten Transaktionen aus SWLD in die Favoritenliste, die zusätzlich noch die wichtigsten Transaktionen zum Zugriff auf das Dictionary, die ABAP-Workbench und den Class Builder enthält. So hat man einen zentralen Einstieg in die unterschiedlichsten Entwicklungswerkzeuge, die für eine komplexe Workflow-Anwendung erforderlich sind.

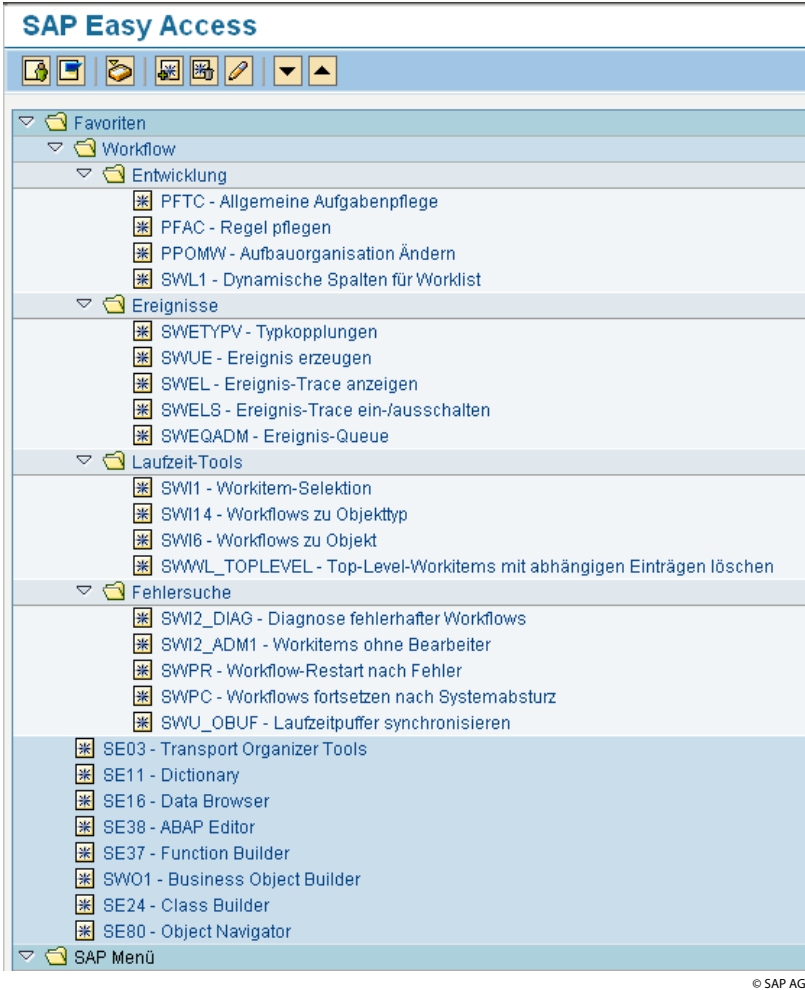
*Empfehlenswert:
eigenes Favoritenmenü
für Workflow*

Die übrigen Menüpunkte erreicht man durch Aufklappen des Menübaums SWLD. Lästig werden von Entwicklern oft die getrennten Transaktionsnamen für Anlegen, Ändern, Löschen von Aufgaben und Standardregeln empfunden. Anstelle der Einzeltransaktionen können in die Favoritenliste die übergeordneten Transaktionen PFTC und PFAC eingefügt werden, die im Baum SWLD nicht angeboten werden.

*PFTC = allgemeine
Aufgabenpflege
PFAC = allgemeine
Regelpflege*

Manche der Transaktionen werden mehrfach angeboten, z.B. die für Ereignisse. In den folgenden Kapiteln werden viele der oben aufgeführten Transaktionen besprochen. Dazu werden am Anfang jeweils Transaktion (TA) und Pfad noch einmal genannt.

Abb. 2–3
 Favoriten mit speziellem
 Ordner für Workflow



2.2 Workflow-Beispiellösungen der SAP

2.2.1 Tutorials

In der SAP Library werden vier Tutorials angeboten, die insbesondere Einsteigern das schrittweise Erlernen in Form einer Bedienungsanleitung bieten.

- Workflow-Modellierung
 20 Lektionen zum Anlegen von Aufgaben und Aufbau eines Workflows, inklusive Terminüberwachung, Mailversand
- Pflege der Aufbauorganisation
 3 Lektionen zur Pflege der Aufbauorganisation

- Ereigniserzeugung bei Statusänderungen
8 Lektionen zur Ereigniserzeugung
- Workflow-Programmierung
13 Lektionen zur Programmierung eines neuen BOR-Objektyps

2.2.2 Workflow-Anwendung »Urlaubsantrag«

Dies ist eine fertige Anwendung zur Erfassung und Workflow-gestützten Bearbeitung von Urlaubsanträgen. Sie wird in den SAP-Kursen häufig verwendet. Man kann sie als BOR-Anwendung, als Klassenanwendung oder als Formularanwendung finden.

In vielen Fällen dient diese Anwendung Beratern zu einer ersten Demonstration des Workflow-Systems – vermutlich weil von Urlaub jeder etwas versteht. Alle Entwicklungsobjekte findet man in den Paketen SWH und SWX. Um den einfachsten Workflow AF_process (WS30000015) lauffähig zu machen, muss man folgende Schritte ausführen:

*Workflow-Muster
Urlaubsantrag =
WS30000015*

- Ereigniskopplung aktivieren.
- Org-Einheit mit einer Leiterplanstelle und wenigstens einer zweiten Planstelle anlegen und beide Planstellen besetzen.
- Urlaubsanträge mit Transaktion SWLD → *Demo* → *SWXF erfassen*.

Dann bekommt der Inhaber der Leiterplanstelle die Urlaubsanträge der anderen Mitarbeiter seiner Org-Einheit zur Genehmigung vorgelegt.

2.2.3 Workflows der Aufgabengruppe WFUNIT

In der Aufgabengruppe WFUNIT (TG56400001, öffnen mit PFTC_DIS) befinden sich ca. 80 kleinere Workflow-Muster, die jeweils einen Sachverhalt demonstrieren. Diese Workflows sind ausgesprochen gut geeignet, wenn man gern am Beispiel lernt oder wenn man bei der Anwendung einer konkreten Designmethode Probleme hat.

*80 Workflow-Muster
in TG WFUNIT*

Speziell neuere Designtechniken und Klassentechniken werden behandelt.

Abb. 2-4
Workflow-Muster der
Aufgabengruppe WFUNIT

Aufgabengruppe56400001WFUNIT

BezeichnungUnittest für Workflow

EntwickungsklasseSWF_TST

AnwendungskomponenteE

Beschreibung

Standardaufgaben

Workflow-Muster

Workflow-Muster

Objekt-ID	Objektkürzel	Objektbezeichnung
56400008	WFUNIT_0001	Synchrone Dialogmethoden (ABAP 00)
56400010	WFUNIT_0002	Block
56400011	WFUNIT_0003	Dialogmethode ohne Schlüssel (BOR)
56400013	WFUNIT_0004	Warteschritt obsolet setzen
56400015	WFUNIT_0005	Case
56400017	WFUNIT_0006	Switch
56400018	WFUNIT_0007	Statisch Parallel (Transiente Schritte)
56400022	WFUNIT_0008	Statisch Parallel (Hintergrundschr
56400023	WFUNIT_0009	Dynamische Parallelität (Hintergrund)
56400024	WFUNIT_0010	Schleife (Until)
56400025	WFUNIT_0011	Containeroperation
56400026	WFUNIT_0012	Workflow-Exceptionbehandlung
56400056	WFUNIT_0013	Lokaler Workflow

© SAP AG

2.2.4 Workflows der Aufgabengruppe WF_Verify

In der Aufgabengruppe WF_Verify (TG70000020, öffnen mit PFTC_DIS) befinden sich ca. 60 Workflows zu diversen Standardaufgabenstellungen der Workflow-Entwicklung. Auch diese Beispiele sind zu empfehlen.

Abb. 2-5
Workflow-Muster der
Aufgabengruppe
WF_Verify

Aufgabengruppe70000020WF_Verify

BezeichnungVerifikations- und Test-Workflows

EntwickungsklasseSWH

AnwendungskomponenteE

Beschreibung

Standardaufgaben

Workflow-Muster

Workflow-Muster

Objekt-ID	Objektkürzel	Objektbezeichnung
45301303	WF_Verify032	Mail senden
45301442	WF_Verify033	Datenfluss-Verifikation (Assign/Append)
48900047	WF_Verify031	Restart: Fehler in der Ausführung
48900074	WF_Verify035	Bearbeitung eines Workitem ablehnen
48900086	WF_Verify100	Web-Aktivität: Daten ohne Umschlag
48900090	WF_Verify101	Web-Aktivität: SOAP
48900093	WF_Verify102	Web-Aktivität: Wf-XML: Warten auf Rückm
48900099	WF_Verify103	Web-Aktivität: Wf-XML: erweiterte Komm.
48900109	WF_Verify038	Endebestätigung am Dialogschritt
56400159	WF_Verify048	Abwesenheitsmitteilung mit ABAP-Klasse
56400162	WF_Verify049	Massive Parallelität
56400167	WF_Verify050	Dynamische Sequenz
70000096	ADHOC_NOTE	Merker auf einen Ad-hoc-Workflow

© SAP AG

2.3 SAP Service Marketplace

TA: (früher OSS1)

URL: <http://service.sap.com>

Ohne ständigen Zugriff auf das zentrale Serviceportal der SAP, den »SAP Service Marketplace«, insbesondere auch auf dessen passwortgeschützte Bereiche, sind weder Entwicklung noch Betrieb großer Workflow-Anwendungen denkbar.

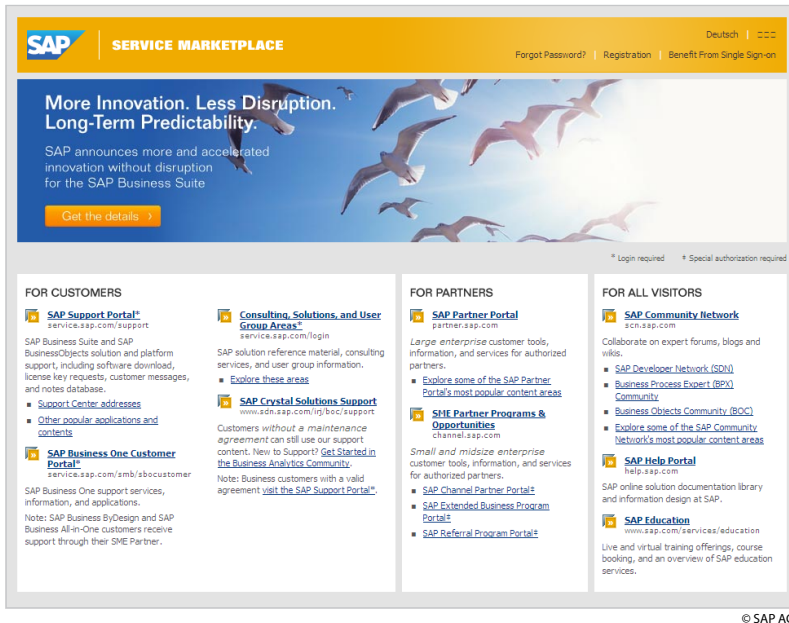


Abb. 2–6

Startseite des SAP Service Marketplace

Hier findet man die drei Bereiche FOR CUSTOMERS, FOR PARTNERS und FOR ALL VISITORS. Für die ersten beiden benötigt man einen S-User (früher OSS-User) zur Anmeldung.

2.3.1 SAP Support Portal mit Hinweissystem

URL: <http://service.sap.com/support>

Das ist der wichtigste Bereich für SAP-Anwender. Hier finden sich unter »Help & Support« die Möglichkeit zur Angabe von Fehlermeldungen (»Report a Product Error«) sowie die Hinweissuche (»Search for SAP Notes«).

Wichtiger als eigene Fehlermeldungen sind meist solche, die von anderen bereits an SAP gestellt wurden, sowie die zugehörigen Antworten der SAP. Beides zusammen wird unter »Notes« geführt. Man

Zugriff auf das SAP-Hinweissystem unbedingt erforderlich

kann in 95% der Fälle davon ausgehen, dass man nicht der erste Anwender mit einem bestimmten Problem ist. Es ist daher immer zu empfehlen, bei vermeintlichen oder tatsächlichen Fehlern zunächst in den Notes zu suchen.

In der Suchmaske füllt man zweckmäßigerweise das Feld »Application Area« mit BC-BMT-WFM für Business Workflow. Das allein ergibt jedoch über 3000 Hinweise, sodass weitere Abgrenzungen notwendig sind.

Abb. 2-7
Suchmaske für
Hinweissystem

SAP NOTES SEARCH

What's new

SAP BusinessObjects Notes are now SAP Knowledge Base Articles.
These SAP Knowledge Base Articles cover both SAP and SAP BusinessObjects products. SAP Knowledge Base Articles are marked with an asterisk (*) in your search results. [Find out more.](#)

Number

Search options

Used Template

no template used

Load Template

Language

☐ German ☒ English ☐ Japanese

Search Term

Search

Search Method

All Terms (AND)

Search Range

All

Search behavior

☒ Linguistic search ☐ Exact search

Application Area

BC-BMT-WFM

Select

Restrictions

No Restriction

Select

Additional Criteria

User defined selection

Select

Search result

Results Per Page

View / Sorting

Configure

Search

Reset

Save as Template

Help

© SAP AG

Application area	Description
AC	Accounting General
AIE	Auto-ID Enterprise
AP	Application Platform
BC	Basis Components
BC-ABA	ABAP Runtime Environment
BC-AC	Adaptive Computing - Please use sub-components
BC-BMT	Business Management
BC-BMT-BPM	Business Process Management
BC-BMT-BRM	Business Rules Management
BC-BMT-BTM	Please use Subcomponents
BC-BMT-CPL	Common Process Layer (internal use only)
BC-BMT-OM	Organizational Management
BC-BMT-PAF	Process Agent Framework
BC-BMT-WFM	Business Workflow
BC-BMT-WFM-BND	Binding & Expressions
BC-BMT-WFM-DEF	Design-time
BC-BMT-WFM-EVT	EventManager
BC-BMT-WFM-MON	Monitoring & Reporting
BC-BMT-WFM-NOT	WebFlow Notifications
BC-BMT-WFM-RUN	Runtime
BC-BMT-WFM-WEB	Web Integration
BC-BMT-WFM-WLC	Worklist Client
BC-BMT-WMD	Modeler for Workflow
BC-BNM	Business Network Management

© SAP AG

Abb. 2-8

Application Area in der Suchmaske des Hinweissystems

Über *Additional Search Criteria* kann man die Suche weiter einschränken. So findet man unter Category = Consulting oft zusammenfassende Erläuterungen von SAP.

Additional Search Criteria

Released On: From: To:

Release Status:

Priority:

Category:

Release-Independent: ☒ No Restriction ☐ only Release-Independent Notes ☐ Exclude Release-Independent Notes

© SAP AG

Abb. 2-9

Zusätzliche Suchkriterien bei der Hinweissuche

Letztlich kann man nach unterschiedlichen Stichwörtern suchen, die sowohl UND- als auch ODER-verknüpft sein können. Die Suche nach Stichwörtern sollte immer der letzte Weg sein, da ja nicht klar ist, welche Stichwörter SAP für einen Sachverhalt tatsächlich verwendet. Das Stichwort »Sammelhinweis« liefert oft gute Dienste, wenn man zu einer Sache einen Überblick über bestehende Hinweise bekommen möchte.

Abb. 2–10
Suchergebnis des
Hinweissystems

14 SAP Notes found Page 1 of 2

Ranking	Application Area	Number	Short text	Released On
<input type="checkbox"/> 1. 0.260	BC-BMT-WFM	160238	Generische Objektservices für ABAP OO	29.11.2011
<input type="checkbox"/> 2. 0.380	BC-BMT-WFM-WLC	1643178	SBWP, 'Gruppierung nach Inhaltstyp': kryptische Unternoten	21.10.2011
<input type="checkbox"/> 3. 0.370	BC-BMT-WFM	1642128	SWDM: Verwendungsnachweis für BOR- und Klassenmethoden	14.10.2011
<input type="checkbox"/> 4. 0.370	BC-BMT-WFM	1642129	SWDM: Klassenmethoden werden nicht angezeigt	14.10.2011
<input type="checkbox"/> 5. 0.290	BC-BMT-WFM-RUN	1636049	Fehlerhandling der Klasse CL_SWF_RUN_HANDLER_CANCEL_WF	07.10.2011
<input type="checkbox"/> 6. 0.380	BC-BMT-WFM-WLC	1175579	Schlechte Performance SBWP	06.10.2011
<input type="checkbox"/> 7. 0.250	BC-BMT-WFM	1633020	Erweiterungen der Laufzeit für Unittest	22.09.2011
<input type="checkbox"/> 8. 0.290	BC-BMT-WFM	1631600	Zugriff auf Workflow Systemkonfiguration	19.09.2011
<input type="checkbox"/> 9. 0.250	BC-BMT-WFM	1624274	Legen aus dem Archiv scheitert an Vererbung	24.08.2011
<input type="checkbox"/> 10. 0.360	BC-BMT-WFM-RUN	1624207	Vererbungshierarchie in CL_SWF_FORMABSENZ	24.08.2011

Marked SAP Notes... [put to Download Basket] go

© SAP AG

2.3.2 SDN-Community

SDN-Teilnahme wertvoll
für Entwickler

Im SDN (SAP Developer Network) diskutieren Entwickler weltweit in unzähligen Foren konkrete Entwicklungsprobleme. Die Foren sind in Kategorien unterteilt. Für die klassenbasierte Workflow-Entwicklung sind folgende Foren besonders wichtig (Forumkategorie → Forum):

- ABAP Development → ABAP Objects
- SAP NetWeaver → SAP Business Workflow

Abb. 2–11
Suche im SDN-Forum
»SAP Business Workflow«

SAP Community Network Forums » SAP NetWeaver » SAP Business Workflow

Search Forum: Go

Forum: SAP Business Workflow

SAP Business Workflow enables companies to model, configure, execute and monitor business scenarios and processes. To adapt them continuously to new strategies. This forum is for all technical discussions about the following tools within SAP NetWeaver: SAP Business Workflow, Universal workflow Collaboration tasks, Home Page Wiki! FAQ Generic discussions about modeling and modeling methodologies can be found in the Business Process Modeling and Methodology forum.

Welcome, Guest

Want to post to the Expert Forums?

Forum Announcements

Important: SCN Migration Announcement (Dec 14, 2011 12:14 PM)
The New SCN launch now planned for early 2012. To learn more please see this blog.
RSS

Top 3 Contributors, last 30 days: 1. Rick Bakker (150); 2. Karri Kemppt (98); 3. Pavan Bhamidipati (84)

Post New Thread Back to Forum List

Messages: 93,752 - Threads: 17,123 - Filter: [All Threads] - Pages: 1,142 - [1 2 3 4 5] Next

Thread	Author	Views	Replies	Last Post
Welcome and Rules of Engagement	comellil	9,462	0	Sep 24, 2008 1:36 PM Last Post By: comellil
Before you ask (there are many workflow answers)	Alan Rickayzen	21,384	0	Mar 2, 2006 6:24 PM Last Post By: Alan Rickayzen
starting task depending on the Date	abdul gaffar	9	1	Jan 10, 2012 9:08 AM Last Post By: nabheemrashed00
How to forward Work Item from SAP Inbox	kjalvardhan	30	6	Jan 10, 2012 7:58 AM Last Post By: vikram singh

Discover the benefits of virtualizing SAP—
• Reduced OPEX for lower TCO
• High scalability AND availability
• IT flexibility for better business decisions
Learn how

hp vmware

© SAP AG

Hier kann man sich in vielen Fällen Hilfe holen. Vielfach werden auch nachnutzbare Codestücke angeboten oder Funktionsbausteine (FBS) und Reports benannt.

2.3.3 SAP Help online und offline

URL: *http://help.sap.com* → SAP NetWeaver → SAP NetWeaver Plattform → SAP Library → SAP NetWeaver 7.0 (Deutsch) → Schlüsselbereiche von SAP NetWeaver → Schlüsselbereiche der Process Integration → Business Process Management → SAP Business Workflow

Über den SAP Service Marketplace bzw. das dort verlinkte SAP Help Portal ist die gesamte SAP-Hilfe online einsehbar, vielfach auch in Deutsch. Die Workflow-Hilfe ist allerdings ziemlich versteckt (s.o.).

Um die Hilfe auch offline verfügbar zu machen, kann man z.B. für den Workflow im Internet unter BCBMTWFMSTART(.pdf) suchen und sich die betreffenden PDF-Files herunterladen. Eine relativ aktuelle Variante findet man z.B. unter *www.ziddu.com*.

3 ABAP Objects – Grundlagen

3.1 Objektorientierung als Abbild der realen Welt

In der realen technischen Welt werden detaillierte Modelle (= Baupläne, Spezifikationen) entwickelt, auf deren Grundlage später hohe Stückzahlen gleichartiger technischer Geräte (Autos, Handys, Waschmaschinen, Fernseher usw.) produziert werden. Diese Baupläne beschreiben genauestens alle verwendeten Bauteile und deren Funktionen. Dabei wird deutlich unterschieden zwischen den nach außen sichtbaren (nutzbaren) Eigenschaften und Funktionen der Geräte und deren innerer Realisierung. Letztere wird oft regelrecht vor dem Nutzer der Geräte verborgen.

Die objektorientierte Programmierung greift diese Gedanken aus der realen Welt auf, dabei geht die Analogie weit, wie der folgende (nicht völlig ernst zu nehmende) Vergleich zwischen Handy-Bau und ABAP OO zeigt:

Begriff aus dem Handy-Bau	Entsprechung in der Objektorientierung
Handymodell, z.B. iPhone 4S	Klasse
Folge aufeinander aufbauender Modelle iPhone 3.0 → iPhone 4.0 → iPhone 4S	Vererbungshierarchie zwischen Klassen
Vorgängermodell/Nachfolgermodell	Superklasse/Subklasse
Nachfolgermodell übernimmt Funktionen des Vorgängermodells, realisiert sie aber anders	Überdefinieren von Methoden der Superklasse in der Subklasse
Trennung von Anwenderfunktionen/Einstellungen und internen Funktionen/Einstellungen	Sichtbarkeit von Komponenten, Schnittstelle einer Klasse
Bau eines konkreten iPhone-Gerätes zu einem konkreten Modell	Instanziierung = Objekterzeugung
Kauf eines konkreten Gerätes durch einen Besitzer	Zuweisung einer Referenz
Versehentliches Verlieren eines Gerätes durch den Besitzer (Gerät liegt irgendwo herrenlos herum)	Objektreferenz versehentlich gelöscht (Objekt existiert referenzlos irgendwo im Hauptspeicher) →

Tab. 3-1

Analogie Handy-Bau –
Objektorientierung

Begriff aus dem Handy-Bau	Entsprechung in der Objektorientierung
Verschrottung eines iPhone-Gerätes	Freigeben der Objektinstanz
Informationen zu einzelnen Modellen (Stückzahlen, Statistiken usw.)	Klassenattribute und -methoden
Anrufen	Instanzereignis auslösen
Anruf annehmen	Instanzereignis behandeln

3.2 Klassen und ihre Komponenten

Klassen sind komplexe Softwarekonstrukte, die folgende Komponenten definieren:

Attribute speichern
Objektdaten.

■ Attribute
Beliebig komplexe Datentypen (Grundtypen, Strukturen, Tabellen, Tabellen von Strukturen, Strukturen mit Tabellen, Daten- und Objektreferenzen, Konstanten) beschreiben Attribute. Bei der Instanziierung werden oft Teile der Attribute mit Werten gefüllt. Häufig erlauben Klassen nicht den direkten Zugriff auf Datenattribute, sondern bieten SET(TER)- und GET(TER)-Methoden für spezielle Attribute an. Das erhöht die Sicherheit.

Methoden kapseln
ABAP-Code.

- Methoden
Das ist ABAP-Code (ähnlich Funktionsbausteinen), der auf die Attribute lesend und schreibend zugreift.
- Events
Über Ereignisse (Events) kann die Klasse bzw. ihre Objekte die Umwelt über Zustandsänderungen informieren.

3.2.1 PUBLIC-, PROTECTED- und PRIVATE-Komponenten

Alle Komponenten werden nach ihrer Sichtbarkeit für den Anwender einer Klasse und für die eigenen Methoden unterschieden:

■ PUBLIC
Dies ist die größtmögliche Sichtbarkeit für beliebige Nutzer. Öffentlichen Komponenten bilden gemeinsam die Schnittstelle der Klasse. Sie sind von außen zu erreichen/aufzurufen und natürlich auch von den Methoden der Klasse selbst.
Eine Besonderheit der PUBLIC-Attribute in ABAP OO ist das READ-ONLY-Kennzeichen. Für solche Attribute ist nur lesender Zugriff erlaubt, was für sie den Einsatz von GETTER-Methoden überflüssig macht und zudem schneller ist.

■ PROTECTED

Dies entspricht mittlerer Sichtbarkeit für Subklassen. Diese Komponenten sind in der Klasse selbst sichtbar, stehen aber auch allen abgeleiteten Klassen zur Verfügung.

■ PRIVATE

Mit dieser niedrigsten Sichtbarkeit für die eigene Klasse stehen die Komponenten nur in der Klasse selbst zur Verfügung.

Die Sichtbarkeit der Komponenten ermöglicht eine stufenweise Kapselung von Daten und Funktionen mit dem Ziel der Daten- und Funktionssicherheit.

*Stufenweise Kapselung
von Komponenten mit
PUBLIC, PROTECTED und
PRIVATE*

3.2.2 Statische und Instanzkomponenten

Klassen sind Typen, zu denen zur Laufzeit eines verwendenden Programms beliebig viele konkrete und wohlunterschiedene Ausprägungen angelegt werden können. Diese Ausprägungen werden als Objekte oder Instanzen bezeichnet.

Statische Attribute existieren je Klasse nur einmal, statische Methoden können ausschließlich auf diese Klassenattribute zugreifen. Instanzattribute existieren je Instanz. Instanzmethoden sehen die Attribute ihrer eigenen Instanz und auch die Klassenattribute.

Statische Komponenten werden häufig für Übersichtsaufgaben (Statistiken) zu allen Objekten der Klasse verwendet. In den Workflow-Klassen wird in den Klassenattributen oft eine Liste aller Objektkinstanzen verwaltet, die so performant wiedergefunden werden können.

*Statisches Attribut =
Klassenattribut = einmal
pro Klasse
Statische Methode =
Klassenmethode = sieht
nur Klassenattribute*

3.2.3 Abstrakte und finale Methoden und Klassen

Diese Eigenschaften von Klassen und Methoden sind nur im Zusammenhang mit Vererbung sinnvoll. Es kann daher u.U. zweckmäßig sein, zunächst das Unterkapitel über Vererbung zu lesen.

Abstrakte Klassen sind nicht instanzierbar, sie dienen nur als Abstraktionsstufe (Vorlage) im Vererbungsbaum. Von abstrakten Klassen werden normalerweise nichtabstrakte Klassen abgeleitet. Eine abstrakte Klasse kann sowohl statische als auch Instanzkomponenten definieren, insbesondere kann sie PUBLIC-Instanzmethoden implementieren. Sie darf sie nur nicht aufrufen, sondern nur weitervererben. Abstrakte Klassen, die ausschließlich statische Komponenten definieren, werden auch als Utility-Klassen bezeichnet, die Hilfsaufgaben ausführen, die keinerlei Instanziierung benötigen. Im Kapitel 11 wird so eine Klasse vorgestellt, die datenbankgestützte Konstanten liefert.

*Abstrakte Klassen sind
nicht instanzierbar.*