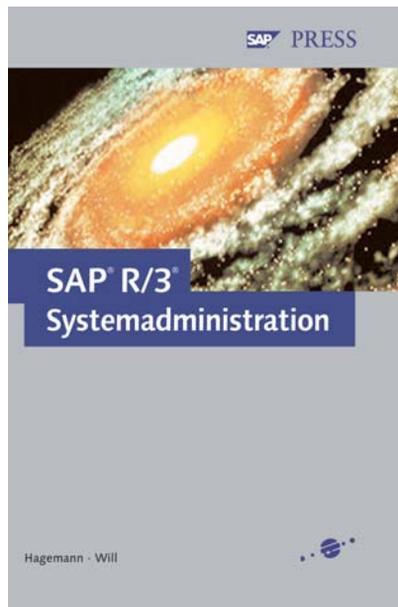


Sigrid Hagemann, Liane Will

SAP® R/3®-System- administration

Basiswissen für das R/3-System-
management



Inhalt

Geleitwort zur Buchreihe 13

Vorwort 15

Einleitung 17

1	Technische Umsetzung der Client/Server-Architektur in R/3	21
1.1	Client/Server-Architektur in R/3	21
1.2	Internet-Anbindung	25
1.2.1	Internet Transaction Server (ITS)	25
1.2.2	Internet Communication Manager (ICM)	26
1.2.3	Business Connector	27
1.3	Präsentationsebene	27
1.4	Applikationsebene	34
1.5	Datenbankebene	40
1.6	Netzwerk	43
1.7	Betriebssystem	44
1.7.1	Verzeichnisstruktur	45
1.7.2	Benutzer	47
1.8	Tipps	47
1.9	Pfade und Transaktionscodes	48
1.10	Weitere Dokumentation	48
1.11	Fragen	48
2	Erste Schritte	51
2.1	Starten von Datenbank und R/3-Instanzen	51
2.2	Stoppen von R/3-Instanzen und Datenbank	59
2.3	Starten der Frontends	60
2.4	Allgemeine administrative Aufgaben	61
2.4.1	Status	61
2.4.2	Systemüberwachung	62
2.4.3	Systemlog	69
2.4.4	Systemnachrichten	69
2.4.5	Listen	70

- 2.4.6 Tabellenpflege 71
- 2.5 Tipps 72
- 2.6 Pfade und Transaktionscodes 73
- 2.7 Weitere Dokumentation 74
- 2.8 Fragen 74

3 Service und Support 77

- 3.1 Remote-Verbindung 77
 - 3.1.1 Allgemeines 77
 - 3.1.2 SAProuter 80
 - 3.1.3 Verbindungsaufbau 82
- 3.2 SAP-Services 83
- 3.3 Wesentliche Dienste 85
 - 3.3.1 Problemmanagement 85
 - 3.3.2 Hinweisdatenbank 87
 - 3.3.3 Serviceverbindungen 87
 - 3.3.4 Registrierung von Entwicklern und Objekten 89
 - 3.3.5 Registrierung von Namensräumen 89
 - 3.3.6 Lizenzschlüssel 90
 - 3.3.7 Migrationskey 91
 - 3.3.8 Support Packages und Software-Download 91
- 3.4 Support-Tools 91
 - 3.4.1 EarlyWatch Alert 91
 - 3.4.2 Note Assistant 92
- 3.5 Überblick über den Solution Manager 93
- 3.6 Tipps 97
- 3.7 Transaktionen und Menüpfade 98
- 3.8 Weitere Dokumentation 98
- 3.9 Fragen 99

4 Installationskonzepte 101

- 4.1 Vorbereitungen 101
- 4.2 Architektur des Installationswerkzeugs R3setup 107
- 4.3 Installationsablauf 110
- 4.4 Änderungen mit SAP Web Application Server 117
 - 4.4.1 SAPinst 117
 - 4.4.2 MCOD: Multiple Components on One Database 121
- 4.5 Nachbereitung 122
- 4.6 Tipps 127

4.7	Transaktionen und Menüpfade	129
4.8	Weitere Dokumentation	129
4.9	Fragen	130
5	Einrichten der Systemlandschaft	133
5.1	Initialisierung des Transportwesens	134
5.2	Aufgaben einer Systemlandschaft	137
5.3	Konfiguration des Transport Management Systems	141
5.3.1	Transportdomänen	142
5.3.2	Transportwege	149
5.3.3	Erweiterte Transportsteuerung	153
5.3.4	QA-Genehmigungsverfahren	154
5.3.5	Transportstrategie	154
5.4	Transportsteuerungsprogramm tp	156
5.5	Tipps	157
5.6	Transaktionen und Menüpfade	157
5.7	Weitere Dokumentation	157
5.8	Fragen	158
6	Softwarelogistik	161
6.1	Implementation Guide	161
6.2	Objektbearbeitung	165
6.2.1	Änderungsaufträge	165
6.2.2	Bearbeitung von Aufträgen mit dem Transport Organizer	167
6.2.3	Transportprotokolle	177
6.2.4	Transport Organizer (Erweiterte Sicht)	182
6.2.5	Transport-Tools	184
6.3	Import von Transportaufträgen	185
6.4	Manuelle Bedienung des Transportsteuerungsprogramms tp	187
6.5	Einspielen von Support Packages und Industry Solutions	189
6.5.1	Einspielen von Support Packages	189
6.5.2	Einspielen von Add-Ons	191
6.6	Tipps	193
6.7	Pfade und Transaktionscodes	193
6.8	Weitere Dokumentation	194
6.9	Fragen	194

7	Mandantenverwaltung	197
7.1	Anlegen neuer Mandanten	199
7.2	Lokales Kopieren	205
7.3	Remote-Kopie	211
7.4	Mandantentransport	214
7.5	Sonderfunktionen	224
7.6	Tipps	225
7.7	Pfade und Transaktionscodes	228
7.8	Weitere Dokumentation	229
7.9	Fragen	230
8	R/3-Benutzer und Berechtigungen	231
8.1	Grundlagen	231
8.2	Benutzerpflege	232
8.2.1	Anlegen eines Benutzers	233
8.2.2	Lizenzdaten	237
8.2.3	Änderungen an Usern/Massenänderungen	238
8.2.4	Anmelde- und Kennwortschutz	238
8.2.5	Standardbenutzer	239
8.3	Berechtigungen	240
8.3.1	Überblick zur Berechtigungsprüfung	241
8.3.2	Berechtigungen und Berechtigungsobjekte	243
8.3.3	Berechtigungsprofile	245
8.3.4	Wichtige Profile im Umfeld der Systemadministration	247
8.3.5	Rollenpflege	248
8.3.6	Wichtige Rollen im Umfeld der Systemadministration	258
8.3.7	Benutzerzuordnung und Benutzerabgleich	259
8.3.8	Transport von Rollen	262
8.3.9	Vorgehen beim Upgrade	262
8.3.10	Fehleranalysen und Traces	263
8.4	Persönliche Einstellungen	264
8.5	Internetbenutzer	265
8.6	Informationen über Benutzer und Berechtigungen	266
8.6.1	Informationssystem	267
8.6.2	Security Audit Log	268
8.7	Zentrale Benutzerverwaltung	270
8.7.1	Einrichten des ALE-Szenarios	271
8.7.2	Aktivierung und Konfiguration der Zentralen Benutzerverwaltung	272
8.7.3	Löschen der Zentralen Benutzerverwaltung	274
8.7.4	Administration der Benutzer in einer Zentralen Benutzerverwaltung	274

8.8	Ausblick: Verzeichnisdienste	275
8.9	Tipps	276
8.10	Transaktionen und Menüpfade	276
8.11	Weitere Dokumentation	278
8.12	Fragen	279
<hr/>		
9	Hintergrundverarbeitung	281
9.1	Konzepte	281
9.2	Definition von Hintergrundjobs	284
9.2.1	Allgemeine Angaben	285
9.2.2	Startzeitpunkt	287
9.2.3	Verarbeitungsschritte	288
9.3	Auswertung	292
9.4	Analysefunktionen	295
9.5	Berechtigungen	299
9.6	Pflegejobs	301
9.7	Externe Schnittstelle	303
9.8	Tipps	303
9.9	Pfade und Transaktionscodes	304
9.10	Weitere Dokumentation	305
9.11	Fragen	306
<hr/>		
10	Verbuchung	307
10.1	Konzepte der Verbuchung	308
10.1.1	Verbuchungsmodus und Verbuchungsmodule	310
10.2	Konfiguration des Verbuchungssystems	312
10.3	Überwachung und Fehleranalyse der Verbuchung	314
10.3.1	Monitoring der Verbuchung	314
10.3.2	Fehleranalyse bei abgebrochenen Verbuchungen	317
10.4	Tipps	320
10.5	Transaktionen und Menüpfade	321
10.6	Weitere Dokumentation	321
10.7	Fragen	322
<hr/>		
11	Ausgabekonfiguration und -verwaltung	323
11.1	Grundlagen	323
11.2	Konfiguration der Spool-Workprozesse	325

- 11.3 Konfiguration der Aufbereitungsserver 327
- 11.4 Konfiguration der Landschaft der Ausgabegeräte 333
- 11.5 Auswertung und Fehlerbehebung 343
 - 11.5.1 Nutzungsstatistik 343
 - 11.5.2 Administrative Aufgaben 346
- 11.6 Berechtigungen 348
- 11.7 Tipps 350
- 11.8 Transaktionen und Menüpfade 350
- 11.9 Weitere Dokumentation 351
- 11.10 Fragen 352

12 Datenarchivierung 355

- 12.1 Archive Development Kit 356
- 12.2 Customizing 360
 - 12.2.1 Archivierungsobjekt-übergreifendes Customizing 364
 - 12.2.2 Archivierungsobjekt-spezifisches Customizing 365
 - 12.2.3 Basis-Customizing 367
 - 12.2.4 Anwendungsspezifisches Customizing 368
- 12.3 Steuerung und Auswertung 368
- 12.4 Tipps 372
- 12.5 Transaktionen und Menüpfade 374
- 12.6 Weitere Dokumentation 375
- 12.7 Fragen 375

13 Datenverteilung und -übernahme 377

- 13.1 RFC-Verbindungen 378
- 13.2 Application Link Enabling 385
 - 13.2.1 Technische Grundlagen 385
 - 13.2.2 Enge und lose Kopplung über BAPIs 389
 - 13.2.3 Konfiguration 390
 - 13.2.4 Beobachtung und Auswertung 397
- 13.3 Datenübernahme 401
 - 13.3.1 Batch-Input 402
 - 13.3.2 Direct-Input 405
 - 13.3.3 Fast-Input/Call Transaction 406
 - 13.3.4 BAPI 406
 - 13.3.5 Legacy System Migration Workbench 406
 - 13.3.6 Datenübernahme-Workbench 407

13.4	SAPconnect	407
13.5	Tipps	408
13.6	Transaktionen und Menüpfade	409
13.7	Weitere Dokumentation	410
13.8	Fragen	411
<hr/>		
14	Pflege der Instanzen	413
14.1	Profilpflege	413
14.2	Betriebsarten	426
14.3	Control Panel	434
14.4	Dynamische Benutzerverteilung	435
14.5	Tipps	438
14.6	Pfade und Transaktionscodes	439
14.7	Weitere Dokumentation	440
14.8	Fragen	441
<hr/>		
15	Systemüberwachung	443
15.1	Server- und Prozessübersichten	443
15.2	Benutzerübersichten	454
15.3	Systemlog	455
15.4	Analyse von Laufzeitfehlern	458
15.5	Trace-Dateien	459
15.6	Sperreinträge	465
15.7	Performance-Monitoring	466
15.7.1	Administrative Grundlagen	467
15.7.2	Monitore	467
15.8	Datenbankadministration	469
15.8.1	Regelmäßig einzuplanende Aufgaben	470
15.8.2	Datenbankfüllgrad und Objekte	472
15.9	Übersicht über regelmäßige Aufgaben	473
15.10	Tipps	476
15.11	Pfade und Transaktionscodes	477
15.12	Weitere Dokumentation	480
15.13	Fragen	480

16	Monitoring-Architektur	481
16.1	Alertmonitor	482
16.1.1	Grundlagen	482
16.1.2	Komponenten	484
16.1.3	Technische Realisierung	487
16.2	Customizing des Alertmonitors	492
16.2.1	Integration entfernter Systeme	492
16.2.2	Anlegen kundenspezifischer Monitore und Monitorsammlungen	493
16.2.3	Spezifische Anpassung der Eigenschaften	495
16.3	Auswertung der Alertmonitore	501
16.4	Customizing-Beispiele	502
16.4.1	Auswertung einer Protokoll- oder Logdatei	502
16.4.2	Autoreaktionsmethode: Mail versenden	504
16.4.3	Filtern des Systemlogs	505
16.5	Tipps	508
16.6	Pfade und Transaktionscodes	510
16.7	Weitere Dokumentation	511
16.8	Fragen	512

Anhang 513

A Kontrollfragen und Antworten 513

B Wichtige Transaktionscodes 529

C Profilparameter 535

D Menüstrukturen 541

E Glossar 545

F Literaturverzeichnis 553

Index 555

9 Hintergrundverarbeitung

Neben dem Dialog-Betrieb können in einem R/3-System Jobs auch im Hintergrund verarbeitet werden. Dies ist insbesondere für lang laufende Programme interessant, die keine interaktiven Eingaben erfordern. In diesem Kapitel steht das Management der Hintergrundjobs im Mittelpunkt. Es wird gezeigt, wie man Hintergrundjobs zeit- und ereignisgesteuert einplant und die Ablaufprotokolle auswertet.

9.1 Konzepte

Grundsätzlich können alle Programme, die keinen expliziten Benutzerdialog erfordern, auch im Hintergrund ausgeführt werden. Sinnvoll ist dies besonders dann, wenn der abzuwickelnde Vorgang zeit- und ressourcenintensiv ist und daher in eine lastarme Zeit verlegt werden soll. Eine Online-Ausführung würde über den gesamten Zeitraum einen Dialogprozess blockieren und so andere Dialogbenutzer indirekt behindern.

Hintergrund-
verarbeitung

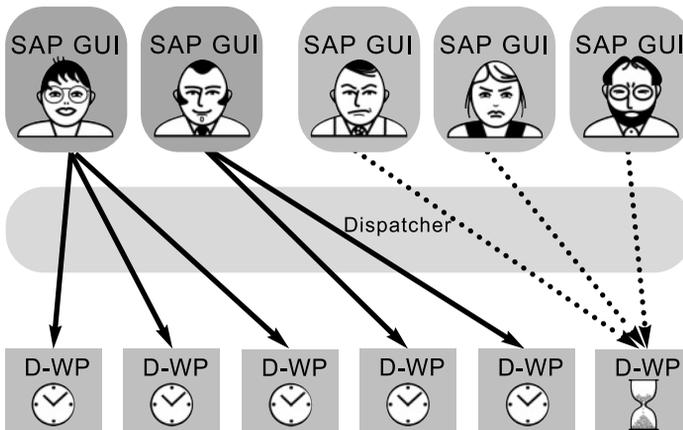


Abbildung 9.1 Langläufer im Dialogbetrieb

Um zu verhindern, dass Benutzer lang laufende Reports interaktiv ausführen, haben die Dialogschritte eine Laufzeitbegrenzung. Standardmäßig liegt diese Grenze bei 600 Sekunden. Nach Ablauf dieser Zeitspanne bricht die Verarbeitung ab. Im Systemprofil ist diese Grenze parametrisierbar (Parameter `rdisp/max_wprun_time`). Die Verarbeitung im Hintergrund ist nicht in dieser Form eingeschränkt.

Die Automatisierung von periodisch zu erledigenden Routinearbeiten ist eine weitere offensichtliche Anwendungsmöglichkeit. Für die Hintergrundverarbeitung stellt das R/3-System den Hintergrundservice mit seinen Hintergrund-Workprozessen (oder einfach nur *Background-Prozesse* genannt) zur Verfügung. Im Gegensatz zur Dialogverarbeitung, bei der jeder LUW (siehe Kapitel 1) der nächste freie Dialogprozess vom Dispatcher zugeordnet wird, besteht bei der Hintergrundverarbeitung während der ganzen Ausführung eine feste Verbindung mit genau einem Background-Prozess. Den Startzeitpunkt des Hintergrundjobs plant der Systemadministrator bzw. der Benutzer selbst. Er kann dabei zwischen Zeit- und Ereignissteuerung wählen.

Zeitgesteuerter Job-Scheduler

Bei der zeitgesteuerten Vorgehensweise wird bei der Einplanung des Jobs ein Startzeitpunkt definiert. Für jede Instanz des R/3-Systems, auf der Hintergrund-Workprozesse konfiguriert sind, ist ein *zeitgesteuerter Job-Scheduler* aktiv, der in definierten Zeitintervallen überprüft, ob Hintergrundjobs zur Verarbeitung anstehen. Die Beschreibungen der anstehenden Jobs werden in zentralen Tabellen der Datenbank gehalten. Bei diesem Scheduler handelt es sich um ein ABAP-Programm, das innerhalb eines ausgewählten Dialogprozesses interpretiert und abgearbeitet wird. Die Auswahl des speziellen Dialogprozesses wird beim Start des R/3-Systems von ihm automatisch vorgenommen. Standardmäßig ist das Zeitintervall, nach dem der zeitgesteuerte Job-Scheduler aktiv wird, auf 60 Sekunden konfiguriert. Der Administrator kann dieses Zeitintervall mit Hilfe des Parameters *rdisp/btctime* im Instanzprofil beliebig anpassen. Auf Grund des Zeitabstandes zwischen zwei Job-Scheduler-Läufen kann es daher Verzögerungen beim Start eines Jobs geben. Man würde das Zeitintervall also verkleinern, wenn die Verzögerungen zu groß für die eigenen Bedürfnisse sind. Ist umgekehrt die mögliche Verzögerung beim Start eines Jobs nicht ausschlaggebend, kann man das Zeitintervall vergrößern. Die damit verbundene Verringerung der Häufigkeit von Läufen des zeitgesteuerten Job-Schedulers hat allerdings kaum Einfluss auf die Last am System.

Ereignisge- steuerter Job- Scheduler

Im Gegensatz zum zeitgesteuerten Job-Scheduler reagiert der *ereignisgesteuerte Job-Scheduler* auf Events. Nach dem Auslösen eines Events veranlasst er den Start von Hintergrundjobs, die mit dem Eintreten dieses Ereignisses anlaufen sollen. Der ereignisgesteuerte Job-Scheduler wird ebenfalls durch einen Dialog-Workprozess abgearbeitet, die zu verwendende Instanz legen Sie über den Parameter *rdisp/btcname* = *<Rechnername>* im Standardprofil des R/3-Systems (*DEFAULT.PFL*) fest.

Die Ereignisse, auf die reagiert werden soll, müssen zunächst im R/3-System als so genannte *Events* definiert sein. Standardmäßig werden mit dem R/3-System bereits eine Reihe von Ereignissen ausgeliefert. Eine Übersicht erhalten Sie über die ►**Eventpflege**. Die bereits zum Lieferumfang gehörenden Ereignisse nennt man auch *Systemereignisse*. Sie werden häufig für die interne R/3-Steuerung benutzt, können jedoch auch von den R/3-Benutzern für eigene Zwecke verwendet werden.

Systemereignisse

Darüber hinaus können die Benutzer über den gleichen Menüpfad neue eigene Ereignisse, so genannte *Benutzerereignisse*, definieren. Die Ereignisdefinition ist zunächst nicht mehr als ein Eintrag in eine Tabelle.

Benutzerereignisse

Ein Ereignis kann auf verschiedene Weise getriggert werden:

Auslösen

- zu Testzwecken manuell über ►**Event auslösen**
- durch Benutzung des Funktionsbausteins BP_EVENT_RAISE aus einem ABAP-Programm innerhalb des R/3-Systems
- mit Hilfe des externen Programms **sapevt**

Aus einem externen Programm heraus kann mittels **sapevt** ein Ereignis im R/3-System ausgelöst werden. **sapevt** ist in dem SAP-Standard-Verzeichnis für ausführbare Programme (siehe Kapitel 1) verfügbar. Es wird wie folgt verwendet:

sapevt

```
sapevt <EventID> [-p <Parameter>] [-t]  
pf=<Profil>|name=<R/3-Systemname> nr=<R/3-Systemnummer>
```

Die Option `-t` veranlasst das Schreiben einer Protokolldatei `dev_evt` im Aufrufverzeichnis von **sapevt**. Mit Hilfe der Option `-p` kann ein Parameter, der ein R/3-Modul bestimmt (z. B. FI), übergeben werden. Damit wird eine Zuordnung der Ereignisse zu den Arbeitsgebieten erreicht. Diese Zuordnung hat lediglich beschreibenden Charakter.

Beispielsweise löst der Aufruf

```
sapevt SAP_TRIGGER_RDDIMPDP name=Q01 nr=00
```

im R/3-System Q01 das Ereignis SAP_TRIGGER_RDDIMPDP aus.

Innerhalb des R/3-Systems wird die Ereignissteuerung z. B. beim Transport von Objekten zwischen R/3-Systemen angewendet. Die mit Hilfe des Transportsteuerungsprogramms **tp** durchgeführten Transporte verlaufen in mehreren Phasen. Über den eigentlichen Datenimport hinaus müssen die einzelnen Objekte häufig auch generiert bzw. aktiviert werden. Daher löst das Programm **tp** nach Abschluss des Datenimports das Ereignis

nis SAP_TRIGGER_RDDIMPDP aus. In einem R/3-System ist stets der Job RDDIMPDP in Abhängigkeit von diesem Ereignis eingeplant. Tritt das Ereignis SAP_TRIGGER_RDDIMPDP ein, wird automatisch der Job RDDIMPDP im Hintergrund ausgeführt.

Die Verwendung dieser Technik erlaubt eine größere Flexibilität. Nicht immer ist es zeitlich vorhersehbar, wann Aktionen abgeschlossen sind, und eine Abhängigkeit zwischen Hintergrundjobs ist damit kaum herstellbar. Die Ereignissteuerung eröffnet dabei neue Möglichkeiten.

9.2 Definition von Hintergrundjobs

Für das Einrichten von Hintergrundjobs nutzen Sie die ►**Jobdefinition** (siehe Abbildung 9.2).

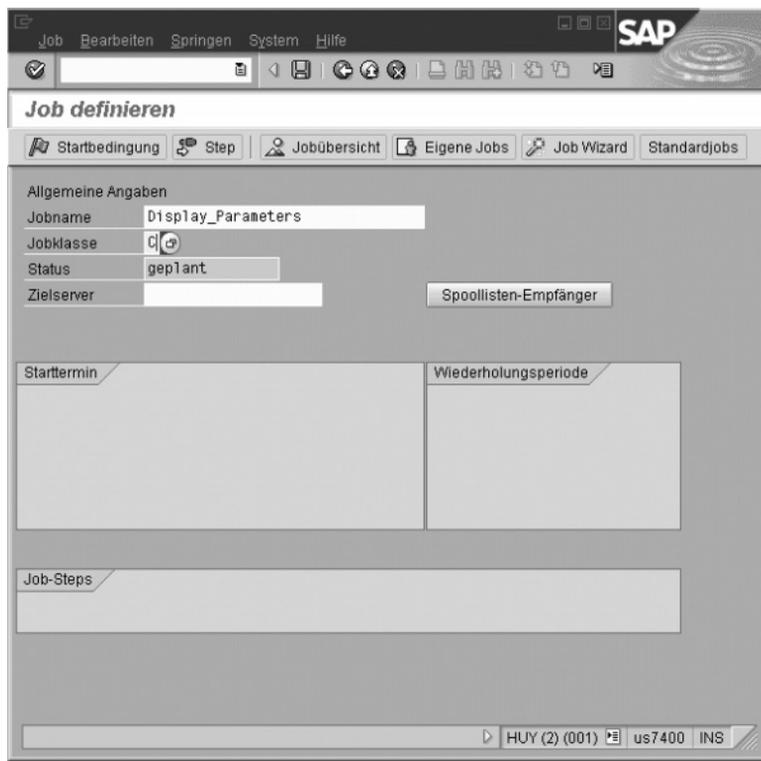


Abbildung 9.2 Eingangsbild Jobdefinition

Häufig ist die Planung von Hintergrundjobs auch direkt in die Anwendungen integriert, wie z. B. beim Kopieren von Mandanten oder beim Benutzerstammabgleich. Je nach Anwendung kann das Erscheinungsbild der

Bildschirmmasken zur Erfassung der Jobdaten differieren oder bestimmte Jobeigenschaften werden bereits von der Anwendung vorbelegt. Die in diesem Kapitel beschriebenen Grundsätze und Möglichkeiten der Hintergrundverarbeitung bleiben jedoch erhalten und können auf diese Spezialfälle übertragen werden.

Die Definition von Hintergrundjobs setzt sich aus drei wesentlichen Aspekten zusammen:

- ▶ allgemeine Angaben wie Jobname, Jobklasse und Zielrechner
- ▶ Angaben zum Startzeitpunkt bzw. der Zuordnung eines auslösenden Ereignisses
- ▶ Auflistung der Verarbeitungsschritte

9.2.1 Allgemeine Angaben

Den Ausgangspunkt für die Definition eines Hintergrundjobs bilden die allgemeinen Angaben (siehe Abbildung 9.2). Als Jobnamen sollten Sie eine möglichst aussagekräftige Zeichenfolge wählen, denn alle Protokolle und Übersichten, die später ausgewertet werden müssen, basieren auf diesem Jobnamen. Aus technischer Sicht ist der Name unerheblich, er braucht auch nicht eindeutig zu sein.

Die Priorität bei der Ausführung eines Jobs wird zunächst über die Zuordnung des Jobs zu einer Jobklasse gesteuert. Man unterscheidet die Jobklassen:

Jobklasse

▶ **A – höchste Priorität**

Jobs, die die Funktionstüchtigkeit von R/3 gewährleisten und zeitkritisch sind

▶ **B – mittlere Priorität**

periodische Jobs, die die Funktionstüchtigkeit von R/3 gewährleisten

▶ **C – normale Priorität**

Die Jobklasse C ist die für R/3-Anwender übliche Jobklasse.

Die Vergabe der Systemressourcen erfolgt anhand dieser Jobklasse. Stehen häufig sehr viele Jobs der Klasse C zur Verarbeitung an, so dass auch Jobs der Klassen A und B auf die Freigabe von Background-Prozessen warten müssen, kann der Systemadministrator für die Verarbeitung von Jobs der Klasse A mittels der ▶ **Betriebsartenpflege** eine Anzahl n Background-Prozesse freihalten. Durch diese Konfiguration wird sichergestellt, dass stets n Background-Prozesse zur Ausführung von Jobs der Klasse A bereitstehen. Jobs der Klassen B und C müssen mit der Abarbei-

tung warten, bis mindestens $n+1$ Prozesse verfügbar sind. Die Konfiguration wird in Abschnitt 14.2 im Rahmen der Betriebsartenpflege beschrieben.

Zielserver In einem verteilten R/3-System können Sie die Ausführung eines Jobs einer beliebigen R/3-Instanz mit Background-Service zuweisen. Diese R/3-Instanz wird im Kontext der Hintergrund-Verarbeitung als *Zielserver* bezeichnet. Verzichtet man auf die explizite Angabe eines Zielservers, wird zum Ausführungszeitpunkt der erste verfügbare Background-Prozess ausgewählt.

Die Prioritäten in der Abarbeitung des Auftrags auf einem definierten Hintergrund-Server sind wie folgt:

1. Jobklasse A, Zielserver ist angegeben
2. Jobklasse A, kein Zielserver angegeben
3. Jobklasse B, Zielserver ist angegeben
4. Jobklasse B, kein Zielserver angegeben
5. Jobklasse C, Zielserver ist angegeben
6. Jobklasse C, kein Zielserver angegeben

Sind die anstehenden Jobs nach den oben stehenden Kriterien gleichberechtigt, wird die Wartezeit herangezogen.

Wird ein Zielserver definiert, ist diese Angabe bindend. Ist er nicht verfügbar, wenn der Job gestartet werden soll, übernimmt kein Hintergrund-Workprozess einer anderen Instanz die Abarbeitung. Der Job bleibt in der Queue stehen, bis der definierte Zielserver wieder die Arbeit aufnimmt oder die Verarbeitung explizit auf einen anderen Server umgezogen wird.

Die von einem ABAP-Programm generierte Ausgabe wird im SAP-Spool-System als Spoolauftrag abgelegt. Mit Hilfe von **Spoollisten-Empfänger** kann die Ausgabe einem Benutzer zugeschickt werden. Auf diese Weise können z.B. Administration und Ergebnisauswertung eines Hintergrund-jobs von verschiedenen Personen wahrgenommen werden. Da die Ausgabe recht umfangreich sein kann, sollten Sie bei der Verwendung dieser Option vorsichtig vorgehen. Aus Performancegründen ist die Länge einer über SAPoffice versandten Ausgabeliste auf 1000 Zeilen beschränkt. In der Stepdefinition (siehe Abschnitt 9.2.3) werden Angaben zu den Druckparametern selbst gemacht.

9.2.2 Startzeitpunkt

In einem weiteren Schritt müssen die Parameter gewählt werden, die den Startzeitpunkt bestimmen. Dazu wählen Sie **Starttermin** aus dem Eingangsbildschirm zur Jobdefinition. Der Starttermin kann durch eine Zeitangabe festgelegt oder ereignisgesteuert definiert werden (Abbildung 9.3). Die Zeitangabe und die verwendete Zeitzone basieren auf der Systemzeit. Die zeitgesteuerte Einplanung von Jobs bietet neben der direkten Angabe des Startzeitpunkts auch die Möglichkeit der periodischen Einplanung, wie sie z. B. bei regelmäßigen Auswertungen oder den unten beschriebenen Pflegejobs (siehe Abschnitt 9.6) nützlich sein kann. Dabei können die Zeitabstände beliebig gewählt werden: im Minutenrhythmus, stündlich, täglich, wöchentlich usw. Über die Funktion **Einschränkungen** können Abweichungen von der üblichen Periode definiert werden, was z. B. für die Berücksichtigung von Feiertagen hinsichtlich des gültigen Fabrikkalenders günstig ist. Für zeitsensitive Jobs besteht die Möglichkeit, einen Zeitpunkt zu definieren, nach dessen Verstreichen sie keinesfalls mehr gestartet werden sollen.

Datum/Uhrzeit	
geplanter Start	Datum 24.10.2002 Uhrzeit 12:00:00
kein Start nach	Datum Uhrzeit

Abbildung 9.3 Starttermine bei der Einplanung von Hintergrundjobs

Sie können statt einer zeitlichen Steuerung auch ein definiertes Ereignis als Trigger festlegen. Insbesondere sind auch der Betriebsartenwechsel (siehe Kapitel 14) und ein Jobende als Events definiert, so dass ein Hin-

tergrundjob auch als Folgejob gestartet werden kann. Mit der Option **Start statusabhängig** können Sie den Start des Jobs von der erfolgreichen Beendigung des Vorgängerjobs abhängig machen. Bricht der Vorgängerjob ab, wird der abhängige Nachfolgejob auch in den Status *abgebrochen* versetzt und nicht ausgeführt.

Können Jobs mit Starttermin *Nach Job*, *Nach Ereignis* oder *Bei Betriebsart* nicht gestartet werden, weil bei Eintritt des erwarteten Events keine Hintergrund-Workprozesse frei sind, werden sie für den Start zum nächstmöglichen Zeitpunkt vorgemerkt und dann ebenfalls über den zeitgesteuerten Job-Scheduler gestartet.

9.2.3 Verarbeitungsschritte

Die Definition eines Hintergrundjobs komplettieren Sie durch die Beschreibung der Verarbeitungsschritte, aus denen er sich zusammensetzt. Unter einem Verarbeitungsschritt versteht man jeweils die Ausführung eines selbständigen Programms, z.B. eines ABAP-Programms oder auch eines externen Programms. Ein Hintergrundjob kann aus einem oder mehreren solcher Verarbeitungsschritte bestehen. Zur Definition der Arbeitsschritte wählen Sie aus der Jobdefinition die Funktion **Step** (siehe Abbildung 9.2). Jeder Verarbeitungsschritt oder Step kann unter Zuordnung eines vom Einplaner abweichenden Benutzers ausgeführt werden. Die Berechtigungsprüfungen finden immer auf der Basis des zugeordneten Benutzers statt. Somit können Fälle realisiert werden, in denen für die Jobeinplanung einerseits und -auswertung andererseits unterschiedliche Benutzerkreise zuständig und berechtigt sind. Die Verwendung explizit zugeordneter Benutzer kann die spätere Auswertung der Ergebnisse eines Hintergrundjobs auch erleichtern, da z. B. erzeugte Listen diesem Benutzer eindeutig zugeordnet werden. Eigens zu diesem Zweck können die bereits beschriebenen Background-Benutzer definiert werden (siehe Kapitel 8).

Arbeitsschritte können aus ABAP-Programmen, externen Kommandos oder externen Programmen gebildet werden (siehe Abbildung 9.4).

ABAP-Programm

Wie bereits erwähnt, können alle dialogfreien ABAP-Programme auch im Hintergrund ausgeführt werden. Dazu wählen Sie die Funktion **ABAP-Programm** aus (siehe Abbildung 9.4). Geben Sie den Namen des auszuführenden ABAP-Programms ein und gegebenenfalls eine Sprache an, in der ein eventuell entstehendes Protokoll erzeugt werden soll. Viele ABAP-Programme werden über Variablen gesteuert, wie z. B. das Programm **RSPP-PAR**. Dieses Programm erzeugt eine Liste aller eingestellten Instanzparameter. Der Namensraum der anzuzeigenden Instanzparameter kann vor

der Ausführung vom Benutzer eingeschränkt werden. Um derartige Programme im Hintergrund ausführen zu können, müssen so genannte *Varianten* eines Programms angelegt werden. Dahinter verbirgt sich eine feste Belegung für die Variablen eines Programms, die unter einem Variantennamen gesichert wird. Die Definition einer Variante erfolgt im ►**ABAP-Editor** mittels **Springen • Varianten**. Hier geben Sie einen Variantennamen und die gewünschten Parameterwerte ein. Eine so definierte Variante eines ABAP-Programms können Sie dann für die Hintergrundauführung einplanen. Abbildung 9.4 zeigt die Einplanung des ABAP-Programms **RSPFPAR**, für das zuvor eine Variante »ALLE« zur Erzeugung einer Liste aller aktuell definierten Instanzparameter angelegt wurde. Die Liste soll in deutscher Sprache erzeugt werden. Die Druckausgabe dieser Liste kann über die Funktion **Druckangaben** beeinflusst werden.

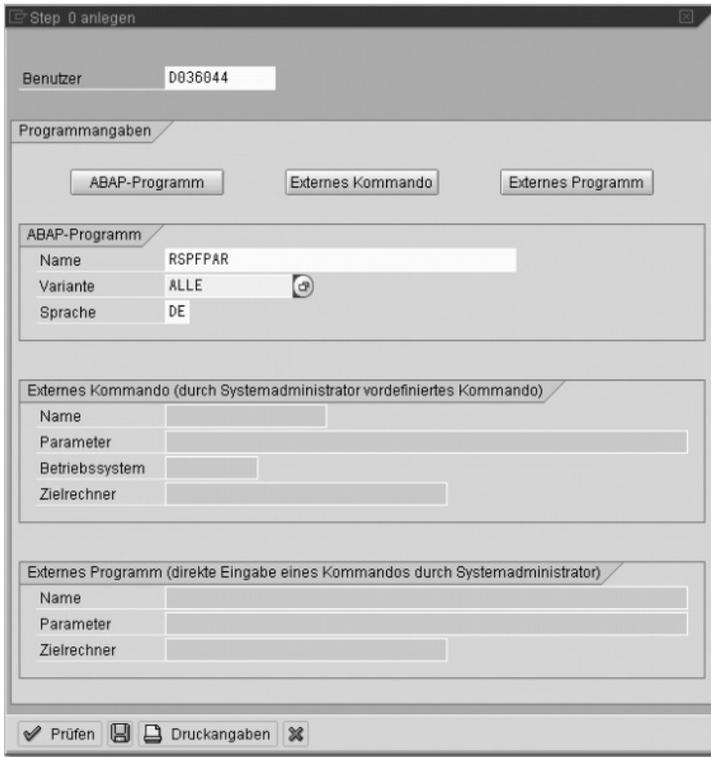


Abbildung 9.4 Stepdefinition

R/3-Benutzer mit Administratorberechtigung können über die Auswahl **Externes Programm** beliebige Programme auf Betriebssystemebene aus dem R/3-System heraus ausführen. Die Übergabe von Parametern ist mög-

**Externe
Programme**

lich, der Name des Zielrechners muss angegeben werden. Für die Ausführung des Programms wird auf dem Zielrechner die Routine **SAPXPG** gestartet, mit der das aufrufende R/3-System über die Kennung des speziellen R/3-Nutzers **SAPCPIC** (siehe Kapitel 8) dann per RFC kommuniziert.

Externe Kommandos

Um den R/3-internen Berechtigungsmechanismus ausnutzen, aber trotzdem in eingeschränktem Maße externe Programme verwenden zu können, sind *externe Kommandos* erweiterbar vorkonfiguriert. Ein externes Kommando besteht aus einem logischen Namen und einem zugeordneten externen Programm mit einer möglichen Parameterbelegung, die in Abhängigkeit vom gewählten Betriebssystem durchaus variieren kann. Vor der Verwendung externer Kommandos in der Hintergrundverarbeitung müssen diese mittels **►Anlegen externer Betriebssystemkommandos** definiert werden.

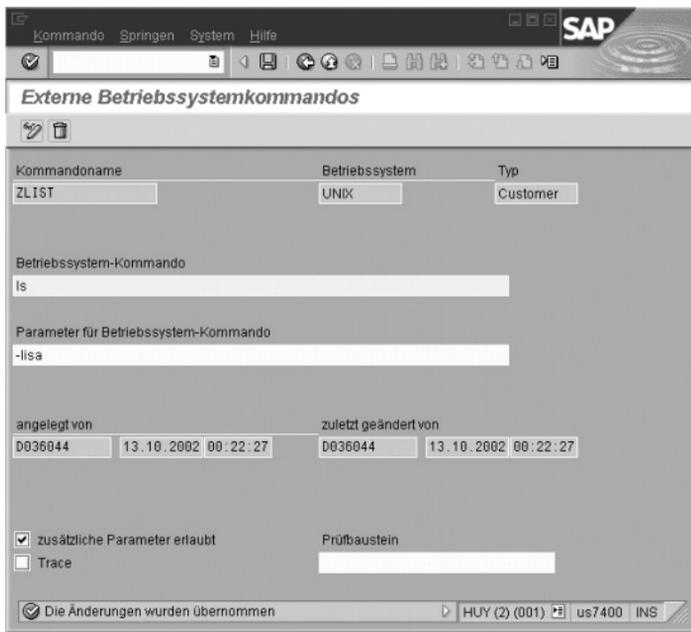


Abbildung 9.5 Anlegen eines externen Kommandos

Im Standardlieferungsumfang von R/3 sind bereits eine Vielzahl von externen Kommandos enthalten. Zusätzlich können beliebige weitere Kommandos vom Systemadministrator unter Berücksichtigung des Kundennamensraums angelegt werden. Abbildung 9.5 zeigt dies z. B. für ein Kommando **ZLIST**, hinter dem sich auf UNIX-Systemen das Kommando **ls** mit dem Parameter **-lisa** zur Anzeige des Inhalts des aktuellen Verzeichnisses verbirgt.

Entsprechend könnte für Windows NT-Systeme ein externes Kommando gleichen Namens, das das entsprechende Programm **dir** aufruft, angelegt werden. Ein so definiertes Kommando kann nicht nur bei der Spezifikation von Hintergrundjobs benutzt, sondern auch aus dem CCMS heraus ausgeführt werden. Dazu starten Sie ►**Externe Betriebssystemkommandos**, markieren das gewünschte Kommando und wählen anschließend **Kommando · Ausführen**.

Über die Definition eines Prüfbausteins kann die Verwendung eines externen Kommandos aus Sicherheitsgründen weiter eingeschränkt werden. Der Prüfbaustein wird vor dem Start des Kommandos durchlaufen, je nach Ergebnis der Prüfroutine wird das Kommando ausgeführt oder nicht. Die Routine SPXG_DUMMY_COMMAND_CHECK ist beispielhaft im System angelegt, sie kann für eigene Prüfzwecke als Vorlage dienen.

Bei der Definition eines Arbeitsschritts innerhalb eines Hintergrundjobs wird das auszuführende externe Kommando über den Namen, z.B. ZLIST, und das betreffende Betriebssystem, z.B. UNIX, bestimmt. Darüber hinaus kann dem Benutzer erlaubt werden, weitere Parameter zu den vordefinierten zu ergänzen. Der Zielrechner muss wie bei externen Programmen immer angegeben werden.

Werden externe Kommandos oder externe Programme in der Stepliste eines Hintergrundjobs verwendet, können Sie zur besseren Integration mittels der Option **Steuerflags** in der Stepdefinition festlegen, ob die Ausgaben und Fehlermeldungen des Betriebssystem-Programms in das Joblog des Steps aufgenommen werden sollen und ob eine synchrone oder asynchrone Ausführung gewünscht ist.

Wurden die allgemeinen Angaben, der Startzeitpunkt und die einzelnen Arbeitsschritte des Hintergrundjobs erfasst, ist die Definition des Hintergrundjobs komplett. Die getätigten Angaben müssen abschließend gesichert werden.

Alle beschriebenen Einträge können auch über den *Job Wizard* schrittweise abgefragt werden. Der Job Wizard lässt sich direkt aus der ►**Jobdefinition** aufrufen.

Job Wizard

Für die Einplanung von Hintergrundjobs stellt R/3 neben der eben beschriebenen menügeführten Methode auch eine Schnittstelle (*Application Programming Interface*, API) zur Verfügung, die den anwenderspezifischen Einsatz in eigenen ABAP-Programmen erlaubt.

API

9.3 Auswertung

Die Auswertung und Beobachtung der Hintergrundjobs erfolgt über die ►Einfache Jobauswahl oder die ►Erweiterte Jobauswahl. Die Jobs können nach verschiedensten Kriterien wie einplanender Benutzer, Zeitraum, Jobperiode oder Ereignis sowie Status (siehe Abbildung 9.6) gefiltert werden. Die Selektionskriterien werden über das Berechtigungskonzept eingeschränkt. Besitzen Sie die Administratorberechtigung für die Hintergrundverarbeitung, können Sie in der erweiterten Jobauswahl die Jobs aller Mandanten anzeigen lassen, sonst nur die des Anmeldemandanten.

The screenshot shows the SAP 'Einfache Jobauswahl' dialog box. It has a title bar with the SAP logo and a menu bar with 'Job', 'Bearbeiten', 'Springen', 'System', and 'Hilfe'. Below the menu is a toolbar with icons for search, refresh, and other functions. The main area is titled 'Einfache Jobauswahl' and contains several sections: 'Jobname' and 'Benutzername' with asterisks indicating required fields; 'Jobstatus' with checkboxes for 'Geplant', 'Freigegeben', 'Bereit', 'Aktiv', 'Fertig', and 'Abgebrochen'; 'Jobstartbedingung' with date pickers for 'von' (13.09.2002) and 'bis' (13.10.2002), and a field for 'oder nach Ereignis'; and 'Job-Step' with a field for 'ABAP-Programmname'. At the bottom right, there is a status bar showing 'IE4 (1) (100)' and 'psasb009 INS'.

Abbildung 9.6 Einfache Jobauswahl

Auf Grund der vorgenommenen Auswahl wird eine Liste aller Hintergrundjobs erzeugt (siehe Abbildung 9.7).

Der Status eines Jobs hat folgende Bedeutung:

► *geplant*

Die Stepdefinitionen des Jobs sind gesichert, ein Startzeitpunkt ist noch nicht definiert.

- ▶ *freigegeben*
Der Job ist geplant und ein Startzeitpunkt ist explizit gesetzt oder der Job wartet auf ein Ereignis.
- ▶ *bereit*
Die Startzeit wurde erreicht oder das erwartete Ereignis ist eingetreten, der Job wartet auf Systemressourcen zur Ausführung.
- ▶ *aktiv*
Der Job wird gerade bearbeitet.
- ▶ *fertig*
Der Job wurde erfolgreich zum Abschluss gebracht.
- ▶ *abgebrochen*
Auf Grund eines Problems musste die Verarbeitung abgebrochen werden. Der Job konnte nicht erfolgreich zu Ende geführt werden.

Jobname	Sp	Job-Ersteller	Status	Startdatum	Startzeit	Dauer (sec.)	Verzögerung (sec.)
<input type="checkbox"/> EU_REORG		D632081	fertig	12.10.2002	01:40:17	618	17
<input type="checkbox"/> F150-20021010-TEST -1		D634916	fertig	09.10.2002	23:00:42	2	0
<input type="checkbox"/> F150-20021010-TEST -3		D634916	fertig	09.10.2002	23:10:55	1	0
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	freigegeben			0	0
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	fertig	07.10.2002	03:30:50	31	50
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	fertig	08.10.2002	03:30:50	35	50
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	fertig	09.10.2002	03:30:14	35	14
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	fertig	10.10.2002	03:30:14	30	14
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	fertig	11.10.2002	03:30:18	35	18
<input type="checkbox"/> IAC_BATCH_PUBLISH		DDIC	fertig	12.10.2002	03:30:18	740	18
<input type="checkbox"/> JG_10_01		D636410	geplant			0	0
<input type="checkbox"/> MDVL_JOB		D636584	geplant			0	0
<input type="checkbox"/> PRODUKTIVSTART		D636410	geplant			0	0
<input type="checkbox"/> RDDFBCK		DDIC	fertig	07.10.2002	08:56:42	1	0
<input type="checkbox"/> RDDFBCK		DDIC	fertig	07.10.2002	09:00:31	1	0
<input type="checkbox"/> RDDIMPDP		D631495	fertig	07.10.2002	08:56:42	1	0

Abbildung 9.7 Liste der Hintergrundjobs

Durch einen Doppelklick auf einen ausgewählten Job wird das Jobprotokoll der Ausführung angezeigt. Neben dem Start- und dem Endzeitpunkt sind gerade bei Abbrüchen wertvolle Informationen über deren Ursachen enthalten. Das Jobprotokoll in Abbildung 9.8 wurde bei einer versuchten Datenextraktion erzeugt. Ursache für die Terminierung waren demnach doppelte Einträge in der Datenbank.

Datum	Uhrzeit	Nachrichtentext	N-Klasse	N-Nummer	N-Typ
09.10.2002	12:24:42	Job wurde gestartet	00	516	S
09.10.2002	12:24:42	Step 001 gestartet (Programm SBIE0001, Variante 40000808000413, Benutzername ALEREMTE)	00	550	S
09.10.2002	12:25:02	ABAP-Processor: ITAB_DUPLICATE_KEY	00	671	A
09.10.2002	12:25:02	Job wurde abgebrochen	00	516	A

Abbildung 9.8 Protokoll eines Abbruchs

In der Jobübersicht sind alle wesentlichen Operationen, die sich auf Hintergrundjobs beziehen, integriert. Dazu gehören:

- ▶ Anzeige der Einplanungsdaten.
- ▶ Abbrechen von Jobs im Status *aktiv*.
- ▶ Löschen von Jobs im Status *geplant*, *freigegeben*, *fertig* oder *abgebrochen*.
- ▶ Zurücknahme der Freigabe von einem oder mehreren Jobs, der Jobstatus ändert sich auf *geplant*.
- ▶ Vergleich von mehreren Jobs. Angegeben werden die allgemeinen Jobinformationen, die Stepdefinition und die Startbedingungen.
- ▶ Umzug auf einen anderen Server.
- ▶ Einfangen eines aktiven Jobs bei vermuteten Problemen (Langläufer). Ein Job, der gerade ein ABAP-Programm ausführt, kann auf diese Weise angehalten und mit dem ABAP-Debugger analysiert werden. Nach Verlassen des Debuggers läuft das Programm weiter.
- ▶ Status aktiver Jobs prüfen (siehe Abschnitt 9.4).
- ▶ Kopieren von geplanten, freigegebenen oder fertigen Jobs, der neue Job bekommt den Status *geplant*.

Neben dieser listenartigen Übersicht können Sie auch eine grafische Darstellung verwenden, die neben Anzeigefunktionen auch das Ändern und Freigeben von Jobs sowie das Prüfen aktiver Jobs ermöglicht. Der grafische Monitor ist über den Pfad ▶**Jobmonitor** erreichbar (siehe Abbildung 9.9). Die Status der Jobs werden hier zur besseren Übersicht farbig dargestellt.

Über die Auswahl ▶**Eigene Jobs** oder ▶**Jobdefinition · Eigene Jobs** können Sie sich außerdem einen Überblick über die eigenen Hintergrundjobs verschaffen.

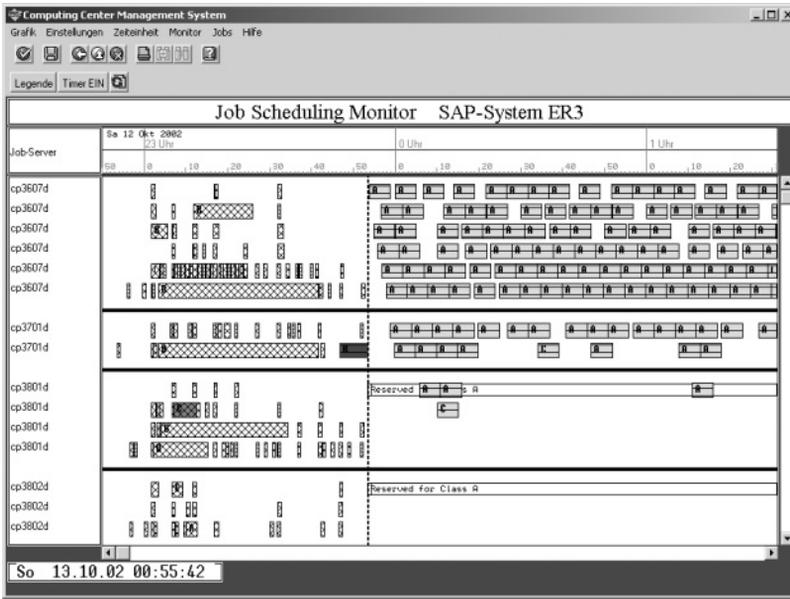


Abbildung 9.9 Jobmonitor

9.4 Analysefunktionen

Da Probleme eines Benutzers in der Hintergrundverarbeitung im Gegensatz zur Dialogarbeit dem Benutzer selbst oft nicht sofort auffallen, sind innerhalb des CCMS weitere Analysefunktionen verfügbar.

Bis R/3-Release 4.6B liefert die Funktion ►**Performanceanalyse** eine Zusammenstellung aller ausgewählten Hintergrundjobs hinsichtlich geplanter und tatsächlicher Startzeit sowie deren Laufzeit. Ab R/3-Release 4.6C sind diese Informationen in die ►**Einfache Jobauswahl** integriert. Größere Abweichungen zwischen geplantem und tatsächlichem Startzeitpunkt lassen auf einen Engpass bei den zur Verfügung stehenden Background-Prozessen schließen, denn das bedeutet, dass nach der Jobfreigabe ein langer Zeitraum verging, bevor ein Background-Prozess für die Verarbeitung zur Verfügung gestellt werden konnte. Lassen sich Performanceengpässe bei der Ausführung der eingeplanten Hintergrundjobs ausschließen, sollte der Administrator in diesem Fall die Ressourcen prüfen und gegebenenfalls die Anzahl der Background-Prozesse erhöhen (Parameter *rdisp/wp_no_btc* in den Instanzprofilen bzw. über die Profilpflege, siehe Kapitel 14).

Laufzeitanalyse

Zombies Beim Starten eines R/3-Systems wird geprüft, ob noch Jobs mit dem in dieser Situation nicht möglichen Status *bereit* oder *aktiv* gefunden werden. Sie werden dann auf den Status *geplant* oder *abgebrochen* gesetzt. Zu einem solchen *Zombie* kann es kommen, wenn ein Applikationsserver heruntergefahren wird, bevor der Job beendet und der Status in der Datenbank umgesetzt werden konnte.

Statusüberprüfung Um zu überprüfen, ob der angezeigte Jobstatus mit dem tatsächlichen Status übereinstimmt oder eine Inkonsistenz vorliegt, können Sie in der ►**Einfachen Jobauswahl** die kritischen Jobs markieren und über **Job • Status prüfen** eine mögliche Diskrepanz feststellen. Der Status der Jobs kann, wenn nötig, auf *geplant* zurückgesetzt oder die Jobs selbst können abgebrochen werden.

Alerts der Hintergrundverarbeitung Einige Parameter aus der Hintergrundverarbeitung wurden in die CCMS-Monitoring-Architektur integriert. Der Monitor *Background Processing* gibt Auskunft über die durchschnittliche Auslastung der Background-Workprozesse, die serverspezifische und gemittelte Länge der Warteschlange von startbereiten Jobs, die mangels Hintergrundserver nicht bearbeitet werden können, sowie die Anzahl abgebrochener Jobs (siehe Abbildung 9.10).

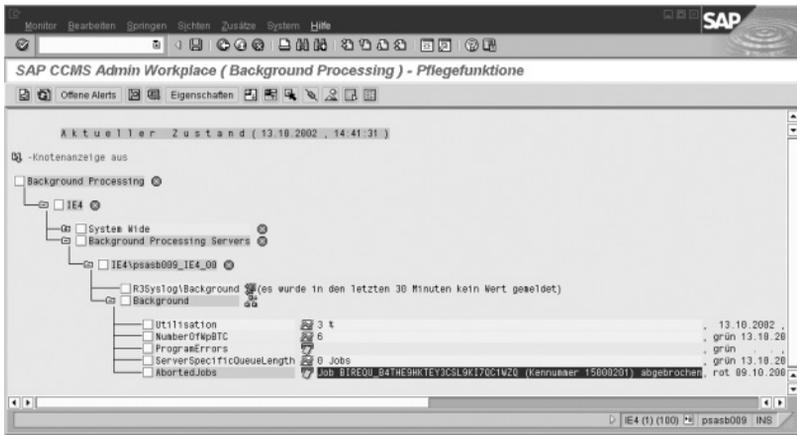


Abbildung 9.10 Integration der Hintergrundverarbeitung in das Alertmonitoring

Steuerungsobjektliste Die korrekte Funktionsweise der Controller in der Hintergrundverarbeitung kann über den ►**Hintergrund-Steuerungsobjekt-Monitor** überwacht werden. Wichtige Komponenten der Hintergrundverarbeitung wie

zeit- und ereignisgesteuerter Job-Scheduler, Zombie Cleanup, Start der externen Programme sowie das Umschalten der Betriebsarten können hier überprüft und über zusätzliche Trace-Ausgaben analysiert werden.

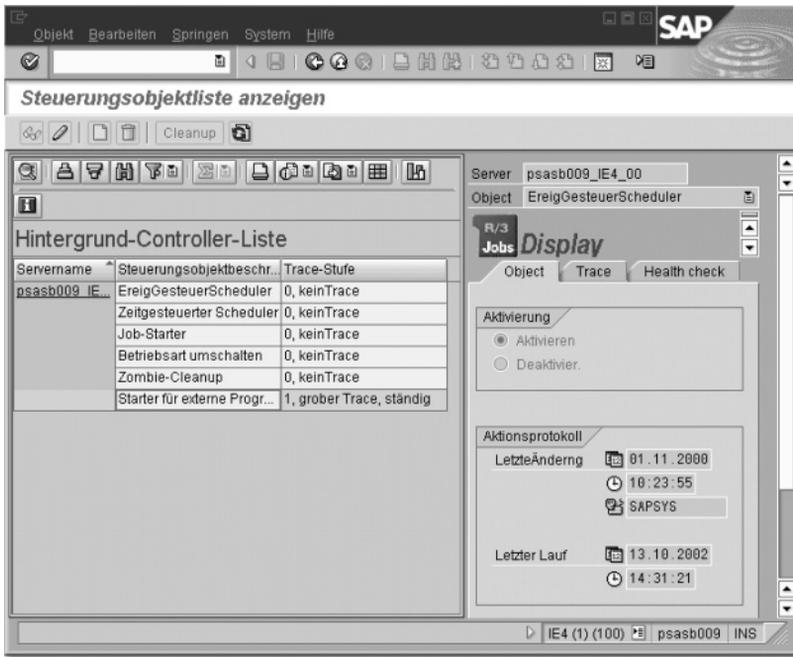


Abbildung 9.11 Monitor der Hintergrundsteuerung

Eine umfassende Analyse aller Aspekte der Hintergrundverarbeitung können Sie über die ►**Analyse der Hintergrundverarbeitung** durchführen. Insbesondere ist es im Rahmen dieses Analysetools möglich, Inkonsistenzen in den Datenbank-Tabellen zur Jobsteuerung zu melden und zu beheben. Im folgenden Listing sehen Sie ein Beispiel für eine Ausgabe dieses Tools:

Analysetool
Hintergrund-
verarbeitung

Listing 9.1 Ausgabe des Analysetools

```
*****
*   Analysetool der Hintergrundverarbeitung
*****
Test: Alle batchfähigen Server ermitteln
*****
*   Servername           Rechnername
*   psasb009_IE4_00     psasb009
*****
* Test: TemSe-Funktionalität testen
```

```

*****
*    ==> TemSe-Check fehlerfrei durchlaufen
*****

* Test: Batchberechtigungen eines Benutzers prüfen
*****

*    zu prüfender Benutzer = D036044
*    ==> folgende Berechtigungen liegen vor:
*        Batchadministrator      : ja
*        Early-Watch             : ja
*        Fremde Jobs löschen     : ja
*        Jobprotokolle anzeigen : ja
*        Jobs freigeben          : ja
*        Fremde Jobs anzeigen    : ja
*****

* Test: Umfeld für Start externer Programme testen
*****

*    ==> Benutzer SAPCPIC nicht definiert in Mandant
*        002
*        In diesem Mandanten ist ein Start externer
*        Programme nicht möglich !
*    ==> Benutzer SAPCPIC nicht definiert in Mandant
*        066
*        In diesem Mandanten ist ein Start externer
*        Programme nicht möglich !
*****

* Test: Konsistenzprüfung der Datenbanktabellen
*****

*    ==> Keine Inkonsistenzen entdeckt !
*    ==> Alle Jobkontexte sind konsistent
*****

* Test: Profileparameter verproben
*****

* Server = psasb009_IE4_00 , Datum = 13.10.2002 ,
* Uhrzeit = 14:35:46
*****

*    rdisp/btctime = 60
*    rdisp/wp no btc = 6
*    ==> Server ist richtig konfiguriert für
*        Hintergrundverarbeitung
*****

```

```

* Test: Lokalen Hostnamen gegen Messageserver verproben
*****
* Server = psasb009_IE4_00 , Datum = 13.10.2002 ,
* Uhrzeit = 14:35:46
*****
*   Lokaler Hostname = psasb009
*   ==> Lokaler Hostname stimmt mit Namen in
*       Messageserver überein
*****
* Test: Status der Batchworkprozesse eines Servers
* ermitteln
*****
* Server = psasb009_IE4_00 , Datum = 13.10.2002 ,
* Uhrzeit = 14:35:46
*****
*   ==> Status der Batch-Workprozesse:
*   WP 1 : wartet
*   WP 2 : wartet
*   WP 3 : wartet
*   WP 4 : wartet
*   WP 5 : wartet
*   WP 6 : wartet
*   Anzahl reservierter Klasse-A-Workprozesse: 0
*****
* Test: Anzahl Requests in Batch-Queue ermitteln
*****
* Server = psasb009_IE4_00 , Datum = 13.10.2002 ,
* Uhrzeit = 14:35:46
*****
*   ==> Anzahl Requests in Batch-Queue = 0
*****

```

9.5 Berechtigungen

Welche Aktionen ein Benutzer im Bereich der Hintergrundverarbeitung ausführen darf, wird wiederum mit Hilfe von Berechtigungen geregelt. Tabelle 9.1 gibt eine Übersicht über die wichtigsten Berechtigungen in diesem Bereich. Ohne besondere Berechtigung dürfen alle Benutzer ihre eigenen Jobs einplanen, abrechnen, löschen und deren Status prüfen. Besondere Berechtigungen werden benötigt, um

- ▶ einen Job zu manipulieren, der von einem anderen Benutzer eingeplant wurde
- ▶ das Jobprotokoll anzuzeigen
- ▶ einen von einem Hintergrundjob generierten Spool-Auftrag anzuzeigen
- ▶ einen Job zur Ausführung freizugeben
- ▶ ein externes Kommando zu benutzen

Berechtigung	Bedeutung
S_BTCH_ADM	Batchadministration
S_BTCH_JOB	Operationen mit Hintergrundjobs, mandantenabhängig mögliche Werte: DELE – Löschen von Jobs anderer Benutzer LIST – Anzeige von Spool-Listen anderer Benutzer PROT – Protokollanzeige anderer Benutzer RELE – Eigene Jobs einplanen und zur Ausführung freigeben SHOW – Anzeige von Job-Details anderer Benutzer Mit Hilfe des Feldes »Jobgruppe« kann die Berechtigung auf ausgewählte Jobnamen begrenzt werden.
S_BTCH_NAM	Verwendung eines expliziten Hintergrundbenutzers
S_DEVELOP	Einfangen von Jobs
S_LOG_COM	Ausführen externer Kommandos benötigte Parameter: COMMAND – Name des logischen Kommandos OPSYSTEM – Betriebssystem HOST – Name des Zielsystems
S_RZL_ADM	CCMS-Systemadministration
S_ADAMI_FCD	Systemberechtigung für Sonderfunktionen

Tabelle 9.1 Berechtigungen für die Hintergrundverarbeitung

Im Gegensatz zu einer Dialogverarbeitung werden im Hintergrundbetrieb keine Passwörter überprüft. Die beteiligten R/3-Benutzer müssen im aktuellen Mandanten lediglich angelegt und nicht gesperrt sein.

9.6 Pflegejobs

Es ist die Aufgabe des Systemadministrators, zum Erhalt der Funktionsfähigkeit seines R/3-Systems bestimmte Jobs in regelmäßigen Abständen ausführen zu lassen. Sie dienen z.B. dem Löschen nicht mehr benötigter Tabellen oder dem Sammeln von statistischen Daten für die Performanceanalyse. Die nachfolgende Tabelle 9.2 gibt eine Übersicht über die wichtigsten Programme. Je nach eingesetzten Applikationen oder eigenen Anwendungen können weitere Jobs erforderlich sein.

Empfohlener Jobname/Beschreibung	ABAP	Variante	Zeitraum
SAP_COLLECTOR_FOR_PERFMONITOR Sammeln der allgemeinen Statistikdaten für die Performanceanalyse des R/3-Systems. mandantenübergreifend Einplanung in Mandant 000 als DDIC	RSCOLL00	nein	stündlich
SAP_COLLECTOR_FOR_JOBSTATISTIC Sammeln der Statistikdaten zur Auswertung der durchschnittlichen Laufzeit periodisch ausgeführter Jobs. mandantenübergreifend	RSBPCOLL	nein	täglich
SAP_REORG_JOBS Löschen aller Protokolle von erfolgreich ausgeführten Jobs. Systemadministrator kann über Varianten definieren, nach wie vielen Tagen ein Protokoll gelöscht werden soll. mandantenübergreifend	RSBTCDEL	ja	täglich
SAP_REORG_JOBSTATISTIC Bereinigung der Laufzeitstatistiken der Hintergrundjobs. Es werden alle Objekte gelöscht, die älter als das angegebene Datum sind. mandantenübergreifend	RSBPSTDE	ja	monatlich
SAP_REORG_BATCHINPUT Löschen abgespielter Batchinput-Mappen und ihrer Protokolle sowie von Protokollen, zu denen keine Mappen mehr existieren. mandantenabhängig	RSBDCREO	ja	täglich, allerdings zu Zeiten ohne Batch-Input-Aktivität

Tabelle 9.2 Wichtige Pflegejobs

SAP_REORG_SPOOL Löschen veralteter Spool-Objekte mandantenabhängig	RSPO0041 / RSPO1041	ja	täglich
kein Standardname Löschen von Spool-Listen, die Überbleibsel abgebrochener Hintergrundprogramme sind. Konsistenzcheck der Spooler-Tabellen mandantenübergreifend	RSPO0043 / RSPO1043	ja	täglich
SAP_REORG_ABAPDUMPS Löschen von Laufzeitfehlereinträgen (Kurz-dumps). mandantenübergreifend	RSSNAPDL	ja	täglich
SAP_REORG_PRIPARAMS Reorganisation der Druckparameter mandantenübergreifend	RSBTCPRI- DEL	nein	monatlich
SAP_CCMS_MONI_BATCH_DP System-Monitoring mandantenabhängig	RSAL_ BATCH_ TOOL_ DISPAT- CHING	nein	stündlich
kein Standardname Umsetzung von Transportanforderungen	RDDIMPDP		ereignis- gesteuert
EU_PUT Verwaltungsjob zur Aktualisierung von Objektlisten und Navigationsindizes mandantenübergreifend	SAPRSLOG		täglich
EU_REORG Verwaltungsjob zur Aktualisierung der Objektlisten nach einem Transport mandantenübergreifend	SAPRSEUT		täglich

Tabelle 9.2 Wichtige Pflegejobs (Forts.)

Weitere Informationen zu den Eigenschaften und Parametern dieser Jobs können Sie über die Dokumentation jedes einzelnen Programms in Erfahrung bringen. Nutzen Sie dazu den ►**ABAP-Editor**. Geben Sie den Programmnamen ein und wählen Sie **Dokumentation • Anzeigen**.

Beginnend mit R/3-Release 4.6C können alle diese genannten Jobs über ►**Jobdefinition • Standardjobs** einzeln oder automatisch mit Standardparametrisierung eingepplant werden.

Neben den basisseitigen Pflegejobs können auch applikationsabhängige Reorganisationen die Systemperformance verbessern. Ein wichtiges Beispiel ist der Report **SBAL_DELETE** zum Löschen des Anwendungslogs.

9.7 Externe Schnittstelle

Über die SAP BC-XBP-Schnittstelle wird eine Möglichkeit bereitgestellt, R/3-Hintergrundjobs über externe Jobmanagement-Systeme anzubinden. Die unterstützten Funktionen sind

- ▶ Definition von Jobs
- ▶ Ändern, Bearbeiten und Löschen von Jobs
- ▶ Starten von Jobs
- ▶ Abbrechen aktiver Jobs
- ▶ Zugriff auf Jobinformationen (Status, Logdateien usw.)

Die an dieser Schnittstelle zertifizierten Produkte sind im SAP Service Marketplace unter dem Alias *background* aufgelistet.

9.8 Tipps

▶ Jobdefinitionen mit Angabe eines Zielrechners

Werden oft Jobdefinitionen mit Angabe eines Zielservers verwendet, muss darauf geachtet werden, bei Konfigurationsänderungen die Jobdefinitionen anzupassen. Dies gilt z. B. bei

- ▶ Umzug eines Applikationsservers auf andere Hardware (Änderung des Servernamens)
- ▶ Änderung der Aufteilung von Workprozessen bei einer Betriebsartendefinition

▶ Löschen von nicht mehr aktuellen Jobs im Status *geplant*

Bei der Anzeige der aktuellen Jobqueue über ▶**Einfache Jobauswahl** ist standardmäßig das Auswahlfeld **Jobstatus: geplant** nicht markiert, so dass dem Administrator nicht mehr benötigte Jobs in diesem Status eventuell nicht auffallen. Gerne wird es auch vergessen, das Auswahlfeld **oder nach Ereignis** zu beachten, so dass eventgetriggerte Jobs nicht angezeigt werden.

▶ Einplanung von Jobs unter einem nicht generischen User

Bei der Einplanung von periodischen Jobs, die über eine lange Zeit laufen sollen, ist es sinnvoll, die einzelnen Steps generischen Hintergrundbenutzern zuzuordnen. Sie vermeiden damit Probleme, die durch das Löschen von Benutzern entstehen, unter deren Account Jobs eingeplant sind.

► **Mindestanzahl von Prozessen**

Es müssen für das Transportwesen mindestens zwei Background-Workprozesse konfiguriert werden, auch wenn keine Hintergrundverarbeitung genutzt werden soll.

► **Ausplanung und Wiedereinplanung aller freigegebenen Jobs**

Seit R/3 4.5B wird der Report **BTCTRNS1** während eines Upgrades dazu verwendet, alle freigegebenen Jobs in einen vom Job-Scheduler nicht erkannten Status zu versetzen, um einen unerwünschten Start zu verhindern. Nach dem Upgrade können die Jobs über den Report **BTCTRNS2** wieder in den ursprünglichen Status zurückversetzt werden. Diese Funktionalität kann natürlich auch für andere Zwecke verwendet werden.

► **Verschiebung der Startzeiten einzelner zeitgesteuerter Job-Scheduler**

Werden mehrere Instanzen mit Hintergrund-Workprozessen genutzt, kann es sinnvoll sein, die Parameter *rdisp/btctime* in den Instanzprofilen auf unterschiedliche Werte zu setzen, um eine bessere Lastverteilung zu erzielen.

► **Probleme bei selbst einplanenden periodischen Jobs mit limitierter Startzeit**

Bei der Nutzung von periodischen Jobs, die sich am Laufende selbst wieder einplanen und einen Zeitpunkt definieren, nach dem sie nicht mehr gestartet werden können, kann es bei längerem Systemstillstand dazu kommen, dass die Jobs nicht mehr gestartet werden. Diese Jobs müssen manuell überwacht werden.

9.9 Pfade und Transaktionscodes

ABAP-Editor: SAP-Menü • Werkzeuge • CCMS • Jobs • Umgebung prüfen (SE38)

Analyse der Hintergrundverarbeitung: SAP-Menü • Werkzeuge • CCMS • Jobs • Umgebung prüfen (SM65)

Anlegen externer Betriebssystemkommandos: SAP-Menü • Werkzeuge • CCMS • Konfiguration • Externe Kommandos (SM69)

Betriebsartenpflege: SAP-Menü • Werkzeuge • CCMS • Konfiguration • Betriebsarten/Instanzen (RZ04)

Eigene Jobs: System • Eigene Jobs (SMX)

Einfache Jobauswahl: SAP-Menü • Werkzeuge • CCMS • Jobs • Pflege (SM37)

Erweiterte Jobauswahl: SAP-Menü • Werkzeuge • CCMS • Jobs • Pflege •
Erweiterte Jobauswahl (SM37C)

Event auslösen: SAP-Menü • Werkzeuge • CCMS • Jobs • Event auslösen
(SM64)

Eventpflege: SAP-Menü • Werkzeuge • CCMS • Jobs • Event pflegen
(SM62)

Externe Betriebssystemkommandos: SAP-Menü • Werkzeuge • CCMS •
Jobs • Externe Kommandos (SM49)

Hintergrund-Steuerungsobjekt-Monitor: SAP-Menü • Werkzeuge •
CCMS • Jobs • Hintergrundobjekte (SM61)

Jobdefinition: SAP-Menü • Werkzeuge • CCMS • Jobs • Definition
(SM36)

Jobmonitor: SAP-Menü • Werkzeuge • CCMS • Steuerung/Monitoring •
Jobeinplanungsmonitor (RZ01)

Performanceanalyse: SAP-Menü • Werkzeuge • CCMS • Jobs • Perfor-
manceanalyse (SM39)

9.10 Weitere Dokumentation

Quicklinks

► SAP Service Marketplace, Alias *background*

SAP Service Marketplace-Hinweise

Die nachfolgende Tabelle 9.3 gibt einen Überblick über wichtige SAP Service Marketplace-Hinweise im Zusammenhang mit der R/3-Hintergrundverarbeitung.

Inhalt	Hinweis
Standardjobs, Reorg-Jobs	16083
Verteilung von Hintergrundjobs auf Applikationsserver	24092
Fehleranalyse im Hintergrund- Verarbeitungssystem	37104
Verhalten der Transaktionen SM37 und SM37C	422000

Tabelle 9.3 Hinweise zur Hintergrundverarbeitung

9.11 Fragen

1. Mit welcher Transaktion können die Jobprotokolle ausgewertet werden?

- a. SE38
- b. SM37
- c. S000

2. Mit welchem externen Programm können Ereignisse im R/3-System ausgelöst werden?

- a. sapevt
- b. sapxpg
- c. sapstart
- d. spmon

3. Was bedeutet der Status »bereit« für einen Hintergrundjob?

- a. Die Einplanung des Jobs wurde vervollständigt und gesichert.
- b. Der Job wurde ausgeführt und ist bereit zum Drucken des Protokolls.
- c. Der Job wartet auf Systemressourcen, damit die Ausführung beginnen kann.