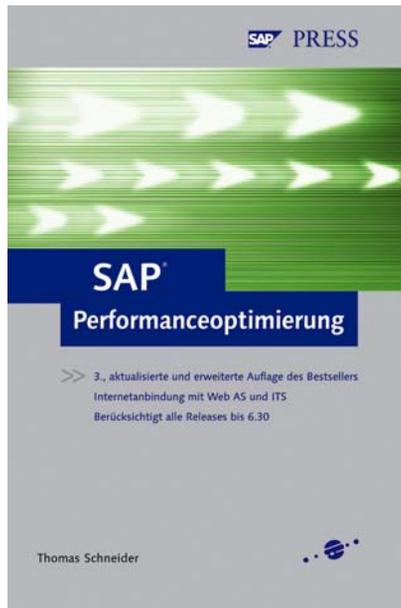


Thomas Schneider

# SAP-Performanceoptimierung

Analyse und Tuning von SAP-Systemen



# Inhalt

<b>Geleitwort zur Buchreihe</b>	<b>13</b>
<b>Danksagung</b>	<b>15</b>
<b>Einleitung</b>	<b>17</b>
Von SAP R/3 zur mySAP Business Suite .....	20
Zu diesem Buch .....	23
<b>1 Performancemanagement einer mySAP-Lösung</b>	<b>29</b>
1.1 Die Architektur von mySAP-Lösungen .....	29
1.1.1 mySAP-Lösungen und -Komponenten .....	29
1.1.2 Client/Server-Architektur .....	32
1.2 Das Überwachungs- und Optimierungskonzept für eine mySAP-Lösung .....	37
1.2.1 Anforderungen an ein Überwachungs- und Optimierungskonzept .....	37
1.2.2 Werkzeuge und Methoden für das Überwachungs- und Optimierungskonzept .....	41
1.2.3 Service Level Management .....	44
1.2.4 Kontinuierliche Überwachung .....	50
1.2.5 Das Konzept für eine kontinuierliche Performanceoptimierung ..	61
1.2.6 Das Solution-Management-Programm der SAP .....	66
1.3 Zusammenfassung .....	70
<b>2 Analyse von Hardware, Datenbank und SAP-Basis</b>	<b>73</b>
2.1 Begriffsklärungen .....	73
2.2 Hardwareanalyse .....	74
2.2.1 Analyse eines Hardwareengpasses (CPU und Hauptspeicher) ...	75
2.2.2 Identifizierung von Schreib/Lese-(I/O-)Problemen .....	80

2.2.3	Weitere Analysen auf Betriebssystemebene .....	81
2.2.4	Zusammenfassung .....	82
<b>2.3</b>	<b>Datenbankanalyse</b> .....	<b>84</b>
2.3.1	Analyse der Datenbankpuffer .....	84
2.3.2	Identifizierung teurer SQL-Anweisungen .....	90
2.3.3	Identifizierung von Schreib/Lese-(I/O-)Problemen .....	96
2.3.4	Weitere Analysen auf Datenbankebene .....	97
2.3.5	Zusammenfassung .....	103
<b>2.4</b>	<b>Analyse der SAP-Speicherkonfiguration</b> .....	<b>104</b>
2.4.1	Analyse der SAP-Puffer .....	105
2.4.2	Analyse des SAP-Extended Memory, des SAP-Heap Memory und des SAP-Roll Memory .....	107
2.4.3	Anzeige des allokierten Speichers .....	109
2.4.4	Weitere Monitore zur detaillierten Analyse .....	111
2.4.5	Zusammenfassung .....	111
<b>2.5</b>	<b>Analyse der SAP-Workprozesse</b> .....	<b>113</b>
2.5.1	Felder der Workprozessübersicht .....	114
2.5.2	Analyse der Workprozesse .....	116
2.5.3	Globale Workprozessübersicht .....	119
2.5.4	Überwachung der Dispatcher-Queue .....	121
2.5.5	Zusammenfassung .....	121
<b>2.6</b>	<b>Zusammenfassung</b> .....	<b>123</b>

## **3 Workload-Analyse 127**

<b>3.1</b>	<b>Der Workload-Monitor</b> .....	<b>128</b>
3.1.1	Funktionalität und Verfügbarkeit .....	128
3.1.2	Mit dem Workload-Monitor arbeiten .....	128
3.1.3	Technische Einstellungen zum Workload-Monitor .....	132
<b>3.2</b>	<b>Workload-Analyse</b> .....	<b>133</b>
3.2.1	Ablauf eines Transaktionsschrittes .....	133
3.2.2	Interpretation der Antwortzeiten .....	136
3.2.3	Aktivität, Durchsatz und Last .....	139
<b>3.3</b>	<b>Workload-Analyse durchführen</b> .....	<b>140</b>
3.3.1	Allgemeines Performanceproblem analysieren .....	141
3.3.2	Spezielles Performanceproblem analysieren .....	148
<b>3.4</b>	<b>Anwendungsmonitor</b> .....	<b>152</b>
3.4.1	Benutzerprofil .....	152
3.4.2	Last pro SAP-Modul .....	153
3.4.3	SAP-Puffer .....	154
<b>3.5</b>	<b>Zusammenfassung</b> .....	<b>155</b>

## **4 Performanceanalyse von ABAP-Programmen 159**

4.1	Einzelstatistik .....	159
4.2	Performance-Trace .....	163
4.2.1	Performance-Trace erstellen .....	163
4.2.2	SQL-Trace auswerten .....	165
4.2.3	Weitere Funktionen im SQL-Trace .....	169
4.2.4	RFC-Trace auswerten .....	172
4.2.5	Enqueue-Trace auswerten .....	172
4.3	Performanceanalyse mit dem ABAP-Trace (Laufzeitanalyse) .....	173
4.3.1	ABAP-Trace erstellen .....	174
4.3.2	ABAP-Trace auswerten .....	175
4.3.3	Varianten verwenden .....	176
4.4	Performanceanalyse mit dem ABAP-Debugger .....	178
4.5	Zusammenfassung .....	180

## **5 Lastverteilung 183**

5.1	SAP-Services .....	183
5.1.1	Message-, Enqueue- und ATP-Service verteilen .....	187
5.1.2	Dialog-, Hintergrund- und Spool-Workprozesse verteilen .....	187
5.1.3	Benutzer und Workprozesse auf CPU-Ressourcen verteilen .....	188
5.2	Dynamische Benutzerverteilung (Logon-Gruppen) und Betriebsarten .....	191
5.2.1	Dynamische Benutzerverteilung konfigurieren .....	191
5.2.2	Betriebsarten planen .....	194
5.3	Verbuchung .....	195
5.3.1	Verbuchungsaufträge überwachen .....	196
5.3.2	Verbuchungsworkprozesse verteilen .....	197
5.3.3	Verbuchungsarten wählen .....	198
5.4	Hardware sizing .....	202
5.4.1	Initiales Sizing planen .....	203
5.4.2	SAP Standard Application Benchmarks .....	207
5.4.3	Hardwarekapazitätsplanung bei Laststeigerung, Releasewechsel oder Migration (Re-Sizing) .....	209
5.4.4	Neue Herausforderungen durch das Internet .....	211
5.5	Systemlandschaft planen .....	212
5.5.1	Verteilung von SAP-Instanzen .....	213
5.5.2	Hardwarekonsolidierung .....	214
5.5.3	Mehrere SAP-Systeme in einer Datenbank .....	216
5.6	Zusammenfassung .....	216

## **6 Schnittstellen 219**

6.1	RFC-Grundlagen .....	219
6.1.1	Begriffe .....	219
6.1.2	Ablauf eines RFCs .....	222
6.2	Schnittstellen zu externen Systemen .....	224
6.2.1	RFC-Verbindungen konfigurieren und testen .....	225
6.2.2	Eingehende und ausgehende Last überwachen .....	228
6.2.3	Parallelisierung von Prozessen mit asynchronen RFCs konfigurieren .....	233
6.2.4	Datenübertragung mit transaktionalen RFCs überwachen .....	234
6.3	Zusammenfassung .....	235

## **7 SAP GUI und Internetanbindung 237**

7.1	SAP GUI 4.6 .....	237
7.1.1	Interaktionsmodell und Performancemessung .....	238
7.1.2	Performance der GUI-Kommunikation analysieren und optimieren .....	240
7.2	SAP Web Application Server (SAP Web AS) .....	246
7.3	SAP Internet Transaction Server (SAP ITS) .....	247
7.3.1	ITS-Grundlagen .....	247
7.3.2	Web-Anbindung planen .....	251
7.3.3	ITS konfigurieren .....	252
7.3.4	ITS-Administrationswerkzeuge .....	255
7.3.5	Engpassanalyse für den ITS durchführen .....	257
7.4	Business Server Pages (BSP) und Internet Communication Manager (ICM) .....	258
7.4.1	Grundlagen von SAP Business Server Pages .....	258
7.4.2	ICM konfigurieren und überwachen .....	262
7.4.3	Performanceanalyse von BSPs durchführen .....	264
7.5	SAP J2EE Engine .....	267
7.5.1	Grundlagen der SAP J2EE Engine .....	268
7.5.2	Konfigurations- und Überwachungswerkzeuge .....	272
7.6	End-to-End-Performanceüberwachung .....	275
7.6.1	Laufzeitanalyse im Webbrowser durchführen .....	275
7.6.2	Web-Anwendungen mit dem CCMS kontinuierlich überwachen .....	278
7.6.3	Werkzeuge von Drittanbietern .....	279
7.7	Zusammenfassung .....	279

## **8 Speicherkonfiguration 283**

8.1	Grundlagen der Speicherkonfiguration .....	283
8.1.1	Begriffsklärungen .....	283
8.1.2	SAP-Roll Memory, SAP-Extended Memory, SAP-Heap Memory .....	285
8.1.3	SAP-EG Memory und SAP-Paging Memory .....	294
8.2	SAP-Speicherbereiche konfigurieren und überwachen .....	296
8.2.1	Auslagerungsspeicher überwachen .....	298
8.2.2	Beschränkungen durch den Adressraum (32- und 64-Bit-Architektur) .....	301
8.2.3	SAP-Speicherbereiche konfigurieren und überwachen .....	306
8.2.4	Hilfe zur Fehlerbehebung .....	310
8.3	Zusammenfassung .....	316

## **9 SAP-Tabellenpufferung 319**

9.1	Vorbemerkung zu den Kapiteln 9, 10 und 11 .....	320
9.2	Grundlagen der Tabellenpufferung .....	321
9.2.1	Pufferungsarten .....	321
9.2.2	Pufferzugriffe .....	324
9.2.3	Puffersynchronisation .....	326
9.2.4	Pufferung einschalten .....	328
9.2.5	Welche Tabellen sollen gepuffert werden? .....	330
9.3	SAP-Tabellenpufferung überwachen .....	332
9.3.1	Tabellenzugriffsstatistik .....	333
9.3.2	Analyse der gepufferten Tabellen .....	336
9.3.3	Analyse der bisher nicht gepufferten Tabellen .....	342
9.3.4	Detaillierte Tabellenanalyse .....	344
9.3.5	Monitor zur Puffersynchronisation (DDLOG-Einträge) .....	346
9.3.6	Shared SQL Area und SQL-Trace .....	348
9.4	Zusammenfassung .....	349

## **10 Sperren 353**

10.1	Sperrkonzepte von Datenbanksystem und SAP-System .....	353
10.1.1	Datenbanksperren .....	354
10.1.2	SAP-Enqueues .....	355
10.2	Überwachung von Sperren .....	357
10.2.1	Datenbanksperren .....	357
10.2.2	SAP-Enqueues .....	360

10.3	Nummernkreispufferung .....	362
10.3.1	Grundlagen .....	362
10.3.2	Nummernkreispufferung einschalten .....	366
10.3.3	Nummernkreispufferung überwachen .....	367
10.4	ATP-Server .....	368
10.4.1	Grundlagen .....	369
10.4.2	ATP-Server konfigurieren .....	371
10.4.3	ATP-Server überwachen .....	374
10.5	Zusammenfassung .....	374

## **11 Optimierung von SQL-Anweisungen 377**

11.1	Teure SQL-Anweisungen identifizieren und analysieren .....	378
11.1.1	Identifizierung und Voranalyse .....	378
11.1.2	Detaillierte Analyse .....	380
11.2	SQL-Anweisungen durch Sekundärindizes optimieren .....	384
11.2.1	Ein wenig Theorie .....	384
11.2.2	Indizes und Tabellenstatistiken administrieren .....	393
11.2.3	Regeln zum Anlegen oder Ändern von Sekundärindizes .....	398
11.3	SQL-Anweisungen im ABAP-Programm optimieren .....	405
11.3.1	Regeln für die performante SQL-Programmierung .....	405
11.3.2	Beispiel für die Optimierung eines ABAP-Programms .....	410
11.3.3	Vorbelegung von Feldern in Reporttransaktionen .....	419
11.4	Zusammenfassung und weitere Hinweise .....	425

## **A Flussdiagramme und Checklisten zur Performanceanalyse 431**

A.1	Flussdiagramme .....	431
A.2	Checklisten .....	437

## **B Datenbankmonitore 447**

B.1	Datenbankprozessmonitor .....	447
B.2	Statistik über SQL-Anweisungen (Shared SQL Area) .....	448
B.3	Festplattenzugriffe .....	449
B.4	Datenbanksperrern (Exclusive Lockwaits) .....	450
B.5	Datenbankpuffer .....	451
B.6	Ausführungspläne .....	462

<b>C</b>	<b>Performancerelevante Konfigurationsparameter</b>	<b>475</b>
C.1	Puffer .....	476
C.2	Speicherverwaltung .....	478
C.3	Weitere Parameter .....	480
C.4	Internet Communication Manager .....	483
C.5	Internet Transaction Server .....	485
<b>D</b>	<b>Wichtige Transaktionscodes</b>	<b>487</b>
<b>E</b>	<b>Fragen und Antworten</b>	<b>489</b>
<b>F</b>	<b>Glossar</b>	<b>501</b>
<b>G</b>	<b>Informationsquellen</b>	<b>511</b>
<b>H</b>	<b>Hinweise im SAP Service Marketplace</b>	<b>515</b>
	<b>Index</b>	<b>519</b>

## 6 Schnittstellen

Schnittstellen stellen einen erheblichen Faktor bei Performanceüberlegungen dar. Nachdem im ersten Abschnitt eine allgemeine Einführung in die RFC-Technik gegeben wird, die die Grundlage von fast allen Schnittstellen ist, widmet sich der zweite Abschnitt der Konfiguration und Überwachung von Schnittstellen zu externen Systemen.

### Wann sollten Sie dieses Kapitel lesen?

In diesem Kapitel finden Sie Informationen zur Konfiguration und Überwachung von RFC-Schnittstellen zu externen Systemen.

### 6.1 RFC-Grundlagen

Remote Function Calls (RFCs) erlauben es einem Programm, ein anderes Programm »remote« (d.h. »an einem anderen Ort«) auszuführen.

#### 6.1.1 Begriffe

RFCs werden zu folgenden Zwecken verwendet:

- ▶ Zur Kommunikation zwischen unterschiedlichen Systemen, so zwischen zwei SAP-Systemen oder zwischen einem SAP-System und einem externen System
- ▶ Innerhalb eines SAP-Systems:
  - ▶ Zur Kommunikation zwischen den Applikationsinstanzen oder zwischen der Applikationsebene und der Präsentationsebene (GUI-Kommunikation)
  - ▶ Zur Parallelisierung von Prozessen: Da ein Programm nacheinander mehrere RFCs asynchron starten kann, ohne deren Verarbeitungsende abwarten zu müssen, werden RFCs dazu benutzt, Prozesse zu parallelisieren und Last dynamisch auf die unterschiedlichen Rechner innerhalb eines SAP-Systems zu verteilen.

Mit RFCs können SAP-Systeme (z. B. SAP R/3, SAP APO, SAP BW) unterschiedlicher Versionen oder SAP-Systeme mit externen Systemen, z. B. ein SAP R/3 mit einem externen Lagerverwaltungssystem, gekoppelt werden. Dabei erstellt das SAP R/3 die Transportaufträge für die Lagerbewegungen und sendet diese Transportaufträge per RFC an das externe System. Hat das Lagerverwaltungssystem die Lagerbewegungen durchgeführt, so führt es seinerseits per RFC im SAP R/3 die Funktion aus, die die Warenbewegung bestätigt.

Wie die Geschäftsprozesse eines Unternehmens auf verschiedene SAP-Komponenten aufgeteilt werden, ist von kritischer Bedeutung für die spätere Betriebsführung des Systems. Für eine verteilte gekoppelte Systemlandschaft spricht, dass die einzelnen Teilsysteme in jeder Beziehung (Aufbau, Upgrade, Administration, Organisation etc.) flexibler zu handhaben sind. Durch die Schnittstellen, die aufgebaut und betrieben werden müssen, wird in Summe der Aufwand für Aufbau und Administration (und der Hardwarebedarf) größer als in einem großen integrierten System.

Mit RFCs lassen sich Systeme auf zwei Arten koppeln; wir unterscheiden:

- ▶ die so genannte »harte« *Kopplung*, bei der das eine System darauf angewiesen ist, dass das Partnersystem verfügbar ist. Ist die Kommunikation gestört, z. B. durch eine Netzwerkstörung oder weil das Partnersystem nicht betriebsbereit ist, kann zwar das andere System als solches weiterarbeiten, die Funktionen, die die RFCs verwenden, werden jedoch mit einem Fehler abbrechen.
- ▶ Die »weiche« *Kopplung* setzt die gegenseitige Verfügbarkeit der Systeme nicht voraus. Vielmehr tauschen die Systeme zwar periodisch Daten aus. Ist aber ein System temporär nicht verfügbar, so kann das andere System fehlerfrei weiter arbeiten. Ein Beispiel für eine »weiche Kopplung« ist die ALE-Kopplung (Application Link Enabling).

Den Prozess, der den RFC startet, bezeichnen wir auch als Sender oder Client, den Prozess, in dem der RFC ausgeführt wird, als Empfänger oder Server. Wir unterscheiden vier Arten von RFCs: synchrone, asynchrone, transaktionale und queued RFCs.

- ▶ Ein *synchroner RFC* zeichnet sich dadurch aus, dass der Sender wartet, während der RFC im Empfängerprozess läuft.
- ▶ Bei einem *asynchronen RFC* wartet der Sender nach dem Starten des RFCs nicht, bis der Empfängerprozess den RFC komplett bearbeitet hat. Stattdessen setzt der Sender nach dem Start des RFCs seine Arbeit fort. Damit kann ein Sender parallel mehrere asynchrone RFCs initiieren. (Bei einem synchronen Aufruf kann dagegen immer nur ein RFC laufen, da der Sender auf das Ende der Bearbeitung wartet.)
- ▶ Ein *transaktionaler RFC (tRFC)* ist ein asynchroner RFC, der unter »besonderen Sicherheitsbedingungen« abläuft. Details hierzu finden sich im Verlauf dieses Kapitels.
- ▶ Ein *queued RFC (qRFC)* ist ein transaktionaler RFC, bei dem die Reihenfolge der Bearbeitung im Zielsystem der Erstellung im Quellsystem entspricht (für alle qRFCs in einer bestimmten Queue).

Einen RFC erkennen Sie in einem ABAP-Programm an der Syntax **CALL FUNCTION <Funktionsname> DESTINATION <Verbindungsname> ...** Die Variable **<Verbindungsname>** enthält den Namen der RFC-Verbindung, auch Destination genannt.

Enthält der Funktionsaufruf den Zusatz **STARTING NEW TASK**, so handelt es sich um einen asynchronen RFC; der Zusatz **IN BACKGROUND TASK** führt zu einem transaktionalen RFC. Enthält der Aufruf nur den Zusatz **DESTINATION**, aber weder **STARTING NEW TASK** noch **IN BACKGROUND TASK**, so wird der RFC als synchroner RFC gestartet.

Mit dem folgenden Beispiel-Coding starten Sie den Funktionsbaustein Z\_BC315 RFC synchron, asynchron und transaktional. (**DESTINATION 'NONE'** bedeutet hierbei, dass Ziel- und Quellsystem identisch sind.)

```
* Synchronous RFC
CALL FUNCTION 'Z_BC315 RFC'
  DESTINATION 'NONE'
  EXCEPTIONS
    argument_error = 1
    send_error     = 2
    OTHERS         = 3.

* Asynchronous RFC
CALL FUNCTION 'Z_BC315 RFC'
  STARTING NEW TASK task
  DESTINATION 'NONE'
  EXCEPTIONS
    communication_failure = 1
    system_failure        = 2
    RESOURCE_FAILURE      = 3.

* Transactional RFC
CALL FUNCTION 'Z_BC315 RFC'
  IN BACKGROUND TASK
  DESTINATION 'NONE'.

COMMIT WORK.
```

Alle RFCs, egal ob synchron, asynchron und transaktional, werden in Dialog-Workprozessen gestartet. Lassen Sie sich also durch die Syntax **IN BACKGROUND TASK** nicht täuschen: Diese hat nichts mit Hintergrund-Workprozessen zu tun. Auch transaktionale RFCs werden in Dialog-

Workprozessen ausgeführt. Eine detaillierte Einführung in die Programmierung von RFCs finden Sie in der ABAP-Online-Hilfe zur Anweisung CALL FUNCTION.

### 6.1.2 Ablauf eines RFCs

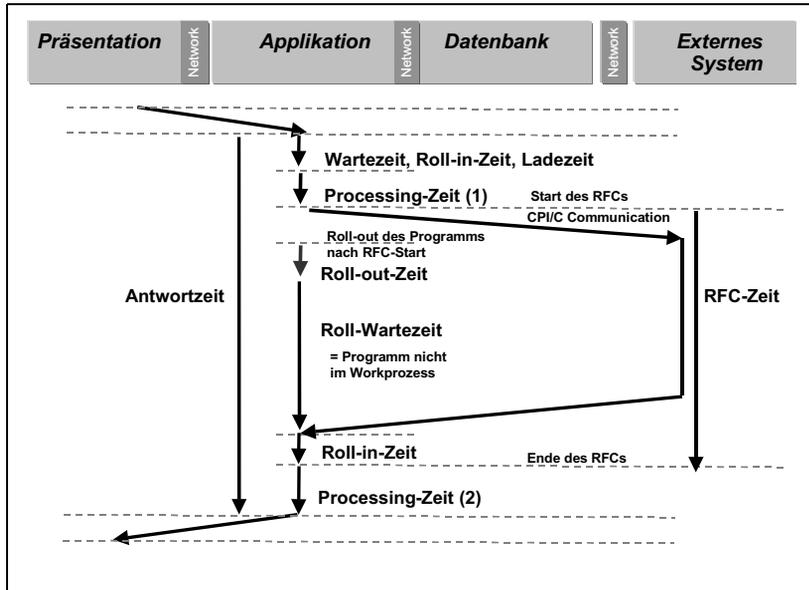


Abbildung 6.1 Ablauf eines Transaktionsschrittes mit synchronem RFC und die dabei gemessenen Zeiten

Wenn ein RFC ausgeführt wird, so läuft folgender Vorgang ab:

1. Zunächst stellt der Sender (d.h. der Workprozess, der den RFC starten will) eine Verbindung zum Empfängersystem her. Diese Verbindung wird über die Gateway-Services der beiden beteiligten Systeme realisiert (Initialisierungsphase). Auf dem Empfängersystem sucht der Dispatcher einen freien Dialog-Workprozess, der den RFC ausführen kann. In der Workprozess-Übersicht (Transaktion SM50 bzw. SM66) steht der Sender-Workprozess in dieser Zeit im Status »hält« mit Grund »CPIC«. Die Spalte **Aktion / Wartegrund** zeigt den Eintrag »CMINIT«.
2. Ist die Verbindung zwischen Sender- und Empfänger-Workprozess aufgebaut, werden die zur Ausführung des RFCs nötigen Daten übertragen. In der Workprozess-Übersicht wird nun sowohl beim Sender- als auch beim Empfänger-Workprozess der Status »hält« mit Grund »CPIC« und **Aktion / Wartegrund** »CMSSEND« angezeigt. Hinter dem

Eintrag CMSEND befindet sich eine Nummer, die so genannte Kommunikations-ID.

3. Im weiteren Ablauf unterscheiden sich der synchrone und der asynchrone RFC. Sind alle nötigen Daten übertragen und ist der RFC auf der Empfängerseite gestartet, so wird beim asynchronen RFC die Verbindung abgebaut, und das Programm auf der Senderseite fährt mit seiner Arbeit fort, ohne das Ende des RFCs abzuwarten. Diese Art der Verarbeitung macht es möglich, dass nun z.B. ein weiterer RFC gestartet wird, der dann parallel zum ersten läuft.
4. Beim synchronen RFC wartet die Senderseite auf das Bearbeitungsende des RFCs. Während der Wartezeit wird der Benutzerkontext auf der Senderseite aus dem Workprozess herausgerollt, so dass der Workprozess anderen Benutzern zur Verfügung steht. Auf der Senderseite ist in der Workprozess-Übersicht von dem laufenden RFC und dem wartenden Programm nichts zu sehen. Das wartende Programm kann auf der Senderseite nur noch in der Benutzerübersicht (Transaktion SM04) als Modus und im Gateway-Monitor (Transaktion SMGW) als offene Kommunikationsverbindung beobachtet werden. Auf der Empfängerseite verursacht der laufende RFC einen Eintrag im Status »läuft« in der Workprozess-Übersicht. In der Benutzerübersicht (Transaktion SM04) kann der von außen kommende RFCs anhand des Terminaleintrages »APPC-TM« identifiziert werden. Ab SAP-Basis 4.6 findet sich in der Benutzerübersicht eine eigene Spalte für den Benutzertyp (Spalte **Type**). Für eine RFC-Verbindung finden Sie den Eintrag »RFC«.
5. Hat die Empfängerseite die Bearbeitung des synchronen RFCs abgeschlossen, so muss der wartende Kontext auf der Senderseite wieder »geweckt«, d.h. in einen Workprozess hineingerollt werden. Bei der anschließenden Datenübertragung vom Empfänger zum Sender stehen beide Workprozesse in der Workprozess-Übersicht im Status »hält« mit Grund »CPIC« und **Aktion / Wartegrund** »CMRECEIVE«.
6. Schließlich wird die Verbindung abgebaut, der Empfänger-Workprozess wird wieder frei und der Sender-Workprozess setzt seine Arbeit fort.

Bei diesem Vorgang sollte der Aufbau der Verbindung (d.h. die CMINIT-Phase) nur wenige Millisekunden dauern. Finden Sie also in der Workprozess-Übersicht häufig diesen Zustand, so kann es an einer Überlastung des Empfängersystems liegen (Details zu diesem Punkt finden Sie in

einem folgenden Abschnitt). Die Dauer der CMSEND- und CMRECEIVE-Phasen hängt stark von der übertragenen Datenmenge und der Netzgeschwindigkeit ab. Für diese Phasen lässt sich keine Zeitangabe machen.

Abbildung 6.1 zeigt schematisch den Ablauf eines synchronen RFCs und die dabei gemessenen Zeiten, die in der Einzelsatzstatistik (Transaktionscode STAD bzw. STAT) und im Workload-Monitor (Transaktionscode ST03 bzw. ST03N) angezeigt werden.

**Roll-Wartezeit** Während das sendende Programm auf die Antwort des synchronen RFCs wartet, wird der Benutzerkontext aus dem Workprozess herausgerollt, so dass der Workprozess anderen Benutzern zur Verfügung steht. Nach Beendigung des RFCs im Empfängersystem erfolgt der erneute Roll-in in einen Workprozess, und der Transaktionsschritt wird fortgesetzt. Während eines RFC-Aufrufs läuft für das rufende Programm die Antwortzeit weiter. Bei der Analyse der »fehlenden Zeiten« (vgl. den Abschnitt 3.2, »Workload-Analyse«) ist also die *Roll-Wartezeit* von der Gesamtantwortzeit zu subtrahieren, da während dieser Zeit in dem rufenden System keine CPU-Ressourcen benötigt werden. In einem statistischen Einzelsatz zu einem Transaktionsschritt mit einem synchronen RFC findet man also eine Roll-Wartezeit. (Bei sehr kurzen RFCs mit einer Antwortzeit von weniger als 500 Mikrosekunden findet allerdings kein Roll-out statt.) Bei einem asynchronen RFC finden Sie keine Roll-Wartezeit (da bei einem asynchronen RFC das Programm nicht herausgerollt wird, sondern seine Arbeit sofort fortsetzt).

**RFC-Zeit** Im Feld **RFC+CPIC Time** der Einzelsatzstatistik und des Workload-Monitors finden Sie die Gesamtzeit des RFC-Aufrufs. Bei einem asynchronen RFC umfasst dies die Phasen 1 bis 3 der obigen Aufzählung. Als Faustregel sollte die *RFC-Zeit* hier nicht mehr als 50 ms pro Aufruf dauern. Beim synchronen RFC umfasst die RFC-Zeit die Phasen 1 bis 6. Damit ist klar, dass die RFC-Zeit hier größer als die Roll-Wartezeit sein muss.

**SAP R/3 3.1** Im Workload-Monitor für SAP R/3 3.1 finden Sie keine Informationen über die RFC- und Roll-Wartezeit. Die mittleren Antwortzeiten für RFC-Aufrufe im gerufenen System finden Sie im Transaktionsprofil (Schaltfläche **Transaction Profile**) als Programm CPIC/RFC.

## 6.2 Schnittstellen zu externen Systemen

In diesem Abschnitt diskutieren wir die Performanceaspekte von Schnittstellen zwischen SAP-Systemen bzw. zwischen SAP-Systemen und externen Systemen.

## 6.2.1 RFC-Verbindungen konfigurieren und testen

RFC-Verbindungen (auch *Destinationen* genannt) richten Sie in der Transaktion **Anzeige und Pflege von RFC-Destinationen** (Transaktionscode SM59) ein. Über das Menü erreichen Sie diese Transaktion wie folgt:

**Werkzeuge · Administration · Administration · Netzwerk · RFC-Verbindungen**

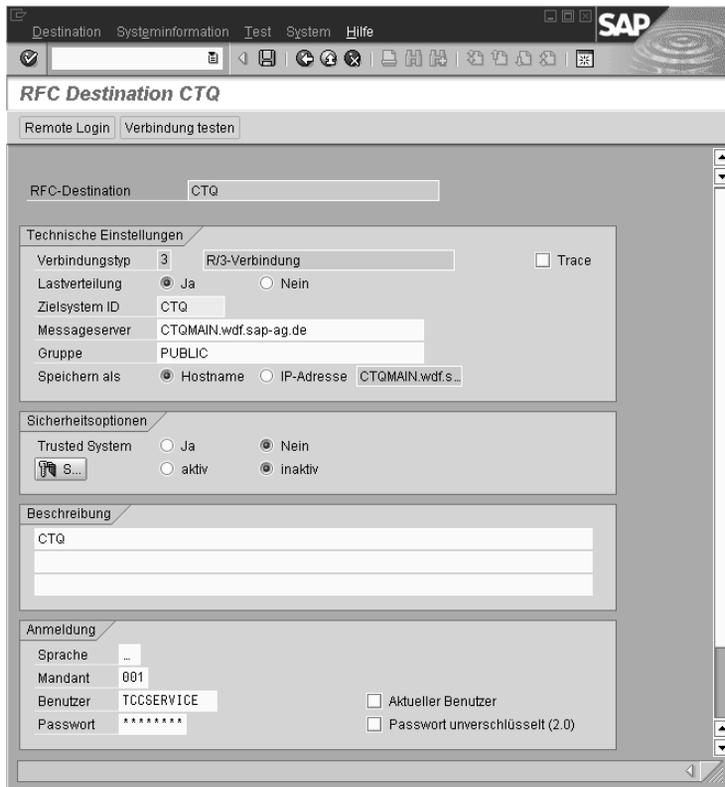
In dieser Transaktion werden zunächst alle verfügbaren RFC-Verbindungen in einer Baumstruktur dargestellt. Tabelle 6.1 beschreibt die vier möglichen Typen von RFC-Verbindungen.

Verbindungstyp	Beschreibung
Interne Verbindung	RFC-Verbindungen zu allen SAP-Applikationsinstanzen des eigenen SAP-Systems. Diese Verbindungen werden bei der Installation Ihres SAP-Systems automatisch in der Form <Rechnername>_<SAP-Systemname>_<Instanznummer> generiert. Hier finden Sie auch die Verbindung »NONE«, die immer auf die aktuelle Instanz zeigt.
R/3-Verbindungen	RFC-Verbindungen zu anderen SAP-Systemen, u.a. auch Verbindungen der Form TMSADM@<System 1>.DOMAIN_<System 2>: Diese werden vom Transport Management System (TMS) benötigt und von diesem beim Aufbau generiert.
TCP/IP-Verbindungen	An dieser Stelle existieren Verbindungen zu Nicht-SAP-Systemen. Zahlreiche Standarddestinationen sind hier bereits vorkonfiguriert.
Verbindungen über ABAP/4-Treiber	Hier nicht von Interesse

**Tabelle 6.1** RFC-Verbindungstypen

Wählen Sie eine Verbindung mit einem Doppelklick aus, gelangen Sie auf einen Bildschirm mit Details zu dieser Verbindung. Dieser Bildschirm hat je nach Verbindungstyp ein unterschiedliches Aussehen. Abbildung 6.2 zeigt ein Beispiel.

Neben den Details zur Konfiguration der RFC-Verbindung finden Sie in der Schaltflächenleiste die Funktion **Verbindung testen**. Diese Funktion sollten Sie zunächst ausführen, wenn Sie der Meinung sind, Probleme mit einer Verbindung zu haben. Sie meldet sich am Empfängersystem an und überträgt einige Testdaten. Als Ergebnis wird Ihnen die Antwortzeit für die Datenübertragung dargestellt bzw. eine Fehlermeldung, wenn die Anmeldung nicht erfolgreich war.



**Abbildung 6.2** Konfiguration einer externen RFC-Verbindung (Transaktionscode SM59)

Die Antwortzeit für die Testdatenübertragung hängt von dem verwendeten Netz ab. Optimale Antwortzeiten liegen im Bereich von 10 bis 100 Millisekunden. Indem Sie den Verbindungstest zu Zeiten mit hoher und niedriger Systemlast durchführen, ermitteln Sie, ob die Verbindung zu Hochlastzeiten überlastet ist. Hohe Antwortzeiten können drei Ursachen haben:

- ▶ Die Netzwerkverbindung ist prinzipiell langsam oder überlastet.
- ▶ Das Empfängersystem ist physisch überlastet.
- ▶ Auf dem Empfängersystem sind zu wenig Prozesse konfiguriert, die RFCs empfangen können.

Verbindungstyp	Ergebnis des Verbindungstests	Maßnahme
R/3	Fehlermeldung »TCP/IP Timeout«	Das Empfängersystem kann nicht angesprochen werden. Überprüfen Sie die Verfügbarkeit des Empfängersystems sowie die Netzwerkverbindung
	Hohe Antwortzeiten (> 100 ms) bei einer Verbindung zu einem SAP-System	<ol style="list-style-type: none"> <li>Überprüfen Sie die Netzwerkleitung zu dem externen System mit Hilfe des Ping-Befehls oder mit den Werkzeugen des Netzanbieters. Optimale Ping-Anwortzeiten liegen im Bereich von einigen zehn Millisekunden.</li> <li>Anmelden und Performanceuntersuchung auf dem Empfängersystem. Sind alle Workprozesse belegt?</li> </ol>
TCP/IP	Fehlermeldung »Partnerprogramm nicht registriert«	Starten Sie den entsprechenden Empfängerprozess.
	Fehlermeldung »TCP/IP Timeout«	Überprüfen Sie die Netzverbindung zwischen betroffenem SAP-Gateway-Prozess und der Gegenseite. Informationen zum Gateway können über die Schaltfläche <b>Gateway</b> eingesehen werden, wenn das Gateway explizit gepflegt wurde. Falls dieses nicht der Fall ist, wird immer das lokale Gateway des SAP-Applikationsservers verwendet. Falls Sie noch nachträglich Gateway-Einstellungen ändern wollen, kann dies über das Menü <b>Destination • Gateway-Optionen</b> erfolgen. (Dieses erfolgt immer auf beiden Seiten.)
	Hohe Antwortzeiten (> 100 ms) bei einer Verbindung zu einem externen System	<ol style="list-style-type: none"> <li>Überprüfen Sie die Netzwerkleitung zu dem externen System mit Hilfe des Ping-Befehls oder mit den Werkzeugen des Netzanbieters. Optimale Ping-Anwortzeiten liegen im Bereich von einigen zehn Millisekunden.</li> <li>Überprüfen Sie, wie viele parallele Empfangsprozesse auf der Gegenseite zum Empfang der IDOCs gestartet wurden. Es kann sinnvoll sein, bei hohem Lastaufkommen auf der Empfangsseite die Anzahl der Prozesse zu erhöhen.</li> <li>Bitte beachten Sie die SAP-Hinweise 63930 und 44844 zu RFC-Verbindungen mit Gateway-Registrierung.</li> </ol>

**Tabelle 6.2** Ursachen für Probleme beim Verbindungsaufbau

Abhängig vom Empfängersystem können für die Verbindung unterschiedliche Details konfiguriert werden. Ist das Empfängersystem ein SAP-System, so ist zunächst zu entscheiden, ob die Anmeldung an das Empfängersystem über Logon-Gruppen oder direkt über einen festgelegten Applikationsserver erfolgen soll. Diese Auswahl erfolgt durch Setzen der

**SAP-System als Empfänger**

Option **Lastverteilung**. Wird die Lastverteilung aktiviert, so sind der Message-Server und die entsprechende Logon-Gruppe des Empfangssystems anzugeben; wird die Verbindung ohne Lastverteilung eingerichtet, so werden die Eingaben von Applikationsserver (Feld **Zielmaschine**) und Instanznummer (Feld **Systemnummer**) angefordert.



Bei der Auswahl sollten Sie sich für die Anmeldung über eine Logon-Gruppe entscheiden. Dieses Verfahren hat gegenüber der direkten Anmeldung an einen dezidierten Applikationsserver zwei Vorteile:

- ▶ Einer Logon-Gruppe können mehrere Applikationsserver zugeordnet werden. Fällt ein Server aus, so kann ein anderer die ankommenden RFCs verarbeiten (Aspekt der Hochverfügbarkeit).
- ▶ Es findet ein Logon-Balancing, d.h. eine Lastverteilung zwischen den Instanzen, statt, das eine Überlastung verhindert (Aspekt der Performanceoptimierung).

**Kontinuierliche  
Überwachung  
von RFC-  
Verbindungen**

Mit dem zentralen CCMS-Überwachungsmonitor (Transaktionscode RZ20) können Sie die Erreichbarkeit wichtiger RFC-Verbindungen kontinuierlich überwachen. Detaillierte Informationen, wie Sie diese Überwachung aktivieren, finden Sie im Dokument »Monitoring RFC Destinations and Services« im SAP Service Marketplace unter dem Alias */systemmanagement*. Folgen Sie dort den Links *System Monitoring and Alert Management, Media Library*.

## **6.2.2 Eingehende und ausgehende Last überwachen**

Die über RFC ein- und ausgehende Last überwachen Sie im RFC-Profil des Workload-Monitors. Dieses finden Sie in SAP-Basis 4.6 (ab Basis Support-Package 22) im neuen Workload-Monitor (Transaktionscode ST03N), ansonsten im alten Workload-Monitor (Transaktionscode ST03) unter:

**Goto • Profiles • RFC Profile**

Die Profile unter den Menüpunkten **Clients** und **Client Destinations** entsprechen dabei den ausgehenden RFCs (das lokale System ist Sender bzw. Client); unter **Server** bzw. **Server Destinations** finden Sie die Last durch einkommende RFCs (das lokale System ist Empfänger bzw. Server). Tabelle 6.3, Tabelle 6.4 und Tabelle 6.5 stellen die unterschiedlichen RFC-Profile (**Client Destinations**, **Server Destinations**, **Clients** und **Server**), die in den RFC-Profilen angegebenen Performanceindikatoren und die unterschiedlichen Sichten dar.

Profil	Beschreibung
Client Destinations und Server Destinations	Die Einzelsatzstatistik schreibt zu jedem Transaktionsschritt Summenstatistiksätze mit den Daten über alle ausgeführten pro RFC-Destination. In diesen Profilen finden Sie die gesamte durch RFCs erzeugte Last. Achtung: Die Angabe des Funktionsbausteinennamens und die entsprechende Sicht (siehe Tabelle 6.5) ist bedeutungslos (und wird im neuen Workload-Monitor auch nicht mehr ausgegeben).
Clients und Server	Die Einzelsatzstatistik schreibt für die fünf (änderbar über den Parameter <b>stat/rfcrec</b> ) teuersten Funktionsbausteine eines Transaktionsschritts die Funktionsbausteinennamen in den statistischen Satz. In diesen Profilen finden Sie die Last dieser jeweils fünf teuersten Funktionsbausteine. Diese Statistiken eignen sich insbesondere für einen Einstieg in die Anwendungsanalyse, da Sie in diesen Statistiken die Funktionen mit der höchsten Last identifizieren können.

**Tabelle 6.3** RFC-Profile

Feld	Bedeutung
Quantity	Anzahl der Remote Function Calls (RFCs)
Call time	Antwortzeit für die RFCs, gemessen im Sendersystem. Die Differenz zwischen »Call time« und »Execution time« ist die Zeit für Verbindungsaufbau und Datenübertragung zwischen Sender- und Empfängersystem. Als Fausregel sollte diese Zeit nicht größer als 20% der »Call time« sein. Wird dieser Wert überschritten, so sollten Sie nach einem Bottleneck in der Verbindung zwischen Sender und Empfänger suchen.
Execution time	Antwortzeit für die RFCs, gemessen im Empfängersystem. Die »Execution time« ist die Nettozeit für die Ausführung der RFCs im Empfängersystem.
Sent Data	Gesendete Datenmenge
Received Data	Empfangene Datenmenge

**Tabelle 6.4** Felder des RFC-Profiles im Workloadmonitor (Transaktionscode ST03/ST03N)

Sicht	Schaltfläche	Bemerkung
Transaktionscodes	Transaction code	
Benutzernamen	User names	Diese Sicht ist nützlich, wenn ein Benutzer charakteristisch für einen bestimmten Übertragungskanal ist (z. B. könnten Sie Ihre RFC-Verbindung so eingerichtet haben, dass Vertriebsaufträge über einen Benutzer RFC_VERTRIEB in Ihr System gesendet werden).

**Tabelle 6.5** Sichten im RFC-Profil des Workload-Monitors

Sicht	Schaltfläche	Bemerkung
RFC-Funktionsbausteine	Function modules	Diese Sicht eignet sich am besten für einen Einstieg in die Anwendungsanalyse, da Sie in dieser Sicht die Funktionen mit der höchsten Last identifizieren. Diese Sicht ist in den Profilen <b>Client Destinations</b> und <b>Server Destinations</b> bedeutungslos.
Externe RFC-Destinationen	Remote destinations	Als Remote-Destinationen sind z. B. SAP-Instanzen entfernter Systeme angegeben oder auch Rechnernamen, auf denen Programme laufen, mit denen das lokale SAP-System über RFC kommuniziert.
Lokale RFC-Destinationen	Local destinations	Verteilung der RFC-Last auf die Instanzen des lokalen SAP-Systems.

**Tabelle 6.5** Sichten im RFC-Profil des Workload-Monitors (Forts.)

Gehen Sie bei der Auswertung der RFC-Profile wie folgt vor:

1. Sortieren Sie das RFC-Profil nach der Spalte **Call time**. Damit stehen die Funktionen bzw. Destinationen mit der längsten Bearbeitungszeit an der Spitze der Liste. (Hinweis: In einigen SAP-Kernelversionen werden bei Softwareimporten, (Programm  $\tau_p$ ) falsche Zeiten für die »Call time« ermittelt, so dass für den Rechner, von dem aus der Softwareimport gestartet wurde, unsinnig hohe Zeiten angegeben werden. Diese können aber im Profil sofort erkannt werden.)
2. Vergleichen Sie die Werte **Call time** und **Execution time**. Als Faustregel gilt, dass die Differenz nicht größer als 20% sein sollte. Indem Sie diesen Vergleich in den unterschiedlichen Sichten durchführen, identifizieren Sie solche Destinationen, Funktionsbausteine bzw. Benutzer, bei denen ein mögliches Problem in der Kommunikation zwischen Sender und Empfänger existiert. Analysieren Sie den Verbindungsaufbau und die Datenübertragung, indem Sie den entsprechenden Verbindungsaufbau testen und für das dazugehörige Programm einen RFC-Trace erstellen.
3. Weiterhin können Sie das RFC-Profil nach **Execution time**, **Sent Data** und **Received Data** sortieren, um Funktionsbausteine mit hoher Last zu identifizieren.

**SAP Early Watch Alert**

Im SAP EarlyWatch Alert-Bericht finden Sie im Abschnitt »Interface Load« wöchentlich eine Übersicht über die wichtigsten Destinationen und die wichtigsten Funktionsbausteine, die als RFC aufgerufen werden. Die Tabellen im SAP EarlyWatch Alert-Bericht sind eine komprimierte Darstellung aus den beschriebenen RFC-Statistiken.

Um die statistischen Informationen über RFCs im Workload-Monitor (Transaktionscode ST03N) und auch in den statistischen Einzelsätzen (Transaktionscode STAD) interpretieren zu können, muss man verstanden haben, nach welchen Regeln die Statistiken erstellt werden. Ansonsten kann man leicht Fehlschlüssen bei der Interpretation aufsitzen. Deshalb stellen wir im Folgenden dar, wie die RFC-Statistiken erstellt werden und was bei der Interpretation zu beachten ist.

Nehmen wir an, dass in einem Transaktionsschritt RFCs an mehrere Destinationen gesendet werden. Die Destinationen bezeichnen wir als A, B, C etc. Die RFCs, die an die Destination A gesendet werden, bezeichnen wir mit A1, A2, A3 etc., die an B gesendeten RFCs mit B1, B2 etc. Aus Performancegründen ist es nicht möglich, für alle RFCs Statistiken zu schreiben. Die Statistiken werden wie folgt erstellt:

- ▶ Auf der Senderseite (Client-Seite) werden für die fünf teuersten RFCs (innerhalb eines Transaktionsschrittes) detaillierte statistische Informationen gespeichert. Zu diesen Informationen gehören u.a. Benutzer, Destination, Sender- und Empfänger-Instanz, Name des Funktionsbausteins, Aufrufzeit und Ausführungszeit und die übertragene Datenmenge. Diese Statistiken bezeichnet man als *Client-Statistiksätze*. Aus diesen Statistiksätzen wird das RFC-Client-Profil erstellt.
- ▶ Ebenfalls auf der Senderseite werden für die fünf teuersten Destinationen detaillierte statistische Informationen gespeichert, die dann die Summe über alle RFCs enthalten, die an diese Destinationen gesendet werden. Zu diesen Informationen gehören u.a. Benutzer, Destination, Sender- und Empfänger-Instanz, Anzahl der Aufrufe, Aufrufzeit, Ausführungszeit und die übertragene Datenmenge. Ein Name eines Funktionsbausteins wird hier natürlich nicht mitgeschrieben, da die Statistik die Summe über alle RFCs einer Destination ist, also über alle Funktionsbausteine summiert ist. Diese Statistiken bezeichnet man als *Client-Destination-Statistiksätze*. Aus diesen Statistiksätzen wird das RFC-Client-Destination-Profil erstellt.
- ▶ Auf der Empfängerseite (Server-Seite) werden die entsprechenden Statistiken erstellt, die als *Server-* bzw. als *Server-Destination-Statistiksätze* bezeichnet werden.

Aus den bisher beschriebenen Regeln für das Erstellen der Statistiksätze ergibt sich, dass die RFC-Profile im Workload-Monitor nicht die gesamte RFC-Last erfassen, da Statistiken nur jeweils für die teuersten fünf Destinationen und die fünf teuersten Aufrufe (jeweils pro Transaktionsschritt) erfasst werden. Allerdings können Sie die Grenze, wie viele Destinationen und Aufrufe erfasst werden sollen, mit dem Profilparameter **stat/rfcrec**

verändern. Wenn Sie diesen Parameter nicht ändern, hat er den Wert 5. In der Praxis zeigt sich jedoch, dass mit der Standardeinstellung 5 bereits 50% bis 90% der RFC-Last erfasst werden. Dieser Wert hängt von der Größe des Systems ab. Die Erhöhung des Parameters **stat/rfcrec** führt zu einer erhöhten Last für den Statistikkollektor und kann damit auch zu Performanceproblemen führen. Sie sollten daher im Normalfall die Standardeinstellung nicht verändern.

#### Interpretation der RFC-Server-Statistiken

Bei der Analyse von RFC-Statistiken auf dem Empfänger- bzw. Server-System wählen Sie im Workload-Monitor (Transaktionscode ST03N) bzw. in der Einzelsatzanalyse (Transaktionscode STAD) den Tasktyp »RFC« aus, um die Last zu analysieren, die von einkommenden RFCs erzeugt wird. Die weitere Analyse bzgl. Anteilen von CPU-Zeit, Datenbankantwortzeit etc. kann dann prinzipiell so erfolgen, wie bei Transaktionsschritten im Dialog-Task. Allerdings gibt es zwei Besonderheiten zu berücksichtigen, die gelegentlich zu Verwirrung führen.

Die erste Besonderheit tritt auf, wenn ein RFC wiederum einen RFC aufruft. In diesem Fall wird die Zeit, die der zweite RFC für seine Ausführung benötigt, nicht dem ersten RFC zugerechnet, d.h. seine Antwortzeit umfasst nur die Zeit, die er selbst im Workprozess verbringt. Die Roll-Wait-Zeit, d.h. die Zeit, die er wartet, dass der zweite RFC seine Arbeit beendet, ist nicht in der Antwortzeit enthalten. Diese Festlegung wurde getroffen, um »Mehrfachzählungen« von Zeiten bei RFCs zu vermeiden. (Bitte beachten Sie: Bei Transaktionsschritten im Dialog-Task ist die Zählung anders. Wird hier ein RFC aus dem Dialog-Task gestartet, umfasst die Antwortzeit sehr wohl die Zeit für die Ausführung des RFCs, auch wenn der Workprozess herausrollt. Diese Festlegung wurde getroffen, da die Antwortzeit im Dialog-Task natürlich die Zeit widerspiegeln soll, die ein Benutzer vor seinem Bildschirm wartet.)

Eine zweite Besonderheit tritt auf, wenn der Sender seine Verbindung zum Empfänger nicht schließt. Beschreiben wir diesen Fall anhand eines Beispiels: Nehmen wir an, dass ein Sender einen RFC in einem Empfängersystem (einem SAP-System) ausführt und nach der Ausführung die Verbindung aus Performancegründen nicht wieder schließt. Dieses Verfahren wird oft angewendet, wenn viele RFCs hintereinander ausgeführt werden, denn jeder Verbindungsaufbau ist mit einem Performancenachteil verbunden. Nehmen wir in unserem Beispiel weiterhin an, dass jeder RFC sehr schnell ist, d.h. dass jede Ausführung z.B. 20 ms benötigt. Wie verhält sich das Empfängersystem in diesem Fall? Nach der Ausführung des ersten RFCs wartet der RFC-Server-Workprozess 500 ms. Ruft der

Sender vor Ablauf dieser Wartezeit das Empfängersystem erneut, wird der RFC im selben Workprozess ausgeführt. Das Roll-in und Roll-out des Benutzerkontextes wird somit vermieden. Wird innerhalb der 500 ms kein weiterer RFC ausgeführt, wird der Benutzerkontext aus dem Workprozess herausgerollt und dieser steht anderen Benutzern für Anfragen zur Verfügung. (Der Benutzerkontext und die CPIC-Verbindung bleiben dabei weiterhin erhalten, sie werden nur aus dem Workprozess herausgerollt.) Beim Herausrollen wird ein Statistiksatz erstellt. Die Antwortzeit, die dieser Statistiksatz anzeigt, ist die Zeit, die der Workprozess belegt war. Nehmen wir ein Beispiel, in dem vier RFCs mit einer Antwortzeit von je 20 ms schnell hintereinander ausgeführt werden. Anschließend wartet der Workprozess noch 500 ms. Die Antwortzeit für den RFC-Aufruf wird damit mit ca. 600 ms angegeben. Die Antwortzeiten der vier RFCs finden Sie in den Details des Statistiksatzes.

Zusammenfassend lässt sich sagen: Die Antwortzeit im Dialog-Task gibt die Zeit wieder, die ein Benutzer auf seine Antwort wartet. Dies umfasst sowohl die Zeit, in der die Anfrage im Workprozess bearbeitet wurde, als auch die Zeit, die die Anfrage eventuell auf die Antwort eines externen RFCs gewartet hat. Die Antwortzeit im RFC-Task gibt dagegen die Zeit wieder, in der ein RFC einen Workprozess belegt hat.

### **6.2.3 Parallelisierung von Prozessen mit asynchronen RFCs konfigurieren**

Wie wir bereits gesehen haben, werden *asynchrone RFCs* dazu benutzt, Anwendungen zu parallelisieren, da die Senderseite nach dem Senden des RFCs nicht auf dessen Beendigung warten muss, sondern bereits weitere RFCs versenden kann. Wird der Grad der Parallelisierung nicht begrenzt, so kann es zu einem lawinenartigen Anwachsen der aRFCs kommen, die die Applikationsebene auf der Empfängerseite lahm legen können (alle Workprozesse belegt).

Es gibt zwei Wege, solche Situationen zu vermeiden: Die Möglichkeit, eigene SAP-Instanzen mit einer dezidierten Logon-Gruppe einzurichten, wurde bereits in Abschnitt 5.2.1, »Dynamische Benutzerverteilung konfigurieren«, vorgestellt. Darüber hinaus können für jede SAP-Instanz die Ressourcen für RFC-Verarbeitungen beschränkt werden. So kann für Dialoganwendungen ein gewisser Anteil der Ressourcen reserviert werden. Bei asynchronen RFCs (Sprachelement **STARTING NEW TASK <Task-name> DESTINATION IN GROUP <Group>**) wird die Lastinformation auf der Empfängerseite ausgewertet. An jede Applikationsinstanz werden so

viele aRFCs gesendet, wie sie Ressourcen zur Verfügung stellt. Wenn eine Applikationsinstanz nicht mehr über freie Ressourcen verfügt, dann werden an diese keine aRFCs mehr gesendet. Die Abfrage der Ressourcen wird so lange wiederholt, bis alle aRFCs abgearbeitet sind.

Die Ressourcen der aRFC-Bearbeitung werden über Profilparameter vorgegeben. Mithilfe des Programms **RSARFCLD** können die Quoten dynamisch konfiguriert werden. Weitere Informationen über die Profilparameter finden Sie in Anhang C, »Performancerelevante Konfigurationsparameter«, und in SAP-Hinweis 74141.

### **6.2.4 Datenübertragung mit transaktionalen RFCs überwachen**

Ein transaktionaler RFC ist ein asynchroner RFC, der unter »besonderen Sicherheitsbedingungen« abläuft. Transaktionale RFCs (tRFCs) werden nicht sofort ausgeführt, sondern die Aufrufe werden zunächst in einer internen Tabelle gesammelt. Bei der darauf folgenden **COMMIT WORK**-Anweisung werden alle Aufrufe in der Reihenfolge ihres Aufrufs abgearbeitet. Wurden vor dem **COMMIT WORK** auch Verbuchungssätze erzeugt, so werden die transaktionalen RFCs erst dann ausgeführt, wenn die Verbuchungsbausteine fehlerfrei abgearbeitet werden konnten. Die transaktionalen RFCs einer Transaktion bilden pro Destination eine Logical Unit of Work (LUW).

Alle tRFC-Aufrufe werden auf der Senderseite in den Tabellen ARFCS-STATE und ARFCSDATA aufgezeichnet. Jede LUW ist durch eine weltweit eindeutige ID gekennzeichnet. Beim **COMMIT WORK** werden die unter dieser ID aufgezeichneten Aufrufe auf dem entsprechenden Zielsystem ausgeführt. Tritt bei einem der Aufrufe ein Fehler auf, werden alle von den vorhergehenden Aufrufen durchgeführten Datenbankoperationen (**ROLLBACK**) zurückgenommen und eine entsprechende Fehlermeldung in die Tabelle ARFCSSTATE geschrieben. Konnte eine LUW erfolgreich auf dem Zielsystem ausgeführt werden, wird dieser Erfolg auf dem Zielsystem quittiert. Die entsprechenden Einträge in den Tabellen ARFCSSTATE und ARFCSDATA werden auf der Senderseite gelöscht. Die Fehlermeldungen zu tRFCs können mit Hilfe der Transaktion SM58 ausgewertet werden.



Abgebrochene oder nicht ausgeführte tRFCs führen dazu, dass Inkonsistenzen zwischen Systemen entstehen und beeinträchtigen damit massiv den Geschäftsprozess. Die tägliche Kontrolle der Fehler bei der Schnittstellenverarbeitung (Transaktionscode SM58) ist daher eine wichtige Aufgabe des SAP-Systemadministrators.

Konnte das Zielsystem nicht erreicht werden, weil z. B. die Verbindung zurzeit nicht aktiv ist, wird standardmäßig für jeden nicht erfolgreichen tRFC ein Hintergrundjob eingeplant, um in regelmäßigen Abständen den tRFC erneut zu starten (Report `RSARFCSE` mit der ID des tRFCs als Parameter).

Werden über eine bestimmte Verbindung sehr viele tRFCs übertragen und wird diese Verbindung unterbrochen, so kann es passieren, dass das sendende SAP-System von Hintergrundjobs »überflutet« wird, die immer wieder erfolglos versuchen, tRFCs an das nicht erreichbare System zu senden. Dies kann zu erheblichen Performanceproblemen im Sendersystem führen, wenn das Empfängersystem über einen gewissen Zeitraum nicht verfügbar ist. Daher empfehlen wir, für alle Verbindungen, über die mehr als 50 tRFCs täglich übertragen werden, das Einplanen der Hintergrundjobs im Fehlerfall zu deaktivieren. Wählen Sie dazu in der Transaktion `SM59` die entsprechende Destination aus. Wählen Sie im Menü **Destination • TRFC-Optionen**. Aktivieren Sie folgenden Bildschirm die Option **Batchjob unterdrücken bei Kommunikationsfehler** und bestätigen Sie Ihre Eingabe mit der Funktion **Weiter**. Es werden nun im Fehlerfall keine Hintergrundjobs pro tRFC angelegt und verarbeitet. Statt dieser einzelnen Jobs müssen Sie nun den Report `RSARFCSE` regelmäßig (ca. alle 30 Minuten) mit entsprechenden Destinationen und dem aktuellen Datum (expliziter Parameter für Variante) als Variante einplanen. Dieses Programm sucht nun sequenziell in der Tabelle `ARFCSSTATE` nach noch nicht versendeten tRFCs und versucht diese zu versenden. Diese Umstellung sollte nur von einem erfahrenen Systembetreuer durchgeführt und getestet werden.

Treten beim Versenden von tRFCs Fehler auf, so werden die entsprechenden Einträge in den Tabellen `ARFCSSTATE` und `ARFCSDATA` nicht gelöscht. Daher müssen diese Tabellen in regelmäßigen Abständen reorganisiert werden. Planen Sie zu diesem Zweck den Report `RSARFC01` mindestens einmal pro Woche als Hintergrundjob ein. Andernfalls können auf den Tabellen `ARFCSSTATE` und `ARFCSDATA` Performanceprobleme auftreten. Weitere Hinweise im Zusammenhang mit nicht gelöschten tRFC-Protokolleinträgen gibt SAP-Hinweis 375566, »Sehr viele Einträge in den tRFC- und qRFC-Tabellen«.

## 6.3 Zusammenfassung

Schnittstellen sind zentraler Bestandteil der mySAP-Technologie, Remote Function Calls (RFCs) wiederum stellen die wichtigste Schnittstellentechnik dar.

Performanceproblemen im Bereich der RFC-Schnittstellen kann proaktiv durch eine geeignete Konfiguration der RFC-Verbindung und der Empfängerinstanz vorgebeugt werden. Sollte es dennoch zu Performanceproblemen kommen, so sind u.a. die Workprozess-Übersicht, die RFC-Profile im Workload-Monitor, die Einzelsatzstatistik, der Performance-Trace (RFC-Trace) sowie die Transaktion zur Konfiguration von RFC-Verbindungen geeignete Analysewerkzeuge.

### **Wichtige Begriffe**

Mit folgenden Begriffen sollten Sie nach dem Studium dieses Kapitels vertraut sein:

- ▶ Remote Function Call: RFC, aRFC, tRFC, qRFC
- ▶ Roll-Wartezeit, RFC-Zeit
- ▶ RFC-Profil im Workload-Monitor

### **Fragen**

Fragen finden Sie am Ende des Kapitels 7, »SAP GUI und Internetanbindung«.