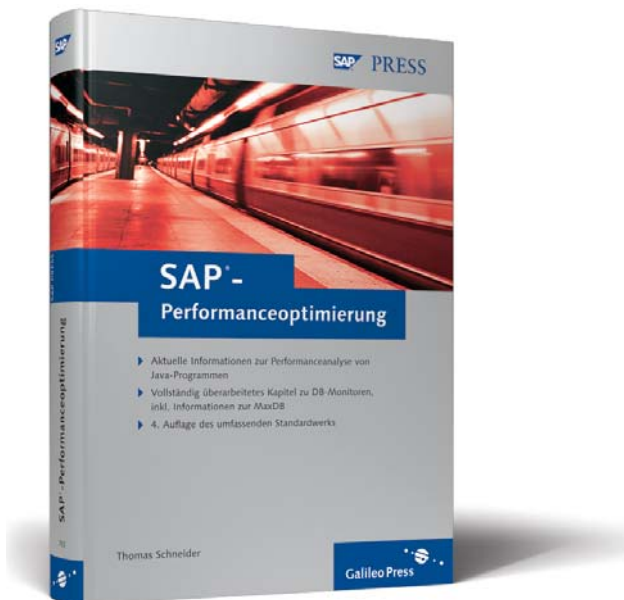


Thomas Schneider

SAP-Performanceoptimierung

Analyse und Tuning von SAP-Systemen



Galileo Press 

Inhalt

Geleitwort zur Buchreihe	13
Danksagung	15
Einleitung	17
Proaktives Performancemanagement	17
Von SAP R/3 zur mySAP Business Suite	20
Zu diesem Buch	23
1 Performancemanagement einer mySAP-Lösung	29
1.1 Die Architektur von mySAP-Lösungen	29
1.1.1 mySAP-Lösungen und -Komponenten	29
1.1.2 Client/Server-Architektur	33
1.2 Das Überwachungs- und Optimierungskonzept für eine mySAP-Lösung	38
1.2.1 Anforderungen an ein Überwachungs- und Optimierungskonzept	38
1.2.2 Werkzeuge und Methoden für das Überwachungs- und Optimierungskonzept	42
1.2.3 Service Level Management	45
1.2.4 Kontinuierliche Überwachung	51
1.2.5 Das Konzept für eine kontinuierliche Performanceoptimierung	62
1.2.6 SAP Solution Manager	67
1.3 Zusammenfassung	71
2 Analyse von Hardware, Datenbank und SAP-Basis	75
2.1 Begriffsklärungen	75
2.2 Hardwareanalyse	76
2.2.1 Analyse eines Hardwareengpasses (CPU und Hauptspeicher) ...	77
2.2.2 Identifizierung von Schreib/Lese-(I/O-)Problemen	82

2.2.3	Weitere Analysen auf Betriebssystemebene	84
2.2.4	Zusammenfassung	84
2.3	Datenbankanalyse	86
2.3.1	Analyse der Datenbankpuffer	86
2.3.2	Identifizierung teurer SQL-Anweisungen	92
2.3.3	Identifizierung von Schreib/Lese-(I/O-)Problemen	98
2.3.4	Weitere Analysen auf Datenbankebene	99
2.3.5	Zusammenfassung	105
2.4	Analyse der SAP-Speicherkonfiguration	106
2.4.1	Analyse der SAP-Puffer	107
2.4.2	Analyse des SAP-Extended Memory, des SAP-Heap Memory und des SAP-Roll Memory	109
2.4.3	Anzeige des allokierten Speichers	111
2.4.4	Weitere Monitore zur detaillierten Analyse	113
2.4.5	Zusammenfassung	113
2.5	Analyse der SAP-Workprozesse	115
2.5.1	Felder der Workprozessübersicht	116
2.5.2	Analyse der Workprozesse	118
2.5.3	Globale Workprozessübersicht	121
2.5.4	Überwachung der Dispatcher-Queue	123
2.5.5	Zusammenfassung	123
2.6	Zusammenfassung	125

3 Workload-Analyse 129

3.1	Der Workload-Monitor	130
3.1.1	Funktionalität und Verfügbarkeit	130
3.1.2	Mit dem Workload-Monitor arbeiten	130
3.1.3	Technische Einstellungen zum Workload-Monitor	134
3.2	Workload-Analyse	135
3.2.1	Ablauf eines Transaktionsschrittes	135
3.2.2	Interpretation der Antwortzeiten	138
3.2.3	Aktivität, Durchsatz und Last	141
3.3	Workload-Analyse durchführen	142
3.3.1	Allgemeines Performanceproblem analysieren	143
3.3.2	Spezielles Performanceproblem analysieren	150
3.4	Anwendungsmonitor	154
3.4.1	Benutzerprofil	154
3.4.2	Last pro SAP-Modul	155
3.4.3	SAP-Puffer	156
3.5	Komponentenübergreifende Workload-Analyse	157
3.6	Zusammenfassung	162

4 Identifizierung von Performance-problemen in ABAP- und Java-Programmen 165

4.1	Einzelstatistik	165
4.2	Zentrale Einzelstatistik (Funktionaler Trace)	169
4.2.1	Mit der zentralen Einzelstatistik arbeiten	169
4.3	Performance-Trace	171
4.3.1	Performance-Trace erstellen	172
4.3.2	SQL-Trace auswerten	173
4.3.3	Weitere Funktionen im SQL-Trace	178
4.3.4	RFC-Trace auswerten	180
4.3.5	Enqueue-Trace auswerten	181
4.4	Performanceanalyse mit dem ABAP-Trace (Laufzeitanalyse)	182
4.4.1	ABAP-Trace erstellen	182
4.4.2	ABAP-Trace auswerten	183
4.4.3	Varianten verwenden	184
4.4.4	Laufzeitanalyse für BSP-Anwendungen aktivieren	186
4.4.5	Ausblick: Einzeltransaktionsanalyse	187
4.4.6	Mit der Einzeltransaktionsanalyse arbeiten	188
4.5	Analyse des Speicherverbrauchs mit dem ABAP-Debugger	189
4.5.1	Häufig beobachtete Performanceprobleme	191
4.6	Perfomancestatistiken für J2EE-Anwendungen (JARM)	193
4.6.1	Mit JARM-Statistiken im Enterprise Portal arbeiten	194
4.6.2	JARM-Trace (Single Activity Trace)	197
4.7	Zusammenfassung	198

5 Lastverteilung 201

5.1	SAP-Services	201
5.1.1	Message-, Enqueue- und ATP-Service verteilen	205
5.1.2	Dialog-, Hintergrund- und Spool-Workprozesse verteilen	205
5.1.3	Benutzer und Workprozesse auf CPU-Ressourcen verteilen	206
5.2	Dynamische Benutzerverteilung (Logon-Gruppen) und Betriebsarten	209
5.2.1	Dynamische Benutzerverteilung konfigurieren	209
5.2.2	Betriebsarten planen	212
5.3	Verbuchung	213
5.3.1	Verbuchungsaufträge überwachen	214
5.3.2	Verbuchungsworkprozesse verteilen	215
5.3.3	Verbuchungsarten wählen	216
5.4	Hardware sizing	220
5.4.1	Initiales Sizing planen	221
5.4.2	SAP Standard Application Benchmarks	225

5.4.3	Hardwarekapazitätsplanung bei Laststeigerung, Releasewechsel oder Migration (Re-Sizing)	227
5.4.4	Neue Herausforderungen durch das Internet	229
5.5	Systemlandschaft planen	230
5.5.1	Verteilung von SAP-Instanzen	231
5.5.2	Hardwarekonsolidierung	232
5.5.3	Mehrere SAP-Systeme in einer Datenbank	234
5.6	Zusammenfassung	234

6 Schnittstellen 237

6.1	RFC-Grundlagen	237
6.1.1	Begriffe	237
6.1.2	Ablauf eines RFCs	240
6.2	Schnittstellen zu externen Systemen	242
6.2.1	RFC-Verbindungen konfigurieren und testen	243
6.2.2	Eingehende und ausgehende Last überwachen	246
6.2.3	Parallelisierung von Prozessen mit asynchronen RFCs konfigurieren	251
6.2.4	Datenübertragung mit transaktionalen RFCs überwachen	252
6.3	Zusammenfassung	253

7 SAP GUI und Internetanbindung 255

7.1	SAP GUI 4.6	255
7.1.1	Interaktionsmodell und Performancemessung	256
7.1.2	Performance der GUI-Kommunikation analysieren und optimieren	258
7.2	SAP Web Application Server (SAP Web AS)	264
7.3	SAP Internet Transaction Server (SAP ITS)	265
7.3.1	ITS-Grundlagen	265
7.3.2	Web-Anbindung planen	269
7.3.3	ITS konfigurieren	270
7.3.4	ITS-Administrationswerkzeuge	273
7.3.5	Engpassanalyse für den ITS durchführen	275
7.4	Business Server Pages (BSP) und Internet Communication Manager (ICM)	276
7.4.1	Grundlagen von SAP Business Server Pages	276
7.4.2	ICM konfigurieren und überwachen	280
7.4.3	Performanceanalyse von BSPs durchführen	282
7.5	SAP J2EE Engine	285
7.5.1	Grundlagen der SAP J2EE Engine	286
7.5.2	Konfigurations- und Überwachungswerkzeuge	290
7.6	End-to-End-Performanceüberwachung	293
7.6.1	Laufzeitanalyse im Webbrowser durchführen	293

7.6.2	Web-Anwendungen mit dem CCMS kontinuierlich überwachen	296
7.6.3	Werkzeuge von Drittanbietern	297
7.7	Zusammenfassung	297

8 Speicherkonfiguration 301

8.1	Grundlagen der Speicherkonfiguration	301
8.1.1	Begriffsklärungen	301
8.1.2	SAP-Roll Memory, SAP-Extended Memory, SAP-Heap Memory	303
8.1.3	SAP-EG Memory und SAP-Paging Memory	312
8.2	SAP-Speicherbereiche konfigurieren und überwachen	314
8.2.1	Auslagerungsspeicher überwachen	316
8.2.2	Beschränkungen durch den Adressraum (32- und 64-Bit-Architektur)	319
8.2.3	SAP-Speicherbereiche konfigurieren und überwachen	324
8.2.4	Hilfe zur Fehlerbehebung	328
8.3	Zusammenfassung	334

9 SAP-Tabellenpufferung 337

9.1	Vorbemerkung zu den Kapiteln 9, 10 und 11	338
9.2	Grundlagen der Tabellenpufferung	339
9.2.1	Pufferungsarten	339
9.2.2	Pufferzugriffe	342
9.2.3	Puffersynchronisation	344
9.2.4	Pufferung einschalten	346
9.2.5	Welche Tabellen sollen gepuffert werden?	348
9.3	SAP-Tabellenpufferung überwachen	350
9.3.1	Tabellenzugriffsstatistik	351
9.3.2	Analyse der gepufferten Tabellen	354
9.3.3	Analyse der bisher nicht gepufferten Tabellen	360
9.3.4	Detaillierte Tabellenanalyse	362
9.3.5	Monitor zur Puffersynchronisation (DDLOG-Einträge)	364
9.3.6	Shared SQL Area und SQL-Trace	366
9.4	Zusammenfassung	367

10 Sperren 371

10.1	Sperrenkonzepte von Datenbanksystem und SAP-System	371
10.1.1	Datenbanksperren	372
10.1.2	SAP-Enqueues	373

10.2	Überwachung von Sperren	375
10.2.1	Datenbanksperren	375
10.2.2	SAP-Enqueues	378
10.3	Nummernkreispufferung	380
10.3.1	Grundlagen	380
10.3.2	Nummernkreispufferung einschalten	384
10.3.3	Nummernkreispufferung überwachen	385
10.4	ATP-Server	386
10.4.1	Grundlagen	387
10.4.2	ATP-Server konfigurieren	389
10.4.3	ATP-Server überwachen	392
10.5	Zusammenfassung	392

11 Optimierung von SQL-Anweisungen 395

11.1	Teure SQL-Anweisungen identifizieren und analysieren	396
11.1.1	Identifizierung und Voranalyse	396
11.1.2	Detaillierte Analyse	398
11.2	SQL-Anweisungen durch Sekundärindizes optimieren	402
11.2.1	Ein wenig Theorie	402
11.2.2	Indizes und Tabellenstatistiken administrieren	411
11.2.3	Regeln zum Anlegen oder Ändern von Sekundärindizes	416
11.3	SQL-Anweisungen im ABAP-Programm optimieren	423
11.3.1	Regeln für die performante SQL-Programmierung	423
11.3.2	Beispiel für die Optimierung eines ABAP-Programms	428
11.3.3	Vorbelegung von Feldern in Reporttransaktionen	437
11.4	Zusammenfassung und weitere Hinweise	443

A Flussdiagramme und Checklisten zur Performanceanalyse 449

A.1	Flussdiagramme	449
A.2	Checklisten	455

B Datenbankmonitore 465

B.1	Datenbankprozessmonitor	465
B.2	Statistik über SQL-Anweisungen (Shared SQL Area)	467
B.3	Festplattenzugriffe	470
B.4	Datenbanksperren (Exclusive Lockwaits)	471
B.5	Datenbankpuffer	472
B.6	Ausführungspläne	483

C	Performancerelevante Konfigurationsparameter	497
C.1	Puffer	498
C.2	Speicherverwaltung	500
C.3	Weitere Parameter	502
C.4	Internet Communication Manager	505
C.5	Internet Transaction Server	507
D	Wichtige Transaktionscodes	509
E	Fragen und Antworten	511
F	Glossar	523
G	Informationsquellen	533
H	Hinweise im SAP Service Marketplace	537
	Index	541

Geleitwort zur Buchreihe

Den Betrieb der SAP-Software-Lösungen in Ihrem Unternehmen mit Erfolg und minimalen Kosten zu ermöglichen, hat bei der SAP AG oberste Priorität. Diese »Lowest Cost of Ownership« erreichen Sie sowohl durch eine effiziente und schnelle Implementierung als auch durch einen optimierten und gesicherten produktiven Betrieb. SAP Active Global Support steht Ihnen hierbei mit der neuen Strategie SAP Solution Management tatkräftig und konsequent zur Seite. Über den gesamten Lebenszyklus einer Lösung bietet SAP dem Kunden die erforderlichen Services, erstklassigen Support, die geeignete Infrastruktur und das relevante Know-how. Gestützt wird die neue Strategie durch drei mächtige Support-Programme: *Safeguarding*, das man vereinfacht als Risikomanagement umschreiben könnte, *Solution Management Optimization*, das der Optimierung der IT-Lösung des Kunden dient, und *Empowering*, das einen gezielten, effektiven Wissenstransfer von SAP hin zum Kunden ermöglicht.

Wissensvermittlung steht auch im Mittelpunkt des vorliegenden Buches aus der Reihe *SAP Technical Support Guides*. Diese Reihe bietet Ihnen einen detaillierten Überblick über technische Aspekte und Konzepte für das Management von SAP-Software-Lösungen. Die Themen der Bücher reichen vom technischen Einführungsprojekt bis hin zum Betrieb eines Software-Systems und des jeweiligen Datenbanksystems.

Ob Sie in das SAP-Systemmanagement gerade einsteigen oder sich weiter qualifizieren möchten, in beiden Fällen profitieren Sie von den praxisnahen Erfahrungen und Informationen aus erster Hand in diesen Büchern. Daneben möchte Ihnen die SAP mit dieser Reihe helfen, sich eigenständig auf eine Prüfung zum »Zertifizierten Technischen Berater« vorzubereiten. Beachten Sie jedoch: Diese Bücher können und wollen eigene Erfahrungen mit den verschiedenen SAP-Lösungen nicht ersetzen! Vielmehr geben Ihnen die Autoren Empfehlungen für die tägliche Arbeit mit der Software.

Die Innovation der SAP-Lösungen bringt immer wieder neue Herausforderungen und Lösungen auch für das Systemmanagement mit sich. Damit steigen auch die Anforderungen an kundeneigene oder externe Support-Organisationen. Die Fähigkeiten und Kenntnisse dieser Organisationen können wesentlich dazu beitragen, Schwierigkeiten beim Einsatz der Software zu vermeiden. Deswegen ist die Vermittlung von Fähigkeiten zur Problemlösung eine der Kernaufgaben dieser Buchreihe.

Bücher erweisen sich in ihrer Form auch in der Zeit des Internets als ideales Medium, um Wissen in kompakter Form zu vermitteln. Gleichzeitig ergänzen sie mit ihren Inhalten die neue Service- und Support-Plattform, den SAP Solution Manager, und die neuen Services, die SAP anbietet. Die Buchreihe liefert Hintergründe zur Handhabung und Funktionsweise der neuen Lösungen der SAP und trägt somit zur Kundenzufriedenheit bei.



Gerhard Oswald

Mitglied des Vorstandes der SAP AG



Uwe Hommel

Senior Vice President der SAP AG

SAP Active Global Support

Rot, im August 2005

Danksagung

Der große Erfolg der ersten Auflagen dieses Buches – man kann sicher sagen, dass es nicht nur bei den Kunden, sondern auch bei vielen Mitarbeitern der SAP ein Eckpfeiler in der Performanceausbildung geworden ist – bringt die Verpflichtung mit sich, eine aktualisierte und erweiterte Fassung vorzulegen. Dieser Verpflichtung komme ich gerne nach. Die Auswahl und die Darstellung der Themen sind im Wesentlichen geprägt von den Erfahrungen, die meine Kolleginnen, Kollegen und ich im konkreten Umgang mit vielen SAP-Systemen im produktiven Einsatz gemacht haben – sei es in den SAP-Services EarlyWatch und GoingLive Check, in Trainings zur Performanceanalyse oder nicht zuletzt bei der Vor-Ort-Analyse von Systemen in Phasen mit kritischen Performanceproblemen. Aufgrund dieser Erfahrungen sind wir zuversichtlich, mit diesem Buch eine breite Palette wichtiger performancerelevanter Themen abzudecken.

Dieses Buch konnte nur aufgrund der Mitwirkung vieler kompetenter Diskussionspartner entstehen. Es ist mir wichtig, zunächst unseren Kollegen und Mentor Augustinus Wohlfart hervorzuheben, dessen plötzlicher und unerwarteter Tod uns tief erschüttert hat. Er hatte als einer der Initiatoren dieser Buchreihe maßgeblichen Anteil daran, dass diese Veröffentlichung entstehen konnte. Ihm möchte ich dieses Buch widmen.

Weiterhin möchte ich mich namentlich bei folgenden Kolleginnen und Kollegen bedanken: Hartwig Brand, Bernhard Braun, Matthias Blümmler, John Landis, Vivian Luechau-de la Roche, Ulrich Marquard, Dirk Müller, Jens Otto, Marc Thier, Fabian Tröndle, Gerold Völker und Liane Will.

Tomas Wehren und Florian Zimniak von Galileo Press möchte ich für die gute Zusammenarbeit danken.

Für die Hilfe bei den in der dritten Auflage hinzugekommenen Teile über Business Server Pages und Java-Anwendungen danke ich ganz herzlich Jens Claussen, Guido Derwand und Mandy Krimmel. Bei der Aktualisierung der Abschnitte zu den unterschiedlichen Datenbanken für die vorliegende vierte Auflage danke ich Brigitte Huy, Anja Kerber, Christian Knapcke, Claudia Langner, Wladimir Maljutin, Isolde Savelsberg-Walter und Gienek Wajda.

Dr. Thomas Schneider

SAP Strategic Research & Development

7 SAP GUI und Internetanbindung

Mit der EnjoySAP-Initiative (SAP R/3 4.6) liefert die SAP eine Überarbeitung ihrer Software aus – mit einer Oberfläche, die für den Benutzer intuitiver und leichter zu lernen und anzuwenden ist als die Vorgängerversionen. Die Grundlage der neuen Oberfläche ist eine neuartige Interaktion zwischen Präsentations- und Applikationsebene, die als *Controls-Technik* bezeichnet wird. Der erste Abschnitt dieses Kapitels ist den Performanceaspekten von Controls gewidmet.

Zunehmend mehr Benutzer melden sich nicht über den klassischen SAP GUI für Windows oder Java Environment an SAP-Systeme an, sondern über einen Webbrowser, was den Vorteil hat, dass auf den Desktop-Computern keine speziellen GUI-Programme installiert werden müssen. Die Kommunikation zwischen dem Webbrowser und der SAP-Applikationsebene wird über eine der folgenden Alternativen vermittelt:

- ▶ Web-Anwendungen auf Basis des SAP Internet Transaction Servers (SAP ITS)
- ▶ Business Server Pages
- ▶ Java Server Pages und Java Servlets auf Basis der J2EE-Laufzeitumgebung des SAP Web Application Servers (SAP J2EE Engine)

Der zweite bis sechste Abschnitt führt in die Konfiguration und Performanceüberwachung solcher Web-basierter Benutzeroberflächen (Web-UIs) ein.

Wann sollten Sie dieses Kapitel lesen?

Dieses Kapitel führt Sie als Administrator in Konfiguration und Überwachung der SAP GUI-Schnittstellen (ab SAP-Basis 4.6) und des SAP Internet Transaction Servers ein. Vor diesem Kapitel sollten Sie den Abschnitt 6.1, »RFC-Grundlagen«, gelesen haben.

7.1 SAP GUI 4.6

Mit der EnjoySAP-Initiative (SAP R/3 4.6) liefert die SAP eine Überarbeitung ihrer Software aus. Konkret ist SAP R/3 4.6 durch drei Neuerungen ausgezeichnet:

- ▶ ein neues visuelles Design
- ▶ ein neues Interaktionsdesign
- ▶ die stärkere Personalisierung der Funktionalität

Technisch basiert das neue Design auf einem komplett neuen Interaktionsmodell für die Kommunikation zwischen Präsentations- und Applikationsebene, der Control-Technik.

Alle drei GUI-Varianten, die für den Zugriff auf das SAP-System zur Verfügung stehen, der SAP GUI for HTML, der SAP GUI for Windows oder der SAP GUI for Java, unterstützen die Verwendung von Controls, sind also »Enjoy-fähig«.

7.1.1 Interaktionsmodell und Performancemessung

EnjoySAP und Controls

Controls sind Oberflächenelemente, die es dem Anwendungsentwickler erlauben, seine Benutzeroberflächen den Bedürfnissen der Benutzer stärker als bisher anzupassen und mehr unterschiedliche Elemente in einen Bildschirm zu integrieren. Typische Controls sind:

- ▶ ABAP List Viewer Control (ALV Control)
- ▶ Tree Control
- ▶ Textedit Control
- ▶ HTML Control



Beispiele für Controls finden Sie in Ihrem SAP-System unter **Werkzeuge · ABAP Workbench · Entwicklung · ABAP Editor · Umfeld · Control-Beispiele** (Transaktionscode DWDM).

Controls sind keine Bildelemente im klassischen Sinne, sondern vielmehr Softwarekomponenten, die im SAP GUI-Programm eigenständig laufen und auch über eigene Funktionalität verfügen, die am GUI abläuft und keine Kommunikation mit der Applikationsebene erfordert.

In einer herkömmlichen Liste oder einer herkömmlichen Baumstruktur (vor SAP-Basis 4.6) bedeutet zum Beispiel jedes Blättern bzw. jedes Auf- und Zuklappen eines Astes im Baum einen Kommunikationsschritt zwischen GUI und Applikationsebene. Verwendet man den neuen List Viewer Control bzw. den Tree Control, so wird bereits beim ersten Aufbau der Liste bzw. des Baumes eine größere Datenmenge übertragen. Somit kann das Navigieren in der Liste bzw. im Baum selbstständig im GUI und ohne Rückgriff auf die Applikationsebene erfolgen. Weitere Aktionen, die bei manchen Controls direkt im GUI abgewickelt werden können, sind Funktionen wie »Suchen« und »Ersetzen«.

Das neue Interaktionsmodell führt also im Mittel zu weniger Kommunikationsschritten zwischen Präsentations- und Applikationsebene. Dies ist ein klarer Vorteil für einen Benutzer, der einen Bildschirm einmal aufruft

und dann häufig darin navigiert. Auf der anderen Seite steigt dafür die Netzwerklast für das initiale Aufbau der Bildschirmseite. Wenn Sie daher einen komplexen Bildschirm nur einmal aufrufen und ihn danach sofort wieder verlassen, ist das neue Interaktionsmodell ein Nachteil. Im Mittel zeigen Messungen, dass sich die mittlere Netzwerklast im Release 4.6 im Vergleich zu den Vorgängerversionen erhöht (SAP-Hinweis 164102).

Welche Aktionen einen Kommunikationsschritt mit der Applikationsebene erfordern, lässt sich ohne Kenntnis der konkreten Programmierung nicht sagen. Anwendungsentwickler sind gehalten, Daten in sinnvollen Portionen an den GUI zu übertragen. Listen oder Bäume, die wenige Seiten lang sind, werden komplett übertragen, sehr lange Listen oder Bäume werden dagegen in Portionen übertragen (z. B. bei Bäumen eine feste Anzahl von Knoten im Voraus). Das neue Interaktionsmodell bedeutet also deutlich mehr Verantwortung für den Anwendungsentwickler, der in seinem Programm entscheiden kann, wie viele Daten in wie vielen Portionen an den GUI übertragen werden sollen.

Um einen Bildschirm aufzubauen, sind in einem Transaktionsschritt eventuell mehrere Interaktionen zwischen der Applikationsebene und dem GUI notwendig. Diese Interaktionen bezeichnen wir als *Roundtrip*. Bei einem Roundtrip werden Daten mit einem synchronen RFC von der Applikationsebene an den GUI übertragen. Die gesamte Zeit der Kommunikation mit dem GUI innerhalb eines Transaktionsschritts wird im Statistiksatz als *GUI-Zeit* verzeichnet. Während das Programm in der Applikationsebene auf die Antwort des GUI wartet, wird es aus dem Workprozess herausgerollt. Im Statistiksatz wird diese Wartezeit als *Roll-Wartezeit* verzeichnet. In einem Statistiksatz zu einem Transaktionsschritt, bei dem Controls aufgebaut werden und kein RFC zu einem externen System erfolgt, wird man in der Regel beobachten, dass Roll-Wartezeit und GUI-Zeit etwa gleich groß sind. Werden in einem Transaktionsschritt sowohl Controls aufgebaut als auch externe RFCs gestartet, so wird die Roll-Wartezeit größer als die GUI-Zeit sein, da die Roll-Wartezeit die Zeit für die RFCs zum GUI und die Zeit für die externen RFCs umfasst.

Roundtrip, Roll-Wartezeit und GUI-Zeit

Die zwischen Applikationsebene und GUI übertragene Datenmenge können Sie im Statistiksatz den Feldern **Terminal out-message** (Applikationsebene an GUI) bzw. **Terminal in-message** entnehmen.

Übertragene Datenmenge

Abbildung 7.1 zeigt schematisch, welche Zeiten bei einem Transaktionsschritt mit Control-Technik gemessen werden.

7.1.2 Performance der GUI-Kommunikation analysieren und optimieren

Einzelstatistik

Wenn Sie den Verdacht haben, dass in Ihrem System Probleme mit der GUI-Kommunikation beim Aufbau von Controls bestehen, dann rufen Sie zunächst die *Einzelstatistik* auf (Transaktionscode STAD, siehe Abschnitt 4.1, »Einzelstatistik«).

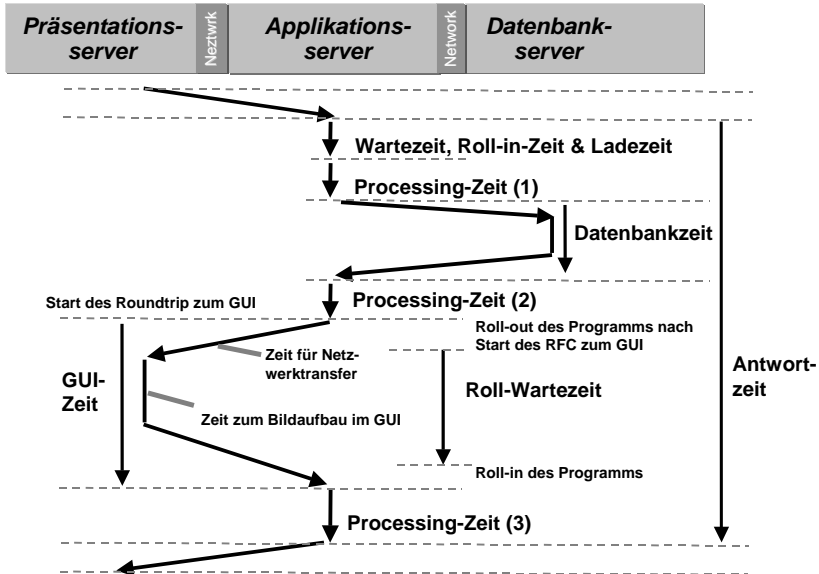


Abbildung 7.1 Ablauf eines Transaktionsschritts mit Control-Technik und die dabei gemessenen Zeiten

1. Schränken Sie im Selektionsbildschirm die Suche ein, z. B. auf den Benutzer, der das Problem meldet, oder auf einen Transaktions- oder Programmnamen sowie einen Zeitbereich.
2. Sie gelangen auf den Übersichtsbildschirm. Mit der Funktion **Select Fields** wählen Sie nun die Felder aus, die schwerpunktmäßig Probleme in der GUI-Kommunikation indizieren: **Roll wait time**, **No. of roundtrips**, **GUI time**, **Terminal in-message** und **Terminal out-message**.
3. Blättern Sie nun in der Liste und suchen Sie nach Transaktionsschritten mit hoher GUI-Zeit oder hoher Datenübertragungsmenge (**Terminal out-message**). Dabei können Sie sich an folgenden Richtlinien orientieren:
 - Im Mittel sollten Sie bei der Datenübertragung an den GUI eine Rate von 1 kByte pro 100 ms erreichen. Im langsamsten Fall sollte eine Sekunde für 1 kByte nicht überschritten werden. Werden diese

Richtwerte häufig überschritten, so steht zu vermuten, dass entweder ein Netzwerkproblem oder ein Hardwareengpass auf dem Präsentationsserver vorliegt.

- ▶ Weiterhin ist es möglich, dass die Übertragungsgeschwindigkeit gut und die GUI-Zeit dennoch hoch ist, weil die übertragene Datenmenge zu hoch ist. Als Richtwerte sollte ein Transaktionsschritt im Mittel nicht mehr als 5 bis 8 kByte übertragen. Auch ein komplexer Bildschirmaufbau sollte nicht mehr als 50 kByte Daten benötigen. Werden diese Richtwerte häufig überschritten, so liegt das Problem im Programm bzw. in seiner Verwendung begründet.

Ein weiteres wichtiges Werkzeug bei der Analyse von Performanceproblemen mit dem GUI ist der *Performance-Trace* (Transaktionscode ST05).

Performance-Trace

1. Aktivieren Sie den Performance-Trace (SQL-Trace, Enqueue-Trace und RFC-Trace) für die zu untersuchende Benutzeraktivität und listen Sie anschließend das Ergebnis des Traces.
2. Suchen Sie im Trace nach RFC-Bausteinen, die als Ziel Ihren Präsentationsserver haben. Suchen Sie dazu nach dem Namen Ihres Präsentationsservers in der Spalte **Object**. Typische Funktionsbausteine für die Übertragung an den GUI sind SAPGUI_PROGRESS_INDICATOR und OLE_FLUSH_CALL (siehe auch Abbildung 4.3 in Kapitel 4).
3. Summieren Sie die Antwortzeit für diese Funktionsbausteine und prüfen Sie, ob diese einen erheblichen Anteil an der gesamten Antwortzeit ausmacht.
4. Vergleichen Sie für jeden Funktionsbaustein Antwortzeit und übertragene Datenmenge. Die Antwortzeit in Mikrosekunden für einen RFC wird im Feld **Duration** in der Trace-Grundliste angezeigt. Um die übertragene Datenmenge zu ermitteln, führen Sie einen Doppelklick auf die entsprechende Zeile in der Trace-Liste aus und entnehmen den Wert dem Feld **Bytes sent**. Vergleichen Sie die Werte mit den oben angegebenen Richtwerten.

Ein Entwicklerwerkzeug zur Performanceanalyse von Controls ist der Automation-Queue-Trace, der sich über den SAP GUI aktivieren lässt. Informationen hierzu finden Sie im SAP-Hinweis 158985.

Stellen Sie fest, dass trotz relativ kleiner Datenmenge die GUI-Zeit hoch ist, so kann dies zwei Ursachen haben: einen Hardwareengpass auf dem Präsentationsserver oder einen Netzwerkengpass. Zur weiteren Analyse ist es oft am einfachsten zu versuchen, aus der Einzelsatzstatistik die Benutzer herauszufiltern, bei denen diese Probleme typischerweise auftreten.

**Hohe GUI-Zeit
trotz moderater
Datenmenge**

Die Felder **Terminal out-message** (in der Einzelsatzstatistik) und **Bytes sent** (im Performance-Trace) geben die unkomprimierte Datenmenge beim Bildschirmaufbau bzw. beim RFC an. Bevor diese Daten über das Netz gehen, werden sie komprimiert, so dass eine deutlich kleinere Datenmenge effektiv übertragen wird.



Abbildung 7.2 zeigt ein Beispiel für einen Statistiksatz mit einer extrem langsamen GUI-Kommunikation. Von den 27,5 Sekunden, die der Transaktionsschritt gedauert hat, entfallen 25,5 Sekunden auf die GUI-Zeit. Die übertragene Datenmenge (Feld **Terminal out-message**, im Bildschirmausschnitt nicht dargestellt), ist in diesem Fall 17 kByte. Die Übertragungsrates entspricht mit 1,5 Sekunden pro kByte nicht unseren Ansprüchen.

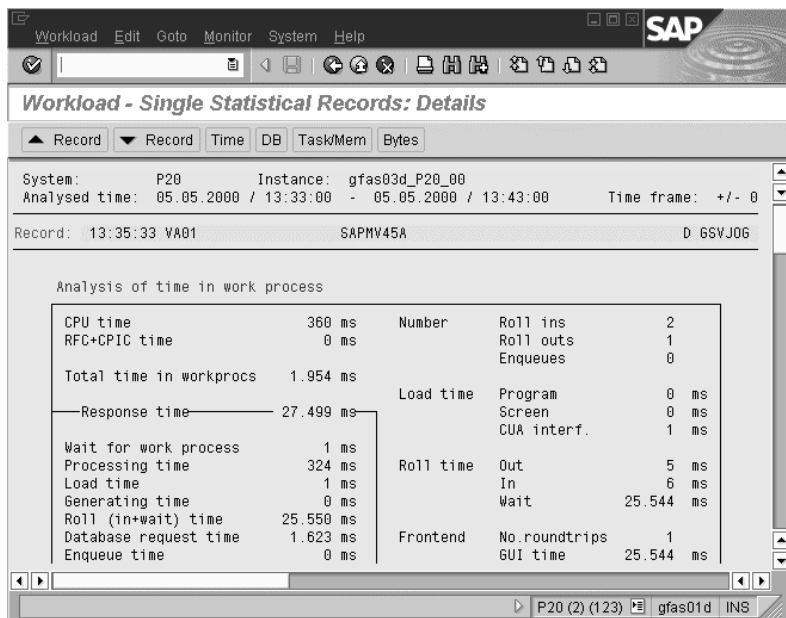


Abbildung 7.2 Beispiel für einen Statistiksatz (Transaktionscode STAD) mit einer extrem langsamen GUI-Kommunikation. Von den 27,5 Sekunden, die der Transaktionsschritt gedauert hat, entfallen 25,5 Sekunden auf die GUI-Zeit.

In einem nächsten Analyseschritt wird man das Netzwerk zwischen der Präsentations- und der Applikationsebene unter die Lupe nehmen. Dazu steht Ihnen im Betriebssystemmonitor die Funktion **Check LAN by Ping** zur Verfügung.

1. Wählen Sie im Betriebssystemmonitor (Transaktionscode ST06) die Funktion **Detailed Analysis Menu · Check LAN by Ping**. Wählen Sie anschließend die Funktion **Presentation Server**.

2. Sie gelangen auf eine Liste mit allen angemeldeten Präsentationsservern. Markieren Sie die Präsentationsserver, die Sie analysieren wollen (oder wählen Sie zehn Präsentationsserver willkürlich aus) und selektieren Sie diese mit der Funktion **Pick**. Starten Sie die Netzwerkanalyse über die Funktion **10 x Ping**.
3. Ein Beispiel für das Ergebnis der Analyse zeigt Ihnen Abbildung 7.3. In diesem Beispiel wurden zehn Präsentationsserver ausgewählt, deren Namen und IP-Adressen in den Spalten **Servername** und **Server-IP** angezeigt werden. Für jeden dieser Server zeigt der Bildschirm die minimale, mittlere und maximale Laufzeit für einen Ping-Befehl (Spalten **Min/ms**, **Avg/ms** und **Max/ms**) sowie die Anzahl der Pings, die nicht beantwortet wurden (in der Spalte **Loss**). Typische Antwortzeiten für einen Ping mit 4 kByte Paketgröße sind:
 - ▶ In einem Local Area Network (LAN): < 20 Millisekunden
 - ▶ In einem Wide Area Network (WAN): < 50 Millisekunden
 - ▶ Bei Modem-Verbindungen (z. B. 56-kByte-Modem): < 250 Millisekunden
 - ▶ Verluste von Datenpaketen (»Losses«) sollten gar nicht auftreten.

The screenshot shows the SAP 'LAN Check by PING (Presentation Server)' window. The title bar includes 'Server System Help' and the SAP logo. Below the title bar is a menu bar with 'Server System Help' and a toolbar with various icons. The main window title is 'LAN Check by PING (Presentation Server)'. Below this is a 'Details' button. The main content area displays the text '05.05.2000 13:40 10 Pings to Presentationsservers from gfas01d'. Below this is a table with the following data:

Servername	Server-IP	Min ms	Avg ms	Max ms	Loss %
dk1016	193.3.14.251	---	---	---	100
dk1022	193.3.14.251	---	---	---	100
dk1024	193.3.14.251	---	---	---	100
dk1025	193.3.14.251	---	---	---	100
sa1amanna	194.194.192.226	476	511	549	0
noretta	194.194.192.25	175	534	1260	0
6FDHTE	194.253.242.65	168	619	989	---
6FDJMB	194.253.242.67	192	518	937	10
gb0039	195.118.134.72	281	288	331	0
gbabi	195.118.134.79	279	296	370	0
Impossible links: 4		168	461	1.260	---

At the bottom of the window, there is a status bar showing 'P20 (1) (123) gfas01d INS'.

Abbildung 7.3 Beispiel für eine Analyse eines Netzwerkes zwischen Applikations- und Präsentationsebene (Transaktionscode ST06). Die Time-outs und langsamen Antwortzeiten zeigen ein massives Kommunikationsproblem auf.

Dieser Bildschirmabgriff stammt von demselben SAP-System wie Abbildung 7.2. Die Analyse zeigt ein völlig korruptes Netzwerk mit massiven Kommunikationsproblemen: Zu einigen Präsentationsservern kann gar keine Verbindung aufgebaut werden, andere antworten mit einer Laufzeit von 300–500 Millisekunden. Damit erklären sich die schlechten GUI-Zeiten, die wir in dem statistischen Satz in Abbildung 7.2 sehen. Aus Sicht des SAP-Systems kann hier zunächst nichts mehr unternommen werden. Der nächste Schritt muss eine Analyse mit den Werkzeugen des Netzwerkherstellers und die »Reparatur« des Netzwerks sein.

Große Datenmenge

Der zweite oben beschriebene Fall liegt vor, wenn die Übertragungsgeschwindigkeit (im Mittel 100 ms pro kByte oder besser) gut ist, es aber zum Problem kommt, weil pro Transaktionsschritt zu viele Daten übertragen werden. Dieses Problem kann nur gelöst werden, wenn in einem Schritt weniger Daten übertragen werden. Dies kann über eine Vereinfachung des Bildschirms geschehen. In vielen Fällen enthalten die Bildschirme, die die SAP als Standard ausliefert, deutlich mehr Informationen, als einzelne Benutzer für ihre Arbeit benötigen. Über Transaktionsvarianten lassen sich die Bildschirme an die konkreten Bedürfnisse der Benutzer anpassen und gleichzeitig lässt sich die Performance verbessern. Wie Sie Transaktionsvarianten zur Performanceoptimierung verwenden, beschreibt SAP-Hinweis 332210 konkret am Beispiel der Transaktion »Kundenauftrag anlegen« (VA01). Führt dies nicht zu einer Lösung des Problems, muss der Entwickler der betroffenen Transaktion diese im Detail analysieren.

Sollte im Performance-Trace die Antwortzeit des Bausteins SAP_GUI_PROGRESS_INDICATOR einen erheblichen Anteil an der gesamt Antwortzeit ausmachen, so können Sie dieses Problem leicht lösen. Der Baustein aktualisiert lediglich die Statusmeldung in der Fußzeile Ihres GUIs während eines Dialogschrittes. Er überträgt z. B. Meldungen wie »Daten werden geladen« und aktualisiert die kleine Uhr in der linken unteren Ecke des Bildschirmfensters, die den Fortschritt einer Operation anzeigt. Sie können diese Statusanzeige einfach auf Benutzerebene deaktivieren, indem Sie den Benutzerparameter SIN auf den Wert »0« setzen.

Weitere Optimierungsmöglichkeiten

Low Speed Connection

In WAN-(Wide Area Network-)Umgebungen empfehlen wir, die Netzwerkkommunikation zwischen GUI und Applikationsebene auf *Low Speed Connection* umzustellen. Dadurch wird die Menge der übertragenen Daten pro Dialogschritt verringert (siehe SAP-Hinweis 164102). Sie akti-

vieren die Low Speed Connection im SAP Logon-Fenster, indem Sie den Eintrag für ein SAP-System markieren und über den Menüpunkt **Properties Advanced** die Option **Low Speed Connection** auswählen.

Mit SAP GUI 4.6 hat die SAP ihrer Benutzeroberfläche ein völlig neues visuelles Design gegeben (*New Visual Design*), das höhere Anforderungen an die Hardware der Präsentationsserver stellt. Entsprechen Ihre Präsentationsserver nicht den in SAP-Hinweis 26 417 beschriebenen Hardwareanforderungen, so kann es sein, dass hohe GUI-Zeiten auftreten, weil die CPU des Präsentationsservers zu langsam ist oder der Präsentationsserver wegen Hauptspeichermangels stark auslagert. In diesem Fall lässt sich Performance gewinnen, indem Sie lokal von dem neuen GUI-Design, mit dem der SAP GUI 4.6 ausgeliefert wird, auf das alte SAP GUI-Design umschalten. Dazu finden Sie lokal auf Ihrem Desktop das Icon **SAP Visual Settings**, mit dem Sie zwischen dem neuen und alten Design hin- und herschalten können. Die Einstellung wird für den nächsten Modus, den Sie öffnen, aktiv.

New Visual Design

Das Aktivieren der Low Speed Connection und das Deaktivieren des New Visual Designs bringt keine funktionalen Einschränkungen mit sich. Controls werden weiterhin ganz normal angezeigt. Das Einschalten der Low Speed Connection ist unabhängig von dem Deaktivieren des New Visual Designs. Ersteres hilft bei einer langsamen Netzwerkverbindung, Letzteres bei einem Hardwareengpass auf dem Präsentationsserver. Das Deaktivieren des New Visual Designs bringt nur dann einen Performancegewinn, wenn alte PCs eingesetzt werden, die nicht den Anforderungen des SAP GUI 4.6 entsprechen. Bei ausreichender PC-Hardware bringt der Umstieg auf das alte Design keinen Vorteil.

In SAP-Basis 4.6 erfolgt der Einstieg für den Benutzer nicht mehr über das bekannte SAP-Standardmenü, sondern über das *SAP Easy Access Menu*. Dort sieht der Benutzer ein personalisiertes Menü, das ihm über seine Rolle zugewiesen wird. Dieses enthält nur noch die Transaktionen, die er für seine Arbeit wirklich benötigt. Alternativ können Sie natürlich auch das globale SAP-Menü verwenden. Beim Einrichten des SAP Easy Access Menu sollten Sie auf eine performante Realisierung achten, denn nichts ist frustrierender für einen Benutzer, als wenn er gleich beim Anmelden auf seinen ersten Bildschirm warten muss. Die wichtigen Empfehlungen sind, auf aufwändige Hintergrundbilder im SAP Easy Access Menu zu verzichten (sie sollten möglichst nicht mehr als 20 kByte groß sein) und die Anzahl der Transaktionen in einer Rolle zu begrenzen (möglichst auf 1 000 oder weniger). SAP-Hinweis 203924 gibt Ihnen die notwendigen Informationen zum effizienten Customizing.

SAP Easy Access Menu

7.2 SAP Web Application Server (SAP Web AS)

Der SAP Web Application Server (technisch SAP-Basis 6.10 und höher) liegt u. a. SAP R/3 Enterprise, SAP BW 3.x oder SAP CRM 3.x zugrunde. Mit dem SAP Web Application Server stehen Ihnen drei Techniken zur Verwirklichung von Web-Anwendungen zur Verfügung (als weitere Erläuterung zu den folgenden Angaben dient auch Abbildung 7.4):

► Web-Anwendungen auf Basis des SAP Internet Transaction Servers (ITS)

Diese Technik steht Ihnen seit SAP R/3 3.1 zur Verfügung. Anwendungen auf Basis des ITS werden von der SAP seit SAP R/3 4.6C nicht mehr realisiert. Um die Investitionen in ITS-Anwendungen auf Seiten der SAP und ihrer Kunden zu schützen, wird der ITS von der SAP weiter gewartet und in den SAP Web Application Server integriert.

► Web-Anwendungen auf der Basis von Business Server Pages

Seit dem SAP Web Application Server 6.10 können Web-Anwendungen direkt in der ABAP-Entwicklungsumgebung entwickelt und in der ABAP-Laufzeitumgebung ausgeführt werden. Diese Anwendungen werden als Business Server Pages (BSPs) bezeichnet. Um diese Web-Anwendungen auszuführen, bedarf es keiner Installation zusätzlicher Softwarekomponenten (auch nicht eines separaten Webserver).

► Web-Anwendungen auf der Basis von Java Server Pages und Java Servlets

Mit der SAP J2EE Engine enthält der SAP Web Application Server eine komplette Entwicklungs- und Laufzeitumgebung für Java-Web-Anwendungen (Java Server Pages und Java Servlets) sowie für Java-»Backend«-Anwendungen (Enterprise Java Beans).

Als Manager für an den SAP Web Application Server gerichtete Web-Requests dient der Internet Communication Manager (ICM) sowie ggf. der SAP Web Dispatcher. Diese nehmen alle Anfragen von Web-Clients entgegen und verteilt diese an den ITS, an den SAP-Dispatcher bzw. an den J2EE-Server.

In den drei folgenden Abschnitten stellen wir die grundlegende Architektur und die Methoden der Performanceüberwachung für ITS-Anwendungen, für BSP-Anwendungen sowie für Java-Anwendungen vor.

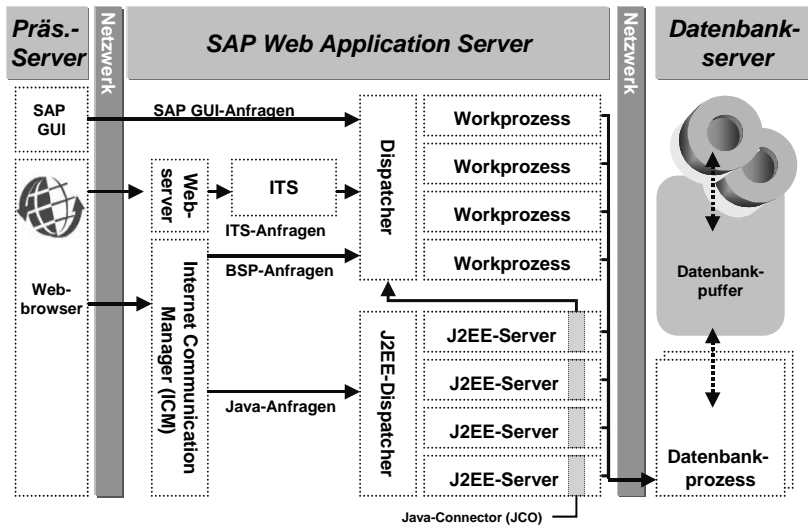


Abbildung 7.4 Struktur des SAP Web Application Servers. Der Internet Communication Server (ICM) ist seit SAP Web AS 6.10 Bestandteil der SAP-Basis, die Integration von »klassischer« SAP-Basis und J2EE Engine erfolgte mit SAP Web AS 6.20, seit SAP Web AS 6.30 ist der Internet Transaction Server (ITS) integriert.

7.3 SAP Internet Transaction Server (SAP ITS)

Seit SAP R/3 3.1 kann über den SAP Internet Transaction Server (SAP ITS) und einen Webserver von einem Webbrowser direkt auf die SAP-Applikationsebene zugegriffen werden. Als Webserver kommen dabei z. B. der Microsoft Internet Information Server oder der Netscape Enterprise Server in Betracht.

7.3.1 ITS-Grundlagen

Der ITS besteht aus zwei Komponenten, dem *WGate* und dem *AGate*. Das *WGate* stellt die Verbindung zum Webserver her. Seine Aufgabe ist es, Anfragen an den Webserver, die an den ITS gerichtet sind, zu erkennen und weiterzuleiten. Auf der NT-Plattform ist das *WGate* eine in den Webserver eingebundene DLL. Das *AGate* ist das Portal zur SAP-Applikationsebene und übernimmt im ITS die Hauptarbeit: Es liest die Eingabedaten aus der HTTP-Anfrage aus dem Internet/Intranet und sendet sie zur SAP-Applikationsebene; die zurückkommenden SAP-Bildschirme wandelt es anschließend in HTML-Seiten um.

AGate und *WGate*

Anwendungsentwicklern stehen mehrere Techniken zur Verfügung, mit denen sie Web-Anwendungen für ihre mySAP-Lösungen realisieren können. Die wichtigsten sind:

- ▶ Beim SAP GUI for HTML wird das Bild der SAP-Transaktion vom AGate generisch in HTML umgesetzt. Mit dem SAP GUI for HTML kann also jede SAP-Transaktion im Browser ausgeführt werden (mit einigen wenigen Einschränkungen). HTML-spezifische Programmierung ist nicht notwendig.
- ▶ Easy Web Transaction (EWT), welche die früher bekannten Internet Application Components (IACs) ablösen: Eine EWT besteht aus einer Transaktion im SAP-System und HTML-Vorlagen auf dem AGate des ITS. Wird aus dem Browser eine EWT aufgerufen, so wird vom AGate die zugehörige Transaktion im SAP-System gestartet. Das AGate meldet sich dabei über das DIAG-Protokoll an und verhält sich also dem Applikationsserver gegenüber wie ein Windows-GUI. Aus dem Bildschirm, den der Applikationsserver zurückliefert, liest das AGate die Bildschirm-inhalte, d. h. Feldinhalte, Tabelleninhalte, Schaltflächen etc., aus und setzt diese in die HTML-Vorlage an den entsprechenden Stellen ein.
- ▶ Bei Easy Web Transactions (EWTs) mit Flow-Logic wird der Dialogfluss auf den ITS verlagert, d. h., Programmtext im ITS (so genannte Flow-Logic-Dateien) steuern, welches Folgebild prozessiert wird. Damit erfolgt nur noch die Datenbeschaffung in der Applikations- und Datenbankebene, in der Regel über RFCs.
- ▶ Beim Web-RFC erfolgt die HTML-Aufbereitung in der Applikations-ebene. Wird eine Web-RFC-Anfrage im Browser ausgeführt, so geben WGate und AGate diese Anfrage als RFC direkt an die Applikations-ebene weiter. Diese beschafft die Daten und generiert bereits die komplette HTML-Seite und übergibt diese über die Schnittstelle des RFCs. AGate und WGate senden die fertige HTML-Seite an den Browser.

Die Techniken der Web-Anwendungen können danach klassifiziert werden, welcher Teil der Transaktion in der Applikationsebene und welcher im ITS ausgeführt wird. Tabelle 7.1 listet im Detail auf, welche Schritte bei den unterschiedlichen Web-Anwendungen durchlaufen werden.

	EWT ohne Flow-Logic	EWT mit Flow-Logic	SAP GUI for HTML	Web-RFC
Anmeldung an SAP-Applikationsebene	DIAG-Protokoll	RFC-Protokoll	DIAG-Protokoll	RFC-Protokoll
Businesslogik (Datenbeschaffung, Fortschreibung und Berechnungen)	Applikations- und Datenbankebene			

Tabelle 7.1 Programmiermodelle für Web-Anwendungen mit dem SAP ITS

	EWT ohne Flow-Logic	EWT mit Flow-Logic	SAP GUI for HTML	Web-RFC
Generierung der HTML-Seite	AGate (aus HTML-Vorlage und SAP-Bildschirm)		AGate (generisch aus SAP-Bildschirm)	Applikations-ebene
Ablauflogik (Dialogfluss)	Applikations-ebene	AGate (mit Flow-Logic-Datei)	Applikations-ebene	

Tabelle 7.1 Programmiermodelle für Web-Anwendungen mit dem SAP ITS (Forts.)

Bevor wir Konfigurations- und Tuningmöglichkeiten für den ITS diskutieren, betrachten wir den Ablauf eines Dialogschritts in einer Web-Anwendung im Detail.

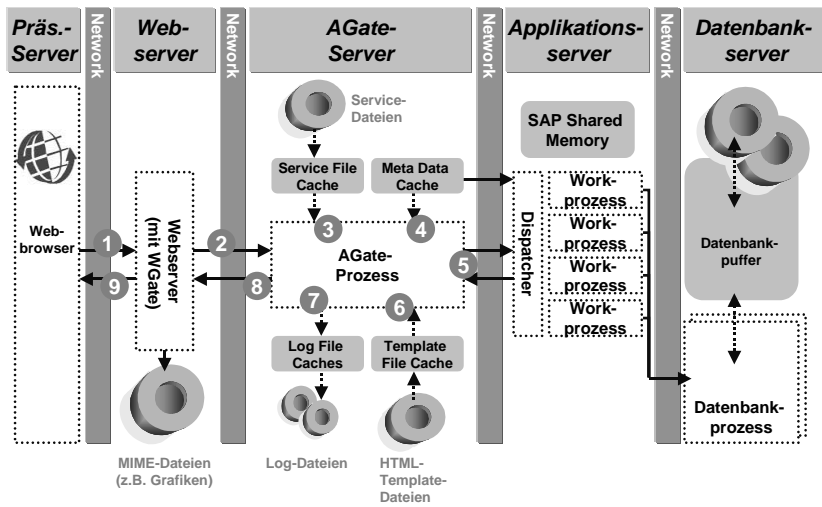


Abbildung 7.5 Transaktionsschritte im SAP ITS

Wenn Sie eine Easy Web Transaction über das Internet aufrufen wollen, so geben Sie typischerweise eine URL wie die folgende ein: `http://<sap-webserver>.<company.com>/scripts/wgate/<rcode>!`

Anmelden an den Applikations-server

Diese URL ist wie folgt zu interpretieren: `<company.com>` steht hier für Ihre Internetadresse und `<sapwebserver>` für den Namen Ihres zum SAP-System gehörigen Webservers. Der Zusatz `/scripts/wgate` zeigt dem Webserver, dass diese Anfrage an das WGate weitergeleitet werden soll. Dieses reicht die Anfrage direkt an das AGate des ITS weiter. In Abbildung 7.5 sind diese Schritte durch die Punkte 1 und 2 dargestellt.

Das AGate interpretiert `<rcode>` als den Namen einer Servicedatei `<rcode>.svc` und sucht in dem Verzeichnis seiner Servicedateien nach der entsprechenden Datei (Schritt 3 in Abbildung 7.5). In dieser Datei findet das AGate u. a. den Namen der SAP-Transaktion, die es aufruft. Nach der Namenskonvention sollte der Name der Servicedatei dem Transaktionscode entsprechen.



Typische Web-Transaktionen sind die Transaktionen des SAP Online Stores (Transaktionscode VW01, WW10) oder aus dem Personalwesen (Employee Self Services wie Zeiterfassung im Web, CATW, Reiseabrechnung im Web, PRWW). Die zugehörigen Servicedateien wären dann gemäß Namenskonvention `vw01.svc` usw.

Mit den Informationen aus der Servicedatei meldet sich das AGate an der Applikationsebene an (5), im Falle von EWTs ohne Flow-Logic und dem SAP GUI for HTML über das DIAG-Protokoll, im Falle von EWTs mit Flow-Logic und dem Web-RFC über das RFC-Protokoll (siehe Tabelle 7.1).

Die Datenbeschaffung für die Transaktion liest die Daten von der Datenbank und berechnet die definierten Abhängigkeiten. Wenn nötig, werden eingegebene Daten geprüft und auf der Datenbank geändert. Die Datenbeschaffung, -fortschreibung und -berechnung erfolgt grundsätzlich in der Applikations- und Datenbankebene.

Generieren der HTML-Seite

Aus dem SAP-Bildschirm, den die Applikationsebene an das AGate zurückgibt, generiert das AGate nun die HTML-Seite. Bei einer EWT sucht es in seinem Template-Verzeichnis nach der HTML-Vorlage (Template), die zu dem entsprechenden Service gehört, und füllt diese mit den Daten aus dem SAP-Bildschirm (6). Die generierte HTML-Seite wird an das WGate geschickt (7), das die HTML-Seite um eingebundene MIME-Dateien (Grafiken, Audio-Dateien) ergänzt und an den Browser sendet (9).

SAP GUI for HTML

Den *SAP GUI for HTML* rufen Sie wie folgt auf: `http://<sapwebserver>.<company.com>/scripts/WGate/webgui/!` Dieser aktiviert im AGate den Service `webgui.svc`. Nach der Eingabe der obigen Adresse bekommen Sie zunächst einen Anmeldebildschirm. Nachdem Sie sich an der Applikationsebene angemeldet haben, übersetzt der `webgui`-Service jede beliebige Transaktion in HTML. Sie können also mit Ihrem Browser arbeiten wie mit Ihrem SAP GUI for Windows oder SAP GUI for Java.

Web-RFC

Um eine *Web-RFC-Anfrage* zu starten, nutzen Sie folgende Syntax: `http://<sapwebserver>.<company.com>/scripts/WGate/webRFC/?_function=<function module> &<_variable1>=<value> ...` Sie erkennen in der Adresse zunächst, dass der Service `webRFC.svc` aufgerufen wird. Es folgen, jeweils

durch Fragezeichen getrennt, der Name des RFC-Bausteins, der aufgerufen werden soll, und die Variablen, die an den Baustein übergeben werden sollen. Ein Beispiel, das den Baustein »WWW_GET_REPORT« mit der Variable REPORT = »RSCONN01« aufruft, ist: `http://<sapwebserver>.<company.com>/scripts/WGate/webrfc/!?!_function=www_get_report &_report=rscconn01`. Im Falle des Web-RFCs wird in der Applikationsebene bereits die komplette HTML-Seite generiert. AGate und WGate geben diese also lediglich an den Browser weiter.

Jede dieser Techniken hat ihre Vor- und Nachteile. Wenn man ein Web-Entwicklungsprojekt startet, so muss man entscheiden, welche Technik die funktionalen Anforderungen an das Projekt am besten abdeckt. Die verschiedenen Techniken haben Auswirkungen auf Sizing, Konfiguration und Performance des ITS, was wir in den nächsten Abschnitten diskutieren wollen.

7.3.2 Web-Anbindung planen

Wie bereits eingangs erwähnt, sind alle Anwendungen der SAP Web-fähig, von wenigen Anwendungen, z. B. grafischen Planungswerkzeugen, abgesehen. Das bedeutet, es besteht die Möglichkeit, sich in einem Unternehmen strategisch auf den SAP GUI for HTML als einzige GUI-Lösung festzulegen. Die Alternative wäre, zweigleisig zu fahren und für einige Benutzer/Anwendungen den SAP GUI for HTML als GUI einzuführen und für andere den klassischen SAP GUI for Windows. Wir wollen in diesem Abschnitt beleuchten, von welchen Argumenten Sie sich dabei leiten lassen sollen.

Zunächst gibt es einige SAP-Anwendungen, die grundsätzlich nur mit dem SAP ITS laufen; dazu gehören alle Lösungen für das Internet oder Intranet, z. B. SAP Employee Self Service (SAP ESS), SAP Online Store, Enterprise Buyer Professional (E-Procurement). Für diese Lösungen ist die Entscheidung bereits getroffen. Diese Web-Anwendungen sind auf die Anforderungen des Web optimiert.

Web-Anwendungen

Anders sieht die Situation für viele klassische SAP R/3-Transaktionen aus. Diese laufen sowohl im SAP GUI for HTML als auch im SAP GUI for Windows. Um hier eine Entscheidung zu treffen, sollten Sie Performanceerwägungen mit einbeziehen.

Grundsätzlich ist es so, dass die Verwendung des SAP GUI for HTML einen zusätzlichen Aufwand (CPU-Zeit und Datentransfer) bedeutet, der

Performance
HTML-GUI und
Windows-GUI

zu einem Laufzeitnachteil im Vergleich zum *SAP GUI for Windows* führt. Im Einzelnen sind zu berücksichtigen:

- ▶ die CPU-Zeit für die Umwandlung der SAP-Bildschirme in HTML-Seiten durch den ITS
- ▶ der höhere Datentransfer zwischen ITS und Browser (im Vergleich zum Datentransfer zwischen Applikationsebene und *SAP GUI for Windows*)
- ▶ die Generierungszeit im Webbrowser; sie ist höher als die Bearbeitungszeit im *SAP GUI for Windows*

Um wie viel höher die Antwortzeit für einen Benutzer mit dem *SAP GUI for HTML* im Vergleich zum *SAP GUI for Windows* schließlich ist, hängt von den verwendeten Funktionen und der Hardware ab. Für die CPU-Zeit auf dem ITS und dem Frontend sollte man mit einem Zusatzaufwand in der Größenordnung von einer Sekunde rechnen. Zum höheren Datenverkehr können Sie von folgenden Überlegungen ausgehen: Zu Recht bezeichnet die SAP ihren *SAP GUI for Windows* als »Ultra-thin client«, da der Datenverkehr zwischen Applikationsebene und GUI sehr gering ist, ca. 1–2 kByte pro Bildwechsel für *SAP R/3 4.0* und ca. 3–5 kByte pro Bildwechsel für *SAP R/3 4.6* bei Transaktionen mit Controls. Der Internetstandard dagegen ist um ein Vielfaches höher. Zum Aufbau von HTML-Seiten werden typischerweise zwischen 20 und 200 kByte übertragen. Die Produkte der SAP liegen bei ihrem Datenverkehr hier am unteren Ende, d. h. bei ca. 20–40 kByte. In einem LAN mit hoher Bandbreite macht sich dieser erhöhte Datenverkehr in der Laufzeit praktisch nicht bemerkbar. In einem WAN mit niedriger Bandbreite kann dieser Unterschied dagegen zu massiven Performancenachteilen des *SAP GUI for HTML* im Vergleich zum *SAP GUI for Windows* führen. Wir empfehlen daher Evaluierungsmessungen, wie im Abschnitt 7.6.1, »Laufzeitanalyse im Webbrowser durchführen«, beschrieben.

7.3.3 ITS konfigurieren

Aufgrund seiner Architektur ist der ITS praktisch beliebig skalierbar. In diesem Abschnitt stellen wir dar, wie Sie den ITS optimal an Ihre Lastanforderungen anpassen.

Für ein Entwicklungs-, Test- oder ein kleines Produktivsystem ohne spezielle Hochverfügbarkeitsanforderungen ist es möglich, alle Ebenen der mySAP-Technologie (Präsentationsebene, Internetebene, Applikationsebene, Datenbankebene) auf einem Rechner zu installieren.

Steigen die Anforderungen an Verfügbarkeit und Durchsatz, so ist es sinnvoll, die einzelnen Ebenen auf mehrere Rechner zu verteilen. (Auch Sicherheitsaspekte spielen hier eine Rolle, wie wir weiter unten erläutern werden).

Der erste Schritt einer Skalierung ist, für die Internetebene (ITS und Webserver) separate Rechner zur Verfügung zu stellen und diese damit von der SAP-Applikationsebene und der Datenbankebene zu trennen.

ITS-Instanzen konfigurieren

Wie die Applikationsebene ist auch der ITS in Instanzen organisiert. Auf einem Rechner können also mehrere *ITS-Instanzen* installiert werden, die separat administriert werden. Theoretisch ist es möglich, dass die unterschiedlichen Services einer ITS-Instanz auf verschiedene SAP-Systeme zeigen. Für den produktiven Einsatz ist dies allerdings aus Gründen der Administrierbarkeit nicht empfohlen. Bauen Sie Ihre Systemlandschaft so auf, dass eine ITS-Instanz immer auf ein SAP-System zeigt.

ITS-Instanzen

Die Anmeldung der ITS-Instanz an die Applikationsebene erfolgt dabei entweder über eine dezidierte Instanz (d. h. über die Angabe des Applikationsservers und der Instanznummer) oder über eine *Logon-Gruppe* (Angabe des Message-Servers und einer Logon-Gruppe). Um eine optimale Verfügbarkeit und Lastverteilung zu erreichen, sollten Sie die zweite Alternative wählen. Diese Einstellung wird in den Servicedateien auf dem AGate vorgenommen.

Anmeldung über Logon-Gruppe

Je nach Anforderungen an den Durchsatz können Sie mehrere *ITS-Instanzen* (die auf unterschiedliche SAP-Systeme zeigen können) auf einem Rechner betreiben. Umgekehrt können Sie auch zu einem SAP-System mehrere ITS-Instanzen betreiben, die Sie auf mehrere Rechner verteilen. Um eine optimale Hochverfügbarkeit zu gewährleisten, sind mindestens zwei Rechner für die Internetebene notwendig, auf denen jeweils eine ITS-Instanz läuft, die über eine Logon-Gruppe auf mindestens zwei SAP-Instanzen zeigt, die wiederum auf unterschiedlichen Rechnern konfiguriert sind.

Skalierbarkeit ITS-Instanzen

Aus Sicherheitsgründen muss der Zugriff auf das Filesystem des AGate in jedem Fall geschützt werden, denn wer Zugriff auf das AGate hat, kann einen Service definieren und damit jeden für die Internetbenutzung freigegebenen Funktionsbaustein im Applikationsserver ausführen. Der beste Weg, das AGate zu schützen, ist eine *Firewall* zwischen AGate und WGate, da hier der Datenfluss in der Hand der SAP ist und sehr gut kontrolliert werden kann. Eine Firewall vor dem Webserver (WGate) gilt

Firewall zwischen AGate und WGate

unter Sicherheitsexperten als weit weniger zuverlässig. Als optimale Konfiguration gilt, Webserver (mit WGate) und AGate auf unterschiedlichen Rechnern zu betreiben und diese über eine Firewall zu trennen. Diese Konfiguration ist »Pflicht«, wenn der ITS Ihr SAP-System mit dem Internet verbindet. Bei Intranet-Projekten muss man entscheiden, ob dieser hohe Sicherheitsstandard ebenfalls angebracht ist.

Eine ITS-Konfiguration, die hinsichtlich Verfügbarkeit, Performance und Sicherheit optimiert ist, besteht daher jeweils aus doppelt ausgelegtem Webserver (mit WGate), ITS-AGate und SAP-Instanz, so dass bei Ausfall eines Rechners die Last von dem verbleibenden Rechner übernommen werden kann. Parallel sollte die Datenbank und die Applikationsebene mit dem Enqueue-Server über eine entsprechende Hochverfügbarkeitslösung auf Betriebssystemebene (Switch-over-Lösung) geschützt sein.

Workprozesse, Sessions und Caches konfigurieren

Workprozesse und Sessions Innerhalb einer ITS-Instanz werden Anfragen vom Dispatcher (Mapping-Manager-Prozess) einem ITS-Workprozess (Work Thread) zuordnet. Der ITS-Workprozess kopiert die Daten der Anfrage in einen spezifischen Speicherbereich, die so genannte Benutzersession. Die Anzahl der ITS-Workprozesse und Sessions kann im ITS eingestellt werden. Eine Übersicht über performancerelevante ITS-Parameter finden Sie in Anhang C.

Die Architektur des ITS ähnelt also einer SAP-Applikationsinstanz, die ebenfalls über einen Dispatcher und mehrere Workprozesse verfügt, die auf globale Benutzerkontexte (Sessions) zugreifen. Allerdings verfügt der ITS nicht über eine Dispatcher-Queue. Sind alle ITS-Workprozesse belegt, so sendet der Dispatcher eine Fehlermeldung an den Browser zurück (während eine Applikationsinstanz in diesem Fall die Anfrage in der Dispatcherqueue »parkt«).

Cache-Mechanismen Der ITS verfügt über *Cache-Mechanismen*, um die Performance zu optimieren. So werden die Servicedateien, Schnittstellen von SAP-Funktionsbausteinen, die vom ITS aufgerufen werden, HTML-Vorlagen (Templates), Modulaufrufe (bei der Verwendung von Flow-Logic) und Log-Einträge im Hauptspeicher gehalten. Über Parameter können Sie diese Cache-Mechanismen ein- und ausschalten (Anhang C).

Datenkomprimierung Um die übertragene Datenmenge zu reduzieren, sollten Sie in der ITS-Service-Datei die Komprimierung einschalten. Die Beschreibung der Parameter und den zugehörigen SAP-Hinweis finden Sie in den Anhängen C und G. Die Datenkomprimierung durch den ITS kann allerdings nur dann erfolgen, wenn der Webbrowser dies zulässt.

Folgenden Faktoren können die Performance des ITS limitieren:

- ▶ Hardwareengpässe, d. h. ein CPU- oder Hauptspeicherengpass (hohe Paging-Aktivität)
- ▶ Anzahl der Workprozesse und Benutzersessions: Werden zu wenig von diesen konfiguriert, so kommt es zu Fehlermeldungen, da der ITS die an ihn gerichteten Anfragen nicht mehr bearbeiten kann.
- ▶ Beschränkung des adressierbaren Speichers: Aufgrund der aktuellen Windows-NT-Architektur ist der adressierbare Speicher auf 2 GByte pro AGate-Prozess beschränkt. Fordert der AGate-Prozess während der Laufzeit mehr Speicher an, kommt es zu Fehlermeldungen, die in der Protokolldatei des AGates und des Betriebssystems überwacht werden können.
- ▶ Beschränkungen durch Netzwerkverbindungen: Der AGate-Prozess kann maximal 4096 Benutzersessions zur Applikationsebene aufbauen. Dies beschränkt die maximale Anzahl der angemeldeten Benutzer pro AGate und SAP-Applikationsinstanz. Eventuell müssen daher mehrere AGate und SAP-Applikationsinstanzen konfiguriert werden.
- ▶ Falsche Konfiguration der Cache-Mechanismen: Sind die Cache-Mechanismen nicht richtig aktiviert, wird unnötig viel Dateiverkehr generiert.

Um Hardwareengpässe im Vorfeld einer Installation zu vermeiden, sollten Sie sich vor Produktivstart von Ihrem Hardwarepartner einen Sizing-Vorschlag erstellen lassen. Informationen zum Sizing des ITS finden Sie im SAP Service Marketplace unter <http://service.sap.com/sizing>. Einstellungsempfehlungen zu Workprozessen, Sessions und Cache-Mechanismen finden Sie in Anhang C.

Treten zur Laufzeit Engpässe auf, so können diese über Analysewerkzeuge lokalisiert werden. Werkzeuge und die zugehörigen Methoden sind in den nächsten Abschnitten beschrieben.

7.3.4 ITS-Administrationswerkzeuge

Es gibt mehrere Möglichkeiten, eine ITS-Instanz zu administrieren und zu überwachen. Zum einen können Sie dies über die Administrationsinstanz des ITS tun, die Sie aus einem Browser über folgende URL-Adresse starten: `http://<sapwebserver>.<company.com>:<portnumber>/scripts/wgate/admin/!`, wobei `<company.com>` hier für Ihre Internetadresse steht, `<sapwebserver>` für den zum ITS gehörigen Webserver und `<portnumber>` für den TCP-Port, auf den die Administrationsinstanz »hört«.

Mit der Administrationsinstanz können Sie:

- ▶ ITS-Instanzen starten und stoppen
- ▶ ITS-Parameter ändern
- ▶ die Protokolldateien des ITS (Fehler- und Performanceprotokolle) auswerten
- ▶ die aktuelle Performance und Auslastung ermitteln

Die Protokolldateien, die Sie überwachen können, sind das AGate-Protokoll (*AGate.trc*), das Dispatcherprotokoll (*Mmanager.trc*), das Performanceprotokoll (*PERFORMANCE.LOG*), das Lastverteilungsprotokoll (*LOAD-STAT.LOG*) und das Zugriffsprotokoll (*ACCESS.LOG*).

Globaler CCMS-Überwachungsmonitor

Alle Performancedaten, die Sie im proprietären ITS-Administrationswerkzeug finden, sind auch in den zentralen CCMS-Alert-Monitor (Transaktionscode RZ20) eingebunden. Die ITS-Instanzen mit dem CCMS-Alert-Monitor zu überwachen, hat die Vorteile, dass Sie alle Performancedaten in einem Werkzeug und die gesamte Funktionalität der Alert-Überwachung und -verwaltung zur Verfügung haben. In SAP-Hinweis 418285 finden Sie die Details, um ITS-Instanzen in den CCMS-Alert-Monitor einzubinden. Abbildung 7.6 und Tabelle 7.2 zeigen die ITS-Performancedaten und ihre Beschreibung.

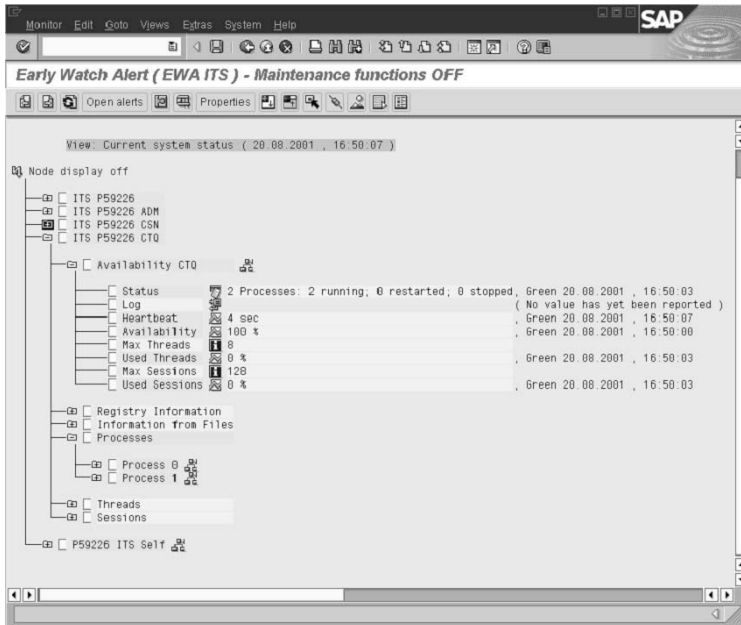


Abbildung 7.6 ITS-Performancedaten im zentralen CCMS-Alert-Monitor

Eintrag im zentralen CCMS-Alert-Monitor	Bedeutung
Status	Information über die gestarteten AGate-Prozesse
Log	Instanznummer des AGate
Heartbeat	»Lebenszeichen« des SAP ITS
Availability	Verfügbarkeit der ITS-Instanz in den letzten 15 Minuten
Max. Threads	Anzahl insgesamt zur Verfügung stehender ITS-Workprozesse (Work Threads)
Used Threads	Anzahl benutzter ITS-Workprozesse (in Prozent)
Max. Sessions	Anzahl insgesamt zur Verfügung stehender Sessions
Used Sessions	Anzahl benutzter Sessions (in Prozent)
Hits*	Anzahl der Zugriffe (Hits) pro Sekunde
Hits A*	Anzahl der aktuellen offenen Zugriffe auf den Applikationsserver
TAT*	Turnaround time: Mittlere Antwortzeit für Zugriff auf den Applikationsserver und Generierung der HTML-Seite
Up time*	Zeit seit Start der ITS-Instanz
User time*	User-CPU-Anteil auf dem ITS-Server
Kernel time*	System-CPU-Anteil auf dem ITS-Server (in %)

Tabelle 7.2 ITS-Perfomancedaten im zentralen CCMS-Alert-Monitor. Die mit einem Stern gekennzeichneten Werte sind in Abbildung 7.6 nicht dargestellt; sie befinden sich im Ast unter »Processes«.

Detailliertere Daten zur Hardwareauslastung der ITS-Server können Sie aus einem SAP-System über den *globalen Betriebssystemmonitor* (Transaktionscode OS07) überwachen (siehe auch Abschnitt 2.2, »Hardwareanalyse«). Dazu müssen Sie die Datenkollektorprogramme SAPOSCOL und RFCOSCOL sowie ein SAP-Gateway auf den Rechnern starten, auf denen ITS-Instanzen laufen. Über die Details der Installation informiert SAP-Hinweis 202934.

Globaler Betriebssystemmonitor

7.3.5 Engpassanalyse für den ITS durchführen

Neigt sich die Anzahl der freien *ITS-Workprozesse* dem Wert Null zu, so kommt es zu Fehlermeldungen, da der ITS die an ihn gerichteten Anfragen nicht mehr bearbeiten kann. In diesem Fall ist zu prüfen, ob das Problem behoben wird, wenn die Anzahl der ITS-Workprozesse vergrößert wird. Im Prinzip gilt hier allerdings dieselbe Regel wie auch auf der Applikationsebene: Die Anzahl der ITS-Workprozesse soll nur dann

Problem: Alle ITS-Workprozesse belegt

erhöht werden, wenn noch Hardwareressourcen frei sind. Das Problem der belegten Workprozesse kann aber auch durch einen Programmierfehler verursacht sein, das bedeutet, dass entweder das Programm auf dem AGate (Business-HTML bzw. Flow-Logic) oder das Programm auf dem Applikationsserver (ABAP) ineffizient programmiert wurde und unnötig viele Ressourcen braucht (bzw. sich im Extremfall in einer Endlosschleife verfängt).

Problem: Zu wenig adressierbarer Speicher

Anhand der Anzeige des freien adressierbaren (virtuellen) Speichers können Sie prüfen, ob dieser zum Engpass wird. Insgesamt kann auf Windows NT pro AGate-Prozess 2 GByte adressiert werden. Wird dieses Limit erreicht, so können Sie das Problem lösen, indem Sie mehrere AGate-Prozesse pro ITS-Instanz starten. Ein Engpass an adressierbarem Speicher führt außerdem zu einem Fehlereintrag im AGate-Protokoll (*AGate.trc*).

Problem: Hardwareengpass

Einen *Hardwareengpass* erkennen Sie an einer hohen CPU-Auslastung und einer hohen Paging-Rate (siehe auch nächster Abschnitt).

Problem: Stillstand wegen mangelndem Plattenplatz

Steht kein Festplattenplatz mehr zur Verfügung, hängt sich der ITS auf. Dies kann z. B. passieren, wenn Protokolldateien über lange Zeit nicht wie empfohlen reorganisiert werden oder vergessen wurde, den Trace-Level nach Erstellung einer detaillierten Laufzeitanalyse zu reduzieren.

7.4 Business Server Pages (BSP) und Internet Communication Manager (ICM)

In diesem Abschnitt stellen wir die grundlegende Architektur und die Methoden der Performanceüberwachung von *Business Server Pages* (BSPs) vor. Ebenfalls eingeführt wird in diesem Abschnitt der *Internet Communication Manager* (ICM), der wie schon erwähnt, nicht nur bei BSPs sondern auch bei Web-Anwendungen auf der Basis von Java Server Pages (JSPs) und Java Servlets zum Einsatz kommt.

7.4.1 Grundlagen von SAP Business Server Pages

Business Server Pages sind eine Möglichkeit, SAP-Web-Anwendungen zu schreiben. BSPs bestehen aus einzelnen Webseiten, die in HTML und ABAP als Scriptsprache geschrieben sind. BSPs werden vollständig in der SAP Development Workbench entwickelt, d. h., es gibt nicht, wie beim SAP ITS oder der SAP J2EE Engine, Bestandteile von BSPs, die in Dateisystemen außerhalb der Datenbank des SAP-Systems abgelegt werden. Die HTML-Seiten werden zur Laufzeit von einem SAP-Workprozess generiert. Die SAP Development Workbench und die SAP-Laufzeitumgebung wur-

den dazu zu SAP-Basis 6.10 um das so genannte *Internet Communication Framework* (ICF) und den *Internet Communication Manager* (ICM) erweitert. Beide sind Bestandteil der Auslieferung und der Installation des SAP Web Application Servers.

Eine BSP-Anwendung in einem SAP-System rufen Sie auf, indem sie folgende URL in Ihrem Webbrowser eingeben: `http://<sapserver>:<port>/sap/bc/bsp/sap/<bsp_anwendung>/<seite>`.

Anmelden an den
Applikations-
server

Dabei steht `<sapserver>` für den Namen eines Anwendungsservers, auf dem das SAP-System läuft. Dieser muss immer voll qualifiziert angegeben werden, also z. B. `sapapp1.stadt.company.com` und nicht nur `sapapp1`. `<port>` ist der TCP/IP-Port auf dem der Anwendungsserver hört. Dieser Wert steht beim SAP Web AS 6.10 standardmäßig auf 8080; beim SAP Web AS 6.20 steht der Wert aus Sicherheitsgründen zunächst auf Null und muss vom Administrator auf den gewünschten Wert geändert werden. `<bsp_anwendung>` ist der Name der BSP-Anwendung und `<seite>` der Name der Seite innerhalb der Anwendung.

Jeder SAP Web Application Server wird mit einer kleinen Anzahl von Testanwendungen und Tutorials ausgeliefert. Um eine BSP beispielhaft auszuführen, geben Sie in Ihren Internet-Browser die folgende URL ein: `http://<sapserver>:8080/sap/bc/bsp/sap/tutorial_2/default.htm`. Dabei steht `<sapserver>` für den Namen des Servers, auf dem der SAP Web Application Server läuft. Daraufhin werden Sie in einem Bildschirmfenster nach Ihrem Benutzernamen und Ihrem Passwort gefragt. Geben Sie dieses korrekt ein, so erscheint im Browser eine kleine Web-Anwendung »Welcome to our Online Book Catalog!«.



Ein Reihe von BSP-Testanwendungen werden mit jedem SAP Web Application Server ausgeliefert; dazu gehören Testanwendungen, deren Name mit »IT« beginnt und Lernanwendungen, deren Name mit »tutorial« beginnt.

Im SAP Web AS 6.10 sind die ausgelieferten BSP-Anwendungen automatisch aktiv. Ab SAP Web AS 6.20 werden BSP-Anwendungen aus Sicherheitsgründen deaktiviert ausgeliefert und müssen explizit aktiviert werden (SAP-Hinweis 517484).



Abbildung 7.7 zeigt die Entwicklung einer BSP-Anwendung. Auf der linken Seite der Abbildung erkennen Sie die HTML-Seite »Your Book Search Results«, wie sie in einem Browser dargestellt wird. Auf der rechten Seite ist das zugehörige Coding dargestellt. Eine BSP-Anwendung besteht zunächst aus der eigentlichen Webseite, die in HTML geschrieben ist und

Entwicklungsum-
gebung für BSPs

in der ABAP-Coding zur Datenaufbereitung in so genannten *Tags* eingebettet ist (z. B. ABAP-Loops zur Darstellung von Tabellen). Das Programmiermodell ist also vergleichbar zu Java Server Pages (JSPs) bei denen Java-Anwendungen in den HTML-Text eingebunden sind.

Internet-Seite

BSP-Anwendung in der SAP Development Workbench

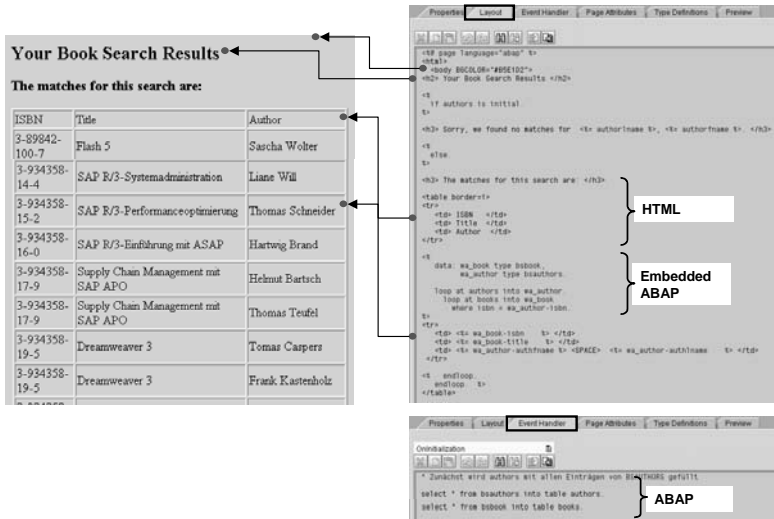


Abbildung 7.7 zeigt auf der linken Seite eine fertige HTML-Seite »Your Book Search Results« und auf der rechten Seite den dazugehörigen Programmtext in HTML und ABAP.

Weiterhin umfasst eine BSP-Anwendung sogenannte *Events*. Dies sind in ABAP geschriebene Programmteile, die zu bestimmten Zeitpunkten, z. B. zur Initialisierung der Webseite oder nach der Dateneingabe prozessiert werden. Diese Events erlauben die Bildsteuerung der Seite und die Datenbeschaffung (z. B. Zugriffe auf die Datenbank). Insgesamt kann eine komplette BSP-Anwendung aus mehreren solcher Webseiten bestehen.

Die SAP-Entwicklungsumgebung (SAP Development Workbench, Transaktionscode SE80) wurde um die entsprechenden Funktionen zur Entwicklung von BSPs erweitert. Um eine BSP in einem SAP-System anzuzeigen, gehen Sie in der Development Workbench wie folgt vor:

1. Wählen Sie die Schaltfläche **Repository Browser**.
2. In dem unter der Schaltfläche befindlichen Auswahlfeld wählen Sie die Kategorie **BSP-Applikation** aus. Geben Sie in dem darunter liegenden Eingabefeld **TUTORIAL_2** ein, und bestätigen Sie die Eingabe.

3. In dem nun erscheinenden Baum finden Sie im Teilbaum »Seiten« nun die HTML-Seiten, die zur entsprechenden BSP-Anwendung gehört. Wählen Sie beispielsweise die Seite »default.htm« aus. Nun befinden Sie sich in der Entwicklungsumgebung für eine spezielle BSP.

- ▶ Um sich eine Vorschau auf die entsprechende Webseite anzuschauen, wählen Sie den Reiter **Vorschau**.
- ▶ Um den Programmcode der Webseite anzuzeigen, wählen Sie den Reiter **Layout**.
- ▶ Um die zu einer BSP gehörigen Events anzuzeigen, d. h. der Programmcode, der zum Beispiel beim Aufruf einer Seite oder bei einer Eingabe ausgeführt wird, wählen Sie den Reiter **Events**.

Die bisherige Darstellung beschreibt das einfache BSP-Programmiermodell. Mit dem SAP Web AS 6.20 wird das *Model View Controller (MVC) Design Pattern* als erweitertes Programmiermodell eingeführt. Ursprünglich im Umfeld von Smalltalk-80 entwickelt, findet es weite Verbreitung in der Gemeinde der Java-, ColdFusion- und PHP-Entwickler. Es erlaubt die strikte Trennung von Datenmodell (»Model«), Darstellung der Daten an der Oberfläche (»View«) und Verarbeitungskontrolle (»Controller«) und wird von der SAP dringend zur Verwendung empfohlen. Für die Methoden der Performanceanalyse macht es grundsätzlich keinen Unterschied, ob Sie das einfache BSP-Programmiermodell oder das MVC Design Pattern verwenden.

**MVC Design
Pattern**

Beim Aufruf einer BSP werden die in Abbildung 7.8 dargestellten Schritte durchlaufen, um eine HTML-Seite zu generieren. Der Webbrowser sendet seine Anfrage zunächst an den Internet Communication Manager (ICM) (1). Dieser prüft, ob die Browseranfrage mithilfe der in seinem Cache gespeicherten Informationen beantwortet werden kann (2). Ist dies nicht der Fall, so übergibt er die Anfrage an den SAP-Dispatcher (4), nachdem er die Daten der Anfrage in so genannte *Memory Pipes* gespeichert hat (3). Memory Pipes gehören zum Shared Memory der SAP-Instanz und dienen der Kommunikation zwischen dem ICM auf der einen und den SAP-Workprozessen auf der anderen Seite. Sofern ein Dialogworkprozess für die Bearbeitung zur Verfügung steht und der Dispatcher die Anfrage nicht in der Queue zwischenparken muss (5), übergibt der Dispatcher die Anfrage einem Workprozess (6). Dieser liest die benötigten Daten aus den Memory Pipes (7) und bearbeitet die Anfrage (8). Nachdem der Workprozess die BSP bearbeitet und die Webseite erstellt hat, stellt er die Daten in die Memory Pipe (9) und übergibt die Kontrolle zurück an den ICM. Dieser sendet die fertige HTML-Seite an den Webbrowser (10–12).

**Generieren der
HTML-Seite**

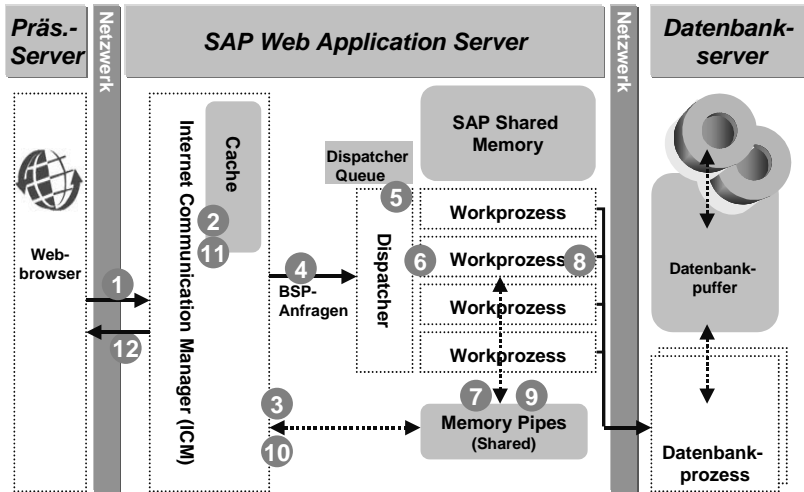


Abbildung 7.8 Ablauf eines Transaktionsschrittes im SAP Web AS beim Aufruf einer BSP. Weitere Erläuterungen im Text.

7.4.2 ICM konfigurieren und überwachen

Der ICM wird über den SAP-Profilparameter `rdisp/start_icman = true` aktiviert. (Im Standardfall ist der ICM nach der Installation des SAP Web Application Servers aktiv).

ICM-Threads,
Queue, Verbindungen

Der ICM vermittelt nur die Kommunikation zwischen Browser und SAP-Workprozessen, er verarbeitet weder Business-Logik noch generiert er die Webseiten, d. h. die Last, die er bewältigen muss, ist im Vergleich zu den SAP-Workprozessen eher gering. Gegenüber dem Betriebssystem ist der ICM ein Prozess (`icman.exe`), intern ist der ICM skalierbar ausgelegt, d. h. er verfügt über mehrere Threads.

Um die Auslastung des ICM zu überwachen, rufen Sie den ICM-Monitor (Transaktionscode `SMICM`) auf. Im Eingangsbildschirm des Monitors erkennen Sie den Status der einzelnen Threads des ICM. Im oberen Teil des Bildschirms finden Sie statistische Informationen über den Status der Threads (in der Zeile »Erzeugte Threads«), der Verbindungen (Zeile »Benutzte Verbindungen«) und der Queue (Zeile »Benutzte Queue-Einträge«). Einen Engpass im ICM erkennen Sie, wenn für mindestens einen dieser drei Parameter der Wert »peak« gleich dem Wert »maximal« ist.

Eine Übersicht über performancerelevante SAP-Profilparameter, die den ICM, die Memory Pipes und den Cache konfigurieren, finden Sie im Anhang C.



Um die Auslastung der Memory Pipes zu überwachen, wählen Sie im ICM-Monitor die Funktion **Springen • Memory Pipes • Daten** anzeigen. Hier finden Sie unter anderem Informationen über die Anzahl der Memory-Pipe-Puffer (Feld **Total #Mpi Buffer** und über deren Auslastung (Feld **Peak buffer usage**). Einen Engpass erkennen Sie, wenn der Wert »peak« gleich dem Wert »maximal« ist.

Memory Pipes

Der ICM verfügt über einen Cache, in dem er Webseiten oder Teile von Webseiten, wie z. B. Bilddateien speichert. Eine Übersicht über die im Cache befindlichen Daten finden Sie im ICM-Monitor unter der Funktion **Springen • HTTP Server Cache • Anzeigen**. Statistische Daten über Füllgrad und Effizienz des Caches finden Sie unter **Springen • HTTP Server Cache • Statistik** anzeigen. Interessante Werte sind hier unter anderem die absolute Größe des Caches (Feld **Cache-Größe (Bytes)**) und die belegte Größe im Feld **Belegter Cache-Speicher (Bytes)**.

Cache-Mechanismen

Um die Lastverteilung zwischen mehreren SAP-Instanzen zu erreichen, bietet die SAP zwei Verfahren an:

Skalierbarkeit und Logon-Balancing

Bei dem ersten Verfahren erfolgt die Verteilung der HTTP-Anfragen über eine Redirection der URL. Dazu sendet der Browser seine Anfrage zunächst an den Message-Server des SAP-Systems. Dieser nimmt die Lastverteilung vor und sendet an den Browser die Adresse desjenigen Anwendungsservers zurück, auf dem er sich anmelden soll. Ist der Benutzer mit seinem Browser einmal an einem Anwendungsserver angemeldet, so verbleibt er dort bis zum Ende seiner Sitzung. Dieses Verfahren ähnelt dem Logon-Balancing mit dem SAP GUI.

Alternativ können Sie den SAP Web Dispatcher verwenden. Dabei handelt es sich um eine separate Software der SAP für das Web-Dispatching. Dies wird von der SAP als Standardmethode empfohlen. Natürlich können Sie für die Lastverteilung auch Web Dispatcher von Drittanbietern verwenden, die sich als software- oder hardwarenahe Lösungen präsentieren.

Eingangs wurde erwähnt, dass es zur Kommunikation zwischen Webbrowser und SAP Web Application Server keines separaten Webservers bedarf. Dies ist zwar theoretisch richtig und für ein Entwicklungssystem oder für ein Produktivsystem, welches nur in einem firmeninternen Intranet verwendet wird, auch in der Praxis anwendbar. Scharfe Sicherheitsrestriktionen erfordern allerdings zwischen Internet und SAP Web Appli-

Firewall

cation Server eine sogenannte »demilitarisierte Zone« (DMZ), die durch einen Webserver gebildet und durch zwei Firewalls begrenzt wird. In der DMZ läuft unter anderem der SAP Web Dispatcher. Details finden Sie u. a. im SAP-Netzwerkguide (siehe Anhang G).

Zentraler CCMS-Überwachungsmonitor

Das Monitoring des ICM kann über den zentralen CCMS-Überwachungsmonitor (Transaktionscode RZ20) erfolgen, in den die wichtigsten Performancekennzahlen über Treads, Verbindungen, Queue und Memory Pipes eingebunden sind. Die Beschreibung der Monitoringobjekte finden Sie in Tabelle 7.3.

Eintrag im zentralen CCMS-Alert-Monitor	Bedeutung
NoOfThreads	Anzahl der aktuell erzeugten ICM-Treads
PeakNoOfThreads	Anzahl der aktuell maximal ICM-Treads
NoOfConnections	Anzahl der aktuellen ICM-Verbindungen
PeakNoOfConnections	Maximale Anzahl der ICM-Verbindungen in der Vergangenheit
QueueLen	Aktuelle Queue-Länge
PeakQueueLen	Maximale Queue-Länge in der Vergangenheit
MPISizeTotal	Größe des Speicherbereiches für Memory Pipes
MPIBufferCount	Anzahl der verfügbaren Memory-Pipes
PeakMPIBufUsed	Maximale Anzahl der in der Vergangenheit genutzten Memory-Pipes

Tabelle 7.3 ICM-Performancekennzahlen im zentralen CCMS-Alert-Monitor. Die Kennzahlen finden Sie im Alert-Baum einer SAP-Instanz unter R3Services • ICM • Allgemein.

7.4.3 Performanceanalyse von BSPs durchführen

Zur Performanceanalyse von BSPs können Sie alle Verfahren und Werkzeuge verwenden, die Sie bereits zur Performanceanalyse von SAP-GUI-Anwendungen kennen. Einige Besonderheiten sind jedoch zu beachten, die wir hier besprechen.

Generelle Performanceanalyse

Wenn Benutzer über schlechte Performance klagen, so starten Sie, wenn das Performanceproblem aktuell besteht, den Workprozessmonitor (Transaktionscode SM50 bzw. SM66). Sind alle Workprozesse belegt, so führen Sie eine Engpassanalyse gemäß Kapitel 2 durch um die Ursache einzugrenzen.

zen und z. B. Benutzer und Programm zu identifizieren, die das Problem verursachen.

Können Sie im Workprozessmonitor keine langlaufenden Prozesse beobachten, stellen die Benutzer aber trotzdem ein Performanceproblem fest, so starten Sie den ICM-Monitor (Transaktionscode SMICM). Sind hier alle ICM-Threads belegt, so kann möglicherweise das Problem behoben werden, indem Sie die Anzahl der ICM-Threads erhöhen.

Um die Performance in der Vergangenheit zu bewerten, so rufen Sie den Workload-Monitor (Transaktionscode ST03N) auf. Der Monitor weist nun neben den bekannten Tasktypen »Dialog«, Background«, »Update« etc. die neuen Tasktypen »HTTP«, »HTTPS« und »SMTP« aus (natürlich nur, sofern entsprechende Anfragen vom System bearbeitet wurden). Anhand dieser Tasktypen können Sie einfach die Aktivität und die Antwortzeiten der entsprechenden Anfragen bewerten.

Um die Performanceanalyse mit dem Workload-Monitor durchführen zu können, muss allerdings das Schreiben der Performancestatistiken aktiviert werden. Im Auslieferungszustand schreibt der SAP-Kernel für HTTP-, HTTPS- und SMTP-Anfragen keine Statistiksätze. Um die Statistiken zu aktivieren, setzen Sie den SAP-Profilparameter **rdisp/no_statistic** auf den Leer-Wert. (D. h., Sie fügen in die Profilparameterdatei die Zeile »rdisp/no_statistic =« ein. Der Kernel-Standardwert ist »PLUGIN«.) Diesen SAP-Profilparameter können Sie bei laufendem System ändern, in dem Sie wie folgt vorgehen:

1. Wählen Sie zunächst im Systemlastmonitor (Transaktionscode ST03N) die Rolle **Experte** (Schaltfläche in der linken oberen Ecke).
2. Folgen Sie dann im Baum den Punkten **Kollektor & Perf. Datenbank, Statistiksätze & -datei, Online Parameter, Dialogschrittstatistik**.
3. Es wird die Tabelle »Laufzeitparameter der Statistiksammlung« angezeigt. Löschen Sie in der Spalte **rdisp/no_statistic** den Wert »PLUGIN«.
(Sofern sich in der Spalte **rdisp/no_statistic** der Wert »PLUGIN« befindet, werden keine Statistiksätze geschrieben. Ist die Spalte leer, so ist das Schreiben der Statistik aktiviert.)
4. Aktivieren Sie Ihre Änderung mit der Schaltfläche **Werte aktivieren**.

Als Ergebnis dieser Aktion finden Sie nun im Workload-Monitor neben den Ihnen bekannten Tasktypen »Dialog«, Background«, »Update, etc. die Tasktypen »HTTP«, »HTTPS« und »SMTP« (natürlich nur, sofern entsprechende Anfragen vom System bearbeitet wurden). Die Online-Akti-

vierung ist nur bis zum nächsten Neustart der entsprechenden SAP-Instanz aktiv. Um die Statistiken permanent einzuschalten, müssen Sie, wie oben beschrieben, den Parameter in der Konfigurationsdatei ändern.

Im Transaktionsprofil des Workload-Monitors können Sie eine Detailanalyse der aufgerufenen Web-Anwendungen durchführen. Sie können im Workload-Monitor wählen, in welcher Granularität er die Performancestatistiken für Webseiten im Transaktionsprofil darstellen soll:

1. Wählen Sie dazu im Systemlastmonitor (Transaktionscode ST03N) die Rolle **Experte** aus (Schaltfläche in der linken oberen Ecke).
2. Folgen Sie dann im Baum den Punkten **Kollektor & Perf. Datenbank, Systemlast-Kollektor, Zu erzeugende Statistik, Dialogschrittstatistik**.
3. Wählen Sie dann eine der folgenden Optionen aus:
 - ▶ **Auflösung vollständig nach Anwendung und Seite:** Die Performancestatistiken werden pro Webseite erstellt, d. h. im Transaktionsprofil finden Sie für jede prozessierte Webseite einen Eintrag.
 - ▶ **Auflösung nach Anwendung:** Wie bereits beschrieben, kann eine BSP-Anwendung aus mehreren Webseiten bestehen. Im Transaktionsprofil wird bei dieser Auflösung eine Performancestatistik pro auf BSP-Anwendung erstellt, die Seiten selbst werden nicht aufgelöst.
 - ▶ **Summierung unter Report SAPMHTTP:** Es erfolgt keine Auflösung nach BSP-Anwendung, d. h. alle Web-Requests werden im Transaktionsprofil unter dem Eintrag »SAPMHTTP« gesammelt.

Das Einschalten der Statistiken bringt nur geringe Performanceeinbußen und ist daher praktisch immer zu empfehlen. Nur bei Systemen unter sehr starker Hochlast sollten Sie testen, ob sich die Performance entscheidend verbessert, wenn Sie die Statistiken für BSP-Anwendungen wieder deaktivieren.

Spezielle Performanceanalyse

Wollen Sie die Performance einer bestimmten BSP im Detail analysieren, so verwenden Sie zunächst die Einzelsatzstatistik (Transaktionscode STAD). Da es sich bei BSPs im Wesentlichen um in ABAP programmierte Anwendungen handelt, stehen Ihnen als Werkzeuge für die Detailanalyse z. B. der Performance-Trace, die ABAP-Laufzeitanalyse und der Debugger für die Detailanalyse zur Verfügung. (Eine Einführung in die entsprechenden Werkzeuge finden Sie in Kapitel 4, »Identifizierung von Performanceproblemen in ABAP- und Java-Programmen«.)

Um in der Einzelsatzstatistik (Transaktionscode STAD) die Statistiken zu BSP-Anwendungen zu analysieren, wählen Sie im Selektionsbildschirm der Einzelsatzstatistik im Feld **Task type** den Wert »H« für HTTP-Anfragen bzw. »T« für HTTPS-Anfragen aus. Um die Einzelsatzstatistik verwenden zu können, muss das Schreiben der Statistik im SAP-Kernel aktiviert sein, wie bereits weiter oben in diesem Abschnitt beschrieben wurde.

Die ABAP-Laufzeitanalyse (Transaktionscode SE30) aktivieren Sie in der Servicepflege (Transaktionscode SICF). Wählen Sie dazu in dem Objektbaum auf der linken Seite des Bildschirms eine BSP aus. Anschließend wählen Sie im Menü die Funktion **Bearbeiten · Laufzeitanalyse · Aktivieren**. Es wird ein Dialogfenster angezeigt, in der Sie den Namen der BSP und den Zeitraum, in dem die Laufzeitanalyse aktiv sein soll, angeben. Markieren Sie das Ankreuzfeld **Nur Aufrufe von SAPGUI-IP**, und wählen Sie die Funktion **Aktivieren**.

Zur Auswertung wählen Sie im Menü die Funktion **System · Hilfsmittel · Laufzeitanalyse · Ausführen** oder geben Sie den Transaktionscode SE30 ein. Sie befinden sich in der Pflege Transaktion für die ABAP-Laufzeitanalyse, die im vierten Kapitel dieses Buches im Detail vorgestellt wird. Um die Laufzeitanalyse einer BSP anzuzeigen, wählen Sie die Schaltfläche **Andere Datei**. Es wird ein Dialogfenster angezeigt, in dem Sie die z. B. über Angabe des Benutzers, der die BSP ausgeführt hat, die entsprechende Messdatei selektieren können. Die Analyse erfolgt wie im vierten Kapitel beschrieben.

Es ist möglich, den ICM alle Anfragen protokollieren zu lassen. Diese Funktionalität aktivieren Sie über einen SAP-Profilparameter (siehe Anhang C). Das Aktivieren dieser Funktionalität kann zu Serialisierungseffekten führen.

ICM-Logging
aller Anfragen

7.5 SAP J2EE Engine

Mit der SAP J2EE Engine hat der SAP Web Application Server einen vollwertigen J2EE Application Server »im Bauch«, der es ermöglicht, Java-Anwendungen auf der Basis von Java Server Pages (JSP), Java Servlets, und Enterprise Java Beans zu entwickeln und zu betreiben.

Die SAP J2EE Engine kann entweder als allein laufende Komponente (z. B. mit mySAP Enterprise Portal 5.0) oder als Bestandteil des SAP Web AS (z. B. mit mySAP CRM 3.0) installiert und betrieben werden. Bei der Installation eines SAP Web AS wird die SAP J2EE Engine automatisch (aber optional) mit installiert.

7.5.1 Grundlagen der SAP J2EE Engine

Java Server Pages (JSP) und Java Servlets sind von einem Benutzer über das HTTP-Protokoll ausführbare Programme. Im Falle von JSPs bestehen die Anwendungen aus HTML-Seiten mit eingebettetem Java-Code; Java Servlets sind Java-Programme, die HTML-Seiten generieren. Java Servlets und JSPs bilden die Präsentations- und Steuerlogik einer Business-Anwendung ab. Enterprise Java Beans dagegen sind Java-Programme, in denen die Anwendungslogik einer Business-Anwendung programmiert ist.

Mit dem SAP Web Application Server ist es möglich, Web-Anwendungen so zu entwickeln, dass die Darstellungs- und Steuerlogik in der Java-Laufzeitumgebung und die Anwendungslogik in der ABAP-Laufzeitumgebung abgebildet ist. Dieses Vorgehen hat zahlreiche Vorteile, unter anderem den Schutz von bereits getätigten Investitionen in ABAP-Anwendungen oder den Rückgriff auf erfahrene Entwickler und Prozesse im ABAP-Bereich. Ein Beispiel für eine solche »Arbeitsteilung« sind viele Anwendungen im mySAP CRM. Allerdings ist es auch möglich, Anwendungen komplett in Java zu realisieren.

Dieser Abschnitt soll Ihnen einen ersten Überblick über die Möglichkeiten der Performanceanalyse geben. Da die SAP J2EE Engine noch ein junges Produkt ist, stehen hier naturgemäß bisher wenig Erfahrungen über Performanceprobleme in großen, produktiven Anwendungen zur Verfügung und auch die Methoden und Werkzeuge der Performanceanalyse sind noch nicht so fein entwickelt wie bei anderen SAP-Komponenten. Wir beschränken uns daher in diesem Abschnitt auf einen Architekturüberblick und die Einbindung der SAP J2EE Engine in die zentrale Monitoring-Infrastruktur des SAP Web AS.

Anmelden an den Applikations-server

Mit der folgenden URL melden Sie sich an eine Java-Anwendung »Hello« an, die als ein Beispiel mit der SAP J2EE Engine ausgeliefert wird:

```
http://<sapserver>:<port>/Hello
```

Allgemein erreichen Sie eine Java-Web-Anwendung über den URL `http://<server>:<port>/<alias>`. Dabei steht `<sapserver>` für den Namen eines Anwendungsservers, auf dem der Dispatcher des J2EE-Clusters läuft, `<port>` ist der TCP/IP-Port, auf dem der Dispatcher hört. Den Port finden Sie in der Konfigurationsdatei `cluster/dispatcher/cfg/services/http.properties`. Im Falle einer Installation der J2EE Engine ohne den SAP Web AS (Standalone-Installation) ist dies der Port 3011. Im Falle einer Installation innerhalb des SAP Web AS ist es der Port 5xx00, wobei xx die zweistellige

Systemnummer der SAP-Instanz ist. *<alias>* ist der frei wählbare Name der Anwendung.

ABAP-Laufzeitumgebung	J2EE-Laufzeitumgebung (SAP J2EE Engine)
SAP-System	J2EE-Cluster
SAP-Instanz	J2EE-Server
SAP-Workprozess	Work-Thread
Message-Server	J2EE-Dispatcher
Benutzerkontexte liegen im Shared Memory einer SAP-Instanz, d.h. Benutzer sind an eine Instanz »gebunden«.	Benutzerkontexte liegen im Local Memory eines J2EE-Servers, d.h. die Benutzer sind an einen Server »gebunden«.
SAP GUI, SAP ITS oder Webbrowser als Präsentationsserver	Webbrowser als Präsentationsserver
Benutzeranfragen werden vom Message Server beim Anmelden auf eine Instanz mit niedriger Last verteilt (wenn Logon-Balancing aktiviert ist). Anschließend kommunizieren Präsentationsserver und SAP-Instanz direkt miteinander.	Benutzeranfragen werden vom J2EE-Dispatcher auf einen Server verteilt. Wenn ein Benutzerkontext bereits besteht, dann wird ein Benutzer wieder auf den Server verteilt, der den Benutzerkontext besitzt. Präsentationsserver und J2EE-Server kommunizieren grundsätzlich nur über den Dispatcher miteinander.

Tabelle 7.4 Entsprechungen zwischen ABAP-Laufzeitumgebung und Java-Laufzeitumgebung

Die Java-Laufzeitumgebung unterscheidet sich von der klassischen ABAP-Laufzeitumgebung, allerdings gibt es auch Parallelen. Die wichtigsten Parallelen und Unterschiede sind in Tabelle 7.4 zusammengestellt. Parallelen und Unterschiede werden besonders klar, wenn wir im Folgenden beschreiben, wie ein Transaktionsschritt in der Java-Laufzeitumgebung abläuft. Dabei nehmen wir zunächst an, dass der J2EE-Cluster aus einem Dispatcher und mehreren Servern besteht (siehe auch Abbildung 7.4 in Abschnitt 7.2

Eine HTTP-Anfrage erreicht zunächst den Dispatcher des J2EE-Clusters. (Dies gilt für eine alleinlaufende Installation der J2EE Engine. Im Falle der Installation der J2EE Engine im SAP Web AS erreicht die Anfrage zunächst den Internet Communication Manager, der diese an die J2EE Engine weiterreicht.) Der Dispatcher des J2EE-Clusters prüft zunächst, ob ein Benutzerkontext bereits angelegt wurde. Ist dies nicht der Fall, so leitet er die Anfrage an einen Server mit niedriger Last weiter. Erreicht den Dispatcher

Generieren der HTML-Seite

eine Anfrage von einem Benutzer, für den bereits ein Kontext angelegt ist, so leitet der Dispatcher die Anfrage an den Server weiter, in dem der Benutzerkontext liegt. Ein Wechsel von Benutzerkontexten zwischen Servern erfolgt nicht.

Damit der Dispatcher seine Aufgabe wahrnehmen kann, übermitteln die Server dem Dispatcher ihre Lastdaten. Dispatcher und Server sind Prozesse auf Betriebssystemebene (dispatcher.exe und server.exe). Sie sind multi-thread-fähig, d. h., sie können mehrere Anfragen parallel bearbeiten. Sie können auf Betriebssystemebene mehr als einen Prozessor belegen.

Der Server legt einen Benutzerkontext an und beginnt die Anfrage zu bearbeiten. Der von den Servern allokierte Speicher ist lokaler Speicher, d. h., der Java-Server verfügt nicht über ein Shared Memory wie eine SAP-Instanz in der ABAP-Laufzeitumgebung.

Um die Anfrage zu bearbeiten, muss der Server in der Regel auch Daten beschaffen und bearbeiten. Dies kann entweder durch direkten Datenbankzugriff oder indirekt über die ABAP-Laufzeitumgebung geschehen. Der Zugriff auf die ABAP-Laufzeitumgebung erfolgt über den Java Connector (JCo). Technisch ist dies auf Seiten der Java-Laufzeitumgebung ein Enterprise Java Bean und auf Seiten der ABAP-Laufzeitumgebung ein BAPI. Alternativ kann der direkte Zugriff auf die Datenbank aus einer Java-Anwendung heraus auch über folgende Schnittstellen erfolgen: Native-JDBC (dynamisches SQL, DB-plattformabhängig, vorbei an den SAP-Erweiterungen wie Tabellenpufferung oder Java-Dictionary), Open SQL for Java (dynamisches SQL, DB-plattform-unabhängig, mit Tabellenpufferung), SQLJ (statisches SQL ähnlich wie im ABAP), EJB-CMP (Container-Managed Persistence). Details zu diesen Schnittstellen entnehmen Sie bitte der technischen Dokumentation zur SAP J2EE Engine.



Betrachten wir beispielhaft eine Anwendung, die Sie auf Grundlage von SAP R/3 Enterprise entwickeln wollen. Sie haben sich dafür entschieden, dass eine Web-Oberfläche für eine Logistik-Anwendung mittels der SAP J2EE Engine realisiert werden soll. Dabei soll auf Tabellen der SAP-Komponente Vertrieb (SD) zugegriffen werden. Die Frage, die Sie sich mit ihrem Entwicklungs-Team stellen, ist nun, ob es sinnvoll ist, direkt aus der Java-Web-Anwendung auf die SAP-Logistik-Tabellen zuzugreifen, oder dies über die Schnittstellen (BAPIs) der ABAP-Laufzeitumgebung zu tun. Verwenden Sie die BAPIs, so greifen Sie wiederum automatisch auf die SAP-Basis-Services wie SAP-Tabellenpufferung, Sperrverwaltung und

eindeutige Dokumentnummernvergabe zu. Ein direkter Zugriff auf die Datenbanktabellen aus der Java-Umgebung dagegen ohne Nutzung dieses Services könnte zu anwendungsseitigen Dateninkonsistenzen führen. Auch grundlegende logistische Funktionen, wie die Verfügbarkeitsprüfung oder die Preisfindung müssten gegebenenfalls in Java neu entwickelt werden, was ebenso die Gefahr der Inkonsistenz in sich birgt. In einem solchen Fall würde man sich also für eine »gemischte« Entwicklung, d. h. für die Benutzung von ABAP-BAPIs entscheiden.

Die Sperrverwaltung in Java ist allerdings seit SAP Web AS 6.30 verfügbar. Die Sperren werden dabei über den Enqueue-Server der ABAP-Laufzeitumgebung gesetzt. Seitdem besteht das Inkonsistenzproblem zumindest im Teil »Sperren« nicht mehr.

Kommen wir nach diesem Beispiel zurück zum Ablauf des Transaktions-schrittes in der J2EE Engine. Hat der Server die Anfrage bearbeitet, so gibt er das Ergebnis der Anfrage an den Dispatcher zurück, der das Ergebnis an den Benutzer sendet, der die Anfrage gestartet hat.

Die Java-Laufzeitumgebung des SAP Web Application Servers skaliert wie folgt:

Skalierbarkeit und Logon-Balancing

- ▶ Die einfachste Installation besteht aus einem Dispatcher und einem Server, die beide auf einem Rechner laufen. Da Dispatcher und Server multi-thread-fähig sind, können auf einer solchen Installation bereits mehrere Benutzer parallel arbeiten. Diese Art der Installation findet also als Entwicklungssystem oder als Produktivsystem im Prototypbetrieb Anwendung.
- ▶ In einer nächsten Stufe besteht ein J2EE-Cluster aus einem Dispatcher und mehreren Servern. Die Server können über mehrere Rechner verteilt sein. Mit einer solchen Installation lässt sich bereits eine große Last abdecken. Der Nachteil ist allerdings, dass nur ein Dispatcher zur Verfügung steht und dieser damit einen Single Point of Failure (SPOF) bildet. Fällt dieser aus, so ist die Anwendung nicht mehr verfügbar.
- ▶ Um einen J2EE-Cluster hoch verfügbar zu machen, sollte man also mehrere Dispatcher und mehrere Server über mehrere Rechner verteilen (jeweils mindestens zwei). Dabei können die Dispatcher auf alle Server zugreifen, d. h., es herrscht nicht, wie bei der ABAP-Laufzeitumgebung eine 1:n-Beziehung von Dispatcher zu Workprozessen, sondern ein m:n-Beziehung zwischen einer Menge von Dispatchern und einer Menge von Servern. Das Balancing zwischen den Dispatchern erfolgt über Web-Routing durch den SAP Web Dispatcher oder durch ein vergleichbares Werkzeug eines SAP-Partners.

Performance-limitierende Faktoren

Unter der SAP J2EE Engine liegt als Laufzeitumgebung die Java Virtual Machine (VM). Die Einstellungen der Java VM haben einen wesentlichen Einfluss auf die Performance der SAP J2EE Engine, insbesondere auf die Anzahl der Anfragen, die von dieser abgehandelt werden können. Dabei sind drei kritische Punkte zu beachten:

- ▶ **Konfiguration der Threads:** Sind alle Threads eines Servers belegt, so führt dies dazu, dass Benutzer warten müssen.
- ▶ **Konfiguration des virtuellen Speichers:** Reicht der virtuelle Speicher für die Java VM nicht aus, so kommt es zu Fehlern in der Speicherverwaltung.
- ▶ **Konfiguration des Garbage-Kollektors:** Der Garbage-Kollektor ist dafür verantwortlich, nicht mehr benötigte Speicherbereiche aufzuräumen. Kommt er dieser Aufgabe zu langsam nach, so kommt es ebenfalls zu Fehlern in der Speicherverwaltung. Auf der anderen Seite kann eine nicht optimale Einstellungen des Garbage-Kollektor auch dazu führen, dass dieser unnötig viel CPU beansprucht und somit zu einem Performanceproblem führt.

Die Belegung der Threads und des Speichers kann in den Monitoren, die im nächsten Abschnitt vorgestellt werden, überwacht werden.

Detaillierte Empfehlungen zu den Einstellungen finden Sie zum einen in den Tuning-Guides zur SAP J2EE Engine und zum anderen im SAP Going-Live Report, der individuell für Ihr System vor dem Produktivstart erstellt wird.

7.5.2 Konfigurations- und Überwachungswerkzeuge

Zum SAP J2EE Engine gehören Konfigurations- und Überwachungswerkzeuge. Im Folgenden sprechen wir diejenigen an, die für eine Performanceanalyse interessant sind.

Visual Administrator

Der *Visual Administrator* ist das Werkzeug, mit dem Sie die SAP J2EE Engine während des Betriebes überwachen. Die wesentlichen Funktionen sind die Konfiguration und Administration der Services (Programme, Benutzer, Verbindungen etc.), die Anzeige der Protokolldateien von Dispatchern und Servern sowie die Anzeige der Performancedaten. Einige der wichtigsten Funktionen besprechen wir im Folgenden.

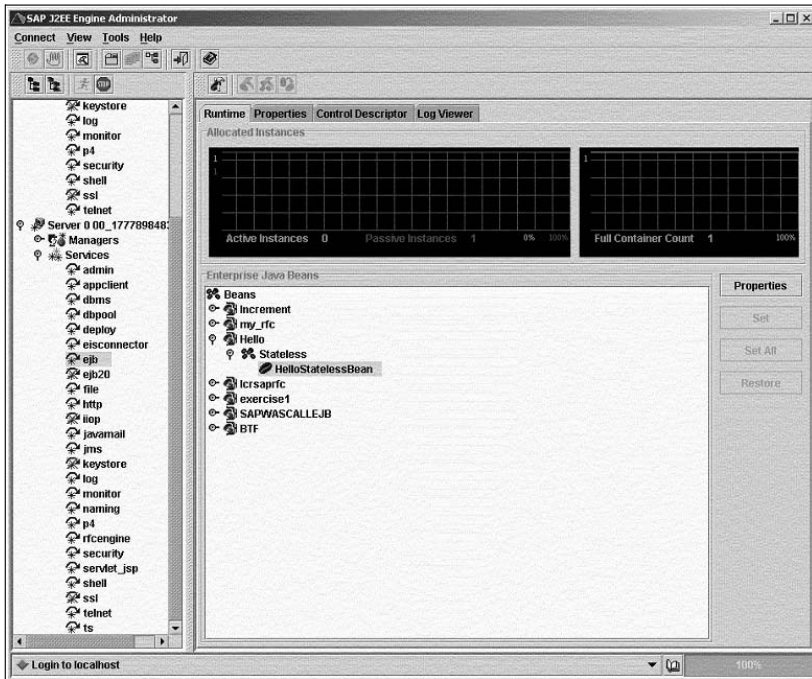


Abbildung 7.9 Der Visual Administrator (VA) ist das wichtigste Werkzeug zur Administration der SAP J2EE Engine. Im linken Bildschirmfenster finden Sie eine Übersicht über die Services des VA. In der Abbildung ist der ejb-Service ausgewählt worden. Im rechten Bildschirmbereich finden Sie detaillierte Informationen des gewählten Services.

Eine Übersicht über die aktiven Java-Anwendungen finden Sie im *deploy*-Service und im *http*-Service des Visual Administrators. Im Letzteren finden Sie auch die eingerichteten Aliase, mit denen Anwendungen aufgerufen werden können. Die eingerichteten Aliase starten dann Anwendungen über die URL `http://<server>:<port>/<alias>`. Im *http*-Service finden Sie auch die Angabe über den TCP/IP-Port, auf dem der Dispatcher hört.

Der *monitor*-Service im Visual Administrator bietet u. a. folgende Monitoringfunktionen:

- ▶ **Thread-Monitor:** Übersicht über den aktuellen Status der Java Worker Threads und deren Historie (z. B. die maximale Anzahl benutzter Threads)
- ▶ **Speichermonitor:** Aktuell allozierter virtueller Speicher (absolut und in Prozent des maximal möglichen Speichers)
- ▶ **Kontextmonitor (Session-Monitor):** Übersicht über Kontexte, bei denen Benutzer durch das Benutzermanagement autorisiert worden sind und über anonyme Kontexte

- ▶ **HTTP-Monitor:** Mittlere und maximale Anzahl von HTTP-Anfragen sowie Anzahl wartender Anfragen

Der *Cluster-Konfigurator (Config Tool)* ist ein Werkzeug, das Sie verwenden, wenn die SAP J2EE Engine gestoppt ist. Mit diesem Werkzeug können Sie ebenfalls die Protokolldateien anzeigen sowie Cluster-Elemente hinzufügen oder löschen.

Die Konfigurationsparameter der SAP J2EE Engine sind in den »Properties«-Dateien abgelegt. Änderungen sollten nur über die Administrationswerkzeuge erfolgen, da nur in diesem Fall Protokolleinträge über die vorgenommenen Änderungen durchgeführt werden.

Globaler CCMS-Überwachungsmonitor

Das Monitoring der die SAP J2EE Engine kann – wie bei anderen SAP-Komponenten auch – im zentralen CCMS-Alert-Monitor (Transaktionscode RZ20) geschehen. In SAP-Hinweis 498179 finden Sie die Details, um den Monitor für die SAP J2EE Engine im CCMS zu aktivieren. Tabelle 7.5 gibt eine Übersicht über die wichtigsten Kennzahlen.

Pfad im zentralen CCMS-Alert-Monitor	Bedeutung
Server X • Workload • Threads • System/ ClientThreads	Kennzahlen zu Threads: Initiale und aktuelle Anzahl, belegte und freie Threads, Anzahl der Anfragen in der Queue
Server X • Workload • Memory	Kennzahlen zum Speicherverbrauch: Allokierter und aktuell benutzter Speicher, Speicherlimit
Server X • Workload • Connections Manipulator	Kennzahlen zu den aktuell vorhandenen Verbindungen
Server X • Workload • Requests • Http requests	Kennzahlen zu Anzahl von Anfragen und Antwortzeiten: Mittlere, maximale und wartende Anzahl von Anfragen, mittlere und maximale Antwortzeit pro Anfrage
Server X • Workload • Logs	Informationen aus den Logs der J2EE Engine
Server X • Workload • Sessions • Login Sessions	Kennzahlen zu Sessions: Minimale, aktuelle und maximale Anzahl von Sessions
Server X • Services • Application Tracing • JARM	Performancedaten zu Java-Anwendungen (JARM), siehe auch Abschnitt 4.6

Tabelle 7.5 Performancedaten des SAP J2EE Engine im zentralen CCMS-Alert-Monitor. Die angegebenen Kennzahlen werden für jeden Dispatcher, für jeden Server und in Summe für den Cluster ausgegeben.

Pfad im zentralen CCMS-Alert-Monitor	Bedeutung
Configuration • Heart Beat	Übersicht und »Lebenszeichen« von gestarteten Services
Configuration • PortsManager	Anzeige der Portnummern für <i>http</i> -, <i>telnet</i> - und <i>p4</i> -Service
Configuration • System Info	Anzeige wichtiger Systemparameter: ID, Rechner und Port des Cluster-Elements, Version der SAP J2EE Engine und der Java-Edition, Name und Version von Betriebssystem und Java VM, Working Directory, Startzeit des Servers

Tabelle 7.5 Performancedaten des SAP J2EE Engine im zentralen CCMS-Alert-Monitor. Die angegebenen Kennzahlen werden für jeden Dispatcher, für jeden Server und in Summe für den Cluster ausgegeben. (Forts.)

Die wichtigsten SAP-Hinweise zur Konfiguration der Java VM und der SAP J2EE Engine finden Sie in Anhang H. Weitere Informationen zur Konfiguration der Java VM finden Sie auch auf den Internetseiten von SUN, die in Anhang G zusammengestellt sind.

7.6 End-to-End-Performanceüberwachung

Die bisher vorgestellten Analysemethoden fokussieren auf die serverseitige Überwachung, d. h., sie analysieren die Integrität der technischen Komponenten (ICM, ITS, J2EE Engine). Es kommt jedoch vor, dass alle Komponenten technisch einwandfrei laufen, die Benutzer sich aber dennoch über schlechte Performance oder Fehler beschweren. In diesem Fall kommen die in diesem Abschnitt beschriebenen Methoden zum Zuge. Sie sind im Wesentlichen unabhängig von dem verwendeten Server, d. h., sie können sowohl für einen SAP Web AS, einen ITS, einen J2EE Engine oder auch für jede andere Servertechnologie verwendet werden.

7.6.1 Laufzeitanalyse im Webbrowser durchführen

Klagen Benutzer über eine schlechte Performance Ihrer Web-Anwendung und können Sie im SAP Web AS keine akuten Engpässe identifizieren, so kommen Probleme auf dem Präsentationsserver oder im Netzwerk als weitere Ursachen in Frage. Um diesem Verdacht nachzugehen, können Sie auf dem Präsentationsserver einen Trace mit einem Analysewerkzeug des jeweiligen Betriebssystems durchführen.

Die folgende Analyse ist nicht auf mit dem SAP Web AS erzeugte Anwendungen beschränkt; Sie können beispielsweise auch die Web-Anwendun-

gen Ihrer Online-Bank mit dieser Methode analysieren, wenn sie Ihnen zu langsam erscheinen. Im Folgenden beschreiben wir die Analyse auf Windows NT mit dem Programm PERFMON; für Windows 95 existiert das Programm SYSMON mit nahezu der gleichen Funktionalität.

1. Starten Sie auf Ihrem Windows-NT-Präsentationsserver PERFMON, z. B. über **Start Run • PERFMON**. (Wenn PERFMON nicht existiert, können Sie dieses über **Start Settings • Control Panel • Add/Remove Programs** nachinstallieren.)
2. Richten Sie PERFMON ein:
 - ▶ Zum einen müssen Sie die CPU-Auslastung auf Ihrem Präsentationsserver messen. Geben Sie dazu im Menü von PERFMON ein: **Edit • Add to Chart**. Wählen Sie aus **Object** den Wert »Processor« aus und für **Counter** den Wert »% Processor Time«.
 - ▶ Zum Zweiten benötigen Sie zur Analyse die übertragene Datenmenge: Wählen Sie dazu für **Object** den Wert »Network Interface« und für **Counter** den Wert »Bytes Received/sec« aus. Hinweis: Je nach Art der Netzwerkanbindung kann die Bezeichnung von **Category** und **Item** von den oben angegebenen Beispielen abweichen. Es kann sein, dass Sie auf NT-Ebene Prozesse starten müssen, die den Netzwerkverkehr überwachen.
3. Beginnen Sie Ihre Analyse: Rufen Sie dazu die Web-Anwendung auf, die Sie testen wollen. Im PERFMON-Bildschirm beobachten Sie, wie kurz nach dem Start der Anwendung Netzwerk- und CPU-Auslastung am Präsentationsserver ansteigen.

Die Auswertung der Analyse ergibt Informationen über:

- ▶ die Browser-Generierungszeit (auch *Rendering-Zeit* genannt): Dies ist die Zeit, zu der Ihr Browser mit der Generierung der HTML-Seite beschäftigt ist. In Abbildung 7.10 erkennen Sie, wie im oberen Bildschirm die CPU-Auslastung nahe 100 % ist.
- ▶ die Netzwerkübertragungszeit: Dies ist die Zeit, in der die Daten an den Browser übertragen werden. In Abbildung 7.10 erkennen Sie im unteren Bildschirm ein Plateau in der Netzwerkübertragungsrate bei 8 kByte/sec. Dies ist die Bandbreite der ISDN-Leitung, mit der in diesem Beispiel der Rechner angebunden ist. Über die Funktion **File • Export Chart** können Sie die Messdaten sichern und in ein Tabellenkalkulationsprogramm übertragen. Die übertragene Datenmenge ermitteln Sie, indem Sie die Werte der Spalte **Bytes Received/sec** summieren.

- ▶ die »Restzeit«: Messen Sie weder CPU-Aktivität an Ihrem Präsentationsserver noch Netzwerkaktivität, obwohl Sie im Browser die Sanduhr sehen, so können Sie daraus schließen, dass die Zeit im Server (in unserem Falle im ITS, der SAP-Applikations- oder Datenbankebene) zu suchen ist.

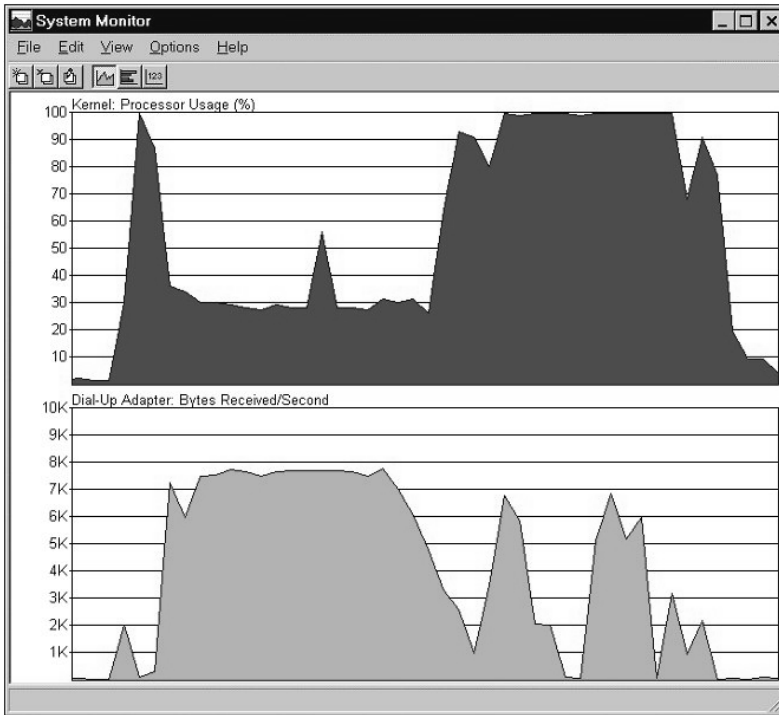


Abbildung 7.10 Analyse einer Web-Anwendung mit SYSMON auf einem Präsentationsserver mit Windows 95 als Betriebssystem und einer ISDN-Netzwerkanbindung mit einer Bandbreite von 8 kByte/sec

Anhand dieser einfachen Analyse können Sie wie folgt das Optimierungspotenzial abschätzen:

- ▶ Die Browser-Generierungszeit ist hoch: In diesem Fall ist zu überlegen, ob eventuell ein wenig leistungsstarker PC verwendet wird. Sie können sofort erkennen, ob eine Investition in neue Hardware zusätzliche Performance bringen würde.
- ▶ Die Netzwerkübertragungszeit ist hoch: Auch hier können Sie sofort berechnen, inwieweit eine schnellere Netzwerkverbindung sich in einer beschleunigten Antwort niederschlägt.

- ▶ In beiden Fällen lässt sich natürlich ebenso Performance gewinnen, indem die Anwendung optimiert wird, d. h. weniger Daten übertragen werden. Die wichtigsten Stichpunkte hierzu sind:
 - ▶ Reduzieren von Bildern
 - ▶ Vereinfachen der Bildschirminhalte bei SAP-GUI-for-HTML-Anwendungen durch Transaktionsvarianten (bereits beschrieben in Abschnitt 7.1.2, »Performance der GUI-Kommunikation analysieren und optimieren«)
 - ▶ Analyse des HTML-Textes im Hinblick auf überflüssiges Coding, z. B. unnötig übertragenes Coding in Includes
- ▶ Die »Restzeit« ist hoch: In diesem Fall sollten Sie eine Performanceanalyse auf ITS und SAP-Applikationsebene vornehmen. Insbesondere sollten Sie die Statistiksätze im SAP-System auswerten, die Sie über die Transaktionscodes STAD (ab SAP-Basis 4.6) bzw. STAT (bis SAP-Basis 4.5) erreichen (siehe auch Kapitel 4). Spezifizieren Sie den Benutzernamen und den Zeitraum der Messung, um die statistischen Sätze anzuzeigen. Aus den statistischen Sätzen entnehmen Sie die Antwortzeit im SAP-System, die CPU-Zeit im SAP-System, die Datenbankzeit und weitere Details.

7.6.2 Web-Anwendungen mit dem CCMS kontinuierlich überwachen

Mit dem Alert-Monitor im CCMS ist es ebenfalls möglich, Abfolgen von Webseiten durch die periodische Abfrage von Test-Requests zu überwachen. In einem Web-Shop könnten Sie mit diesem Werkzeug im Abstand von zehn Minuten testen, ob der Katalogzugriff auf ausgewählte Produkte, der Produktkonfigurator, die Preisfindung oder die Verfügbarkeitsprüfung noch möglich ist. Dazu bietet das CCMS eine Umgebung an, den so genannten *Generic Request and Message Generator* (GRMG), in der Sie eigene Testanwendungen definieren und testen lassen können.

In einem typischen Web-Szenario, z. B. dem Internet Sales im mySAP CRM, läuft ein solcher Test wie folgt ab: Die CCMS Monitoring Infrastructure sendet in periodischen, definierten Abständen eine Testanfrage an die GRMG-Testanwendung. Diese Anwendung ist ein JSP Servlet oder eine SAP ABAP Object Class. Diese wird von der SAP ausgeliefert oder kann auch individuell für die Kundenanwendung geschrieben werden. Diese Testanwendung prüft nun die Verfügbarkeit der Komponenten, die für das Funktionieren der Internetanwendung notwendig ist. Das Ergebnis des Tests wird an die CCMS Monitoring Infrastructure übergeben. Von dort können die entsprechenden Alerts dann z. B. an den grafischen Alert-Monitor im SAP Solution

Manager oder über SAPConnect an einen E-Mail- oder Telefonserver weitergeleitet werden.

Das Monitoring von Testanwendungen ergänzt in idealer Weise das oben beschriebene »interne Monitoring« des ITS oder der J2EE Engine. Es prüft nicht nur die technische Verfügbarkeit der Anwendung, sondern kann auch über das Ergebnis einer Anwendung eine Aussage machen (d. h. z. B.: Wird ein »sinnvoller« Preis berechnet?). Bei einem Alarm, d. h. einem Ausfall einer Web-Anwendung, liefert es allerdings keine weitere Einsicht zur Analyse der Ursache des Fehlers. Es ist auch nicht proaktiv, d. h., es meldet nur reaktiv Fehler, während z. B. die Überwachung der Workprozesse im ITS bereits einen Alarm auslösen könnte, wenn eine bestimmte Auslastung überschritten wird, der Endbenutzer aber noch keine Probleme erkennt.

Eine technische Dokumentation finden Sie im SAP Service Marketplace unter <http://service.sap.com/systemmanagement> • *System Monitoring*

7.6.3 Werkzeuge von Drittanbietern

Zahlreiche Hersteller bieten Werkzeuge zur Überwachung von URLs an. Die folgende Internetseite fasst solche Monitoring-Produkte zusammen: http://dmoz.org/Computers/Software/Internet/Site_Management/Monitoring/.

Mit Hilfe dieser Werkzeuge können Sie zum einen die Verfügbarkeit »strategisch wichtiger« Webseiten zentral überwachen, egal von welchem Server diese geliefert werden. Die Werkzeuge bieten dabei auch einen Content-Check an, d. h., es kann wirklich geprüft werden, ob auch ein korrekter Inhalt angezeigt wird. Sie können weiterhin URL-Transaktionen, d. h. Folgen von HTML-Seiten, definieren, die periodisch abgespielt werden sollen. (Dabei müssen Sie natürlich darauf achten, dass keine echten Dokumente erzeugt werden.)

7.7 Zusammenfassung

Es gibt grundsätzlich zwei »todsichere« Methoden, ein Performanceproblem im Bereich der Frontend-Kommunikation zu erzeugen. Die erste ist, große Datenmengen an den Frontend-Client zu übertragen. Die zweite Methode ist, eine große Anzahl von Roundtrips (Kommunikationsschritten zwischen Frontend-Client und Server) zu programmieren.

Anwendungen auf Basis des SAP GUI reagieren bei einer nicht-optimalen Programmierung oft noch halbwegs fehlertolerant, da das Protokoll zwischen SAP GUI und SAP Application Server das SAP-eigene DIAG-Proto-

koll ist, welches auf eine minimale Datenübertragung getrimmt ist. Bei Web-Anwendungen, die HTTP verwenden, werden dagegen in der Regel selbst bei optimaler Programmierung schon deutlich mehr Daten als bei SAP GUI-Anwendungen übertragen. Kommt hier noch eine nicht-optimale Programmierung hinzu, so ist ein Performanceproblem unvermeidbar.

Eine nicht-optimale Programmierung kann dazu führen, dass eine Anwendung zwar im LAN (Local Area Network), d. h. z. B. am Arbeitsplatz des Entwicklers, noch zufrieden stellend arbeitet, im WAN (Wide Area Network), d. h. z. B. am Arbeitsplatz der Vertriebsmitarbeiterin in einer Außenstelle, aber zu katastrophalen Performanceproblemen führt.

In diesem Abschnitt wurden die Performanceaspekte der vier wesentlichen Techniken der SAP-Frontendprogrammierung beschrieben.

Bei SAP GUI-Anwendungen vermitteln ab SAP-Basis 4.6 RFCs auch die Kommunikation zwischen SAP-Applikationsebene und SAP-Präsentationsebene zum Aufbau von als Controls bezeichneten Bildschirmelementen (die im Rahmen der EnjoySAP-Initiative vermehrt Verwendung finden). Für die Analyse möglicher Performanceprobleme (z. B. im Netzwerk zu den Präsentationsservern) stehen die Einzelsatzstatistik, der Performance-Trace (RFC-Trace) sowie der Netzwerkcheck im Betriebssystemmonitor zur Verfügung.

Eine Reihe von Punkten sind für die Performance der Web-Anbindung mit dem SAP Internet Transaction Server von Bedeutung. Zunächst ist die Auswahl des richtigen GUIs zu nennen: Für manche Benutzergruppen ist der SAP GUI for HTML nicht die goldene Wahl. Es kann sinnvoll sein, auch weiterhin den SAP GUI for Windows (oder Java) zu verwenden. Die Konfiguration des ITS betrifft das redundante Aufsetzen von WGate und AGate, die Konfiguration der ITS-Workprozesse (Threads), Sessions und Caches sowie das Einrichten von Logon-Gruppen zur Anmeldung an das SAP-System. Schließlich sollten Sie mit der Performanceüberwachung und -analyse des ITS vertraut sein. Eine ständige Überwachung leistet der zentrale CCMS-Überwachungsmonitor, im Falle eines Performanceproblems helfen Analysen mit dem ITS-eigenen Administrationswerkzeug weiter.

Beim Betrieb von Web-Anwendungen auf Basis von Business Server Pages werden die Webseiten von den »klassischen« SAP-Workprozessen generiert. Zur Performanceanalyse wurden die bekannten Werkzeuge wie der Workload-Monitor oder der Performance-Trace entsprechend erweitert.

Der Internet Communication Manager fungiert als Kommunikationsbrücke zwischen den SAP-Workprozessen und dem Webbrowser des Benutzers. Er kann über Transaktionen im SAP CCMS konfiguriert und überwacht werden.

Mit der SAP J2EE Engine können Web-Anwendungen mittels Java Server Pages und Java Servlets sowie Enterprise Java Beans entwickelt und betrieben werden. Die SAP J2EE Engine ist in das SAP-Monitoring-Konzept, d. h. in den zentralen CCMS-Überwachungsmonitor, eingebunden.

Die abschließend beschriebene Methode zur Performanceanalyse von HTML-Seiten mit Betriebssystemwerkzeugen (z. B. PERFMON) ist generisch und kann nicht nur bei HTML-Seiten verwendet werden, die durch den SAP Web AS generiert werden, sondern auch für solche, die von anderen Servern generiert werden.

Wichtige Begriffe

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

- ▶ Roll-Wartezeit, RFC-Zeit, GUI-Zeit
- ▶ Controls und Frontend-Kommunikation
- ▶ Auswahl des »richtigen« GUIs: SAP GUI for Windows, SAP GUI for HTML, SAP GUI for Java Environment
- ▶ SAP Internet Transaction Server: WGate, AGate, Thread- und Session-Konzept, Caching im ITS
- ▶ Business Server Pages und Internet Communication Manager
- ▶ Java Server Pages und Java Servlets sowie Enterprise Java Beans
- ▶ Performanceanalyse mit PERFMON

Fragen

1. Was ist eine hohe Roll-Wartezeit?
 - a) Eine hohe Roll-Wartezeit ist ein eindeutiges Indiz für ein Problem mit der GUI-Kommunikation (z. B. im Netzwerk zwischen Präsentations- und Applikationsservern).
 - b) Eine hohe Roll-Wartezeit ist ein eindeutiges Indiz für ein Problem mit der RFC-Kommunikation mit externen SAP- oder Nicht-SAP-Systemen.
 - c) Eine hohe Roll-Wartezeit ist ein eindeutiges Indiz für ein Problem mit der GUI-Kommunikation oder der RFC-Kommunikation.

- d) Eine hohe Roll-Wartezeit kann auch durch ein inperformantes Netzwerk zwischen Applikations- und Datenbankebene hervorgerufen werden.
2. In einem Transaktionsschritt wird eine Transaktion prozessiert, die Controls verwendet, es wird aber kein externer RFC gerufen. Welche Aussagen treffen zu?
- Die GUI-Zeit ist größer als die Roll-Wartezeit.
 - Die RFC-Zeit ist größer als die Roll-Wartezeit.
 - Die Roll-Wartezeit ist immer größer als null.
 - Die Roll-Wartezeit ist in der Regel größer als null, kann aber auch null sein.
 - Die Roll-Wartezeit ist immer gleich null.
3. In einem Transaktionsschritt wird ein Programm prozessiert, das keine Controls und keine synchronen RFCs verwendet, es werden aber asynchrone RFCs gerufen. Welche Aussagen treffen zu?
- Die GUI-Zeit ist größer als die Roll-Wartezeit.
 - Die RFC-Zeit ist größer als die Roll-Wartezeit.
 - Die Roll-Wartezeit ist immer größer als null.
 - Die Roll-Wartezeit ist in der Regel größer als null, kann aber auch null sein.
 - Die Roll-Wartezeit ist immer gleich null.
4. Eine Web-Applikation, die unter Verwendung des ITS und eines SAP-Systems realisiert wurde, läuft »zu langsam«. Welche Analysen nehmen Sie vor?
- Im ITS-Administrationswerkzeug oder im zentralen CCMS-Überwachungsmonitor prüfen Sie, ob alle Workprozesse (Treads) oder Sessions auf dem ITS belegt sind, oder ob die CPU ständig belegt ist.
 - In der Workprozessübersicht des angeschlossenen SAP-Systems prüfen Sie, ob alle Workprozesse belegt sind.
 - Mit einem Performance-Trace und in der Einzelsatzstatistik auf dem angeschlossenen SAP-System analysieren Sie die Antwortzeit auf dem SAP-System und vergleichen diese mit der vom Benutzer gemessenen Antwortzeit auf dem Präsentationsserver.
 - Mit einem Analysewerkzeug auf dem Präsentationsserver (z. B. PERFMON) überprüfen Sie die zum Browser übertragene Datenmenge und die Compilierungszeit für die HTML-Seite im Browser und vergleichen die dafür benötigten Zeiten mit der Gesamtantwortzeit.

Index

32-Bit-Architektur 320
32-Bit-Technologie 302
64-Bit-Architektur 320, 335
64-Bit-Technologie 303

A

A004 358
ABAP 523
abap/heap_area_dia 306, 310, 331, 333
abap/heap_area_nondia 306, 310, 331, 333
abap/heap_area_total 306, 318, 331
abap/heaplimit 150, 306
ABAP-Coding 239
ABAP-Debugger 189, 190, 198
ABAP-Dictionary 523
ABAP-Laufzeitanalyse 182, 183, 198, 423
 BSP-Anwendungen 186
ABAP-Programm 165, 423, 428
ABAP-Programmabbruch 328
ABAP-Trace siehe ABAP-Laufzeitanalyse 182
ABAP-Trace Summary 188
AcceleratedSAP 523
ACID-Prinzipien 523
Address space 302, 319, 523
Administrations-Werkzeuge 445
Adressierbarer Speicher 276
Adressraum siehe Address space
Advanced Business Application Programming 523
Ändern von Sekundärindizes 416
AGate 265, 271
Agent Privat Memory 88
Aggregatfunktionen 424
Aktive Benutzer 141
Aktivität 132, 141, 151
ALE 523
Alertmonitor 523
Allgemeines Performanceproblem 142
Allokierter Speicher 111, 112, 479
American National Standards Institute 523
Analyse der SAP-Basis 75
Analyse eines Hardwareengpasses 77
Anlegen von Sekundärindizes 416
ANSI 523
Antwortzeit 138, 207

Anwendungsmonitor 154
Anwendungsserver siehe Applikations-server
Application Link Enabling 523
Application Support Layer 89
Applikationsebene 33
Applikationsfehler 330
Applikationsoptimierung 18
Applikationsserver 75, 267, 523
Applikationstuning 65
Architektur von mySAP-Anwendungen 29
ArchiveLink 523
Archiver-Stuck 103, 145
Archivierungsobjekt 523
Array-Fetch 176
ASAP 523
Asynchrone Verbuchung 213
Asynchroner RFC 238, 251
ATAB 367, 445
ATP-Server 204, 205, 386, 387, 389, 392, 538
ATP-Service 205
Ausführungsplan 404, 483, 523
Ausgehende Last 246
Auslagerungsspeicher 79, 302, 315, 316, 335, 523
Automatische Alert-Benachrichtigung 60

B

Backup 47
BAPI 524
Batch Input 524
Beendete Prozesse 120
Benchmark 225, 524
Benutzeranbindung 32
Benutzerkontext 136, 303, 304, 524
Benutzerprofil 154
Beschränkungen des Betriebssystems 315
Betriebsart 212, 524
Betriebssystem (Operating System) 301, 335, 528
Betriebssystemgrenze 331
Betriebssystemmonitor 76, 77
Bewegungsdaten 349
Bitmap-Index 444

Blöcke (Blocks) 87
Browser 524
BSP-Anwendungen, Laufzeitanalyse
186
Buffer-Trace 171
Business Application Programming In-
terface 524
Business Connector 524
Business Server Pages (BSP) 276

C

Cache-Mechanismen 272
Caches 36
CALL 304
Catalog Cache siehe Katalogpuffer
CATT 524
CBO 407, 409
CCMS 26, 30, 415, 524
CCMS System Component Repository
(SCR) 157
CCMS-Überwachungsmonitor 246
Change and Transport Organizing 524
Changes 352, 361
Changes und Commits 145
Checklisten zur Performanceanalyse
449, 455
Checkpoint 91
Client/Server-Architektur 33
Client-Destination-Statistiksätze 249
Client-Statistiksätze 249
CO 524
Common Programming Interface Com-
munication 524
Component Overview 195
Computer Aided Test Tool 524
Computing Center Management Sys-
tem (CCMS) siehe CCMS
COM-Routine 137
Control Panel 524
Control-Technik 256
Cost-Based Optimizer (CBO) siehe
CBO
CPI-C 524
CPU-Belastung 81, 147
CPU-Engpass 79
CPU-Ressourcen 206
CPU-Wartezeit 208
CPU-Zeit 138, 140, 143, 149, 207
CTO 524
Cursor 176
Customizing 524

Customizing Organizer 524
Customizing-Daten 349

D

D010S 445
Data Buffer 87
Data Cache 87
Data Control Language (Datenkontroll-
sprache) 525
Data Definition Language (Datendefini-
tionssprache) 525
Data Manipulation Language (Daten-
manipulationsprache) 525
Database Global Memory 88
Database Locks 372
Datenarchivierung 524
Datenbank 75, 525
Datenbankanalyse 75, 86
Datenbankebene 35
Datenbank-Fehlerprotokolldatei 101
Datenbank-Heap (Database Heap) 89
Datenbankindex 338
Datenbankinstanz 75, 326, 525
Datenbankkonsolidierung 231
Datenbanklast 427
Datenbankmonitore 465
Datenbankoperation 175
Datenbankoptimierer 101, 404, 406,
422, 483, 525, 528
Datenbankperformancemonitor 86
Datenbankperformanceproblem 148
Datenbankprozessmonitor 93, 465
Datenbankprozessor 91
Datenbankpuffer 86, 472
Datenbankserver 75, 525
Datenbanksperre 99, 372, 375, 471, 525
Datenbanksystem 371
Datenbank-View 437
Datenbankzeit 137, 143, 148, 149
Datenpuffer 87, 89, 476, 478, 480, 482
Datenpuffer (Buffer Pool) 89
Datenpuffer (Data Cache) 473, 481
Dauerhaftes Performanceproblem 144
DB2 UDB for iSeries 473, 490
DB2 UDB for Unix and Windows 88,
488
DB2 UDB for zSeries 492
DB2 Universal Database 473
DB2 Universal Database für OS/390
476
DB2 Universal Database für z/OS 476

DBA 525
 DB-Analyser 473
 DBIF_RSQL_NO_MEMORY 334
 DB-Procedure-Calls 137
 DB-Procedure-Subrecord 137
 DB-Procedure-Zeit 137
 dbs/io_buf_size 176
 DBSTATC 414
 DCL 525
 DDL 525
 DDLOG 344, 364
 DDNTF 360, 445
 DDNTT 360, 445
 Deadlock 377, 525
 Deaktivierte Verbuchung 119
 Denkzeit 207
 Detaillierte Tabellenanalyse 362
 Development Workbench 30
 DIAG-Protokoll 525
 Dialoglast 146
 Dialog-Service 204, 205
 Dialog-Workprozess 525
 Direct Read 175
 Direktes Lesen 484, 486, 487, 490, 492,
 494, 495
 Dispatcher 204, 525
 Dispatcher-Queue 123, 135
 Dispatcher-Wartezeit 135, 143, 149, 207
 Distributed Statistics Records, DSR 158
 DML 525
 dpmon 449
 Durchsatz 132, 141
 Dynamische Benutzerverteilung 409
 Dynamischer Statement Cache 477
 Dynpro 525

E

EarlyWatch Services 23
 Easy Web Transaction 266, 526
 EDI 525
 Eingehende Last 246
 Einzelsatzpuffer 337
 Einzelsatzpufferung 339
 Einzelsatzstatistik 165, 258
 Einzeltransaktionsanalyse 187
 Electronic Data Interchange (Elektronischer Datenaustausch) 525
 em/address_space_MB 311, 321
 em/blocksize_KB 306
 em/initial_size_MB 110, 119, 306, 320,
 326, 331

em/max_size_MB 311
 Engpassanalyse 129, 275
 EnjoySAP 255
 Enqueues 525, 538
 Enqueue-Service 203, 205
 Enqueue-Tabelle 378
 Enqueue-Trace 171, 181
 Entität 526
 Entwicklerprotokoll 332
 Erweiterter Speicherbereich (EM) 330
 Eskalationsverfahren 50
 EWT 526
 Exclusive Lockwait 99, 106, 127, 375, 471
 Execution Plan siehe Ausführungsplan
 Exklusive Datenbanksperre 100
 Exklusive Lockwait 124
 Expertenmonitore zur Performanceanalyse 43
 Explain Plan siehe Ausführungsplan
 Explain siehe Ausführungsplan
 Export/Import-Puffer 388
 EXSORT_NOT_ENOUGH_MEMORY 334
 Extended Memory 331, 526
 Extensible Markup Language 532
 Externe Modi 304
 Externes System 242

F

Failover-Lösung 205
 Falsche Lastverteilung 80, 120, 148
 FDDI 526
 Fehlende Datenbankindizes 102
 Fest allozierter Speicher (HEAP) 330
 Festplattenplatz 276
 Festplattenzugriff 470
 Fetch 352
 Fetch-Operation 175, 432
 Fiber Distributed Data Interchange 526
 File System Cache 82
 Finanzdienstleistung 229
 Firewall 271, 526
 Flussdiagramme zur Performanceanalyse 449
 FOR ALL ENTRIES 432, 435
 FOR ALL ENTRIES-Klausel 434
 Fragmentierung
 von Indizes 444
 Full Table Scan 405, 417, 484
 Funktionaler Trace 169
 funktionaler Trace 158

G

Generierungszeit 136
Generische Pufferung 340
Generische Region 341
Generischer Tabellenpuffer (TABL) 337
Gepufferte Tabelle 354, 360
Global Cache Hit Ratio 477
Globale Workprozessübersicht 121
Globaler Betriebssystemmonitor 275
Globaler CCMS-Überwachungs-
monitor 274
Globaler Speicher 302
Globales Extended Memory 312
GoingLive Check 23
Grafische Benutzeroberfläche 526
GUI 526
GUI-Kommunikation 258
GUI-Zeit 257, 259

H

Haltende Prozesse 119
Hardwareanalyse 75, 76
Hardwareengpass 80, 276
Hardwarekapazität 148
Hardwarekapazitätsplanung 227
Hardwarekonsolidierung 232
Hardwarelandschaft 204
Hardwarepartner 223
Hardwaresizing 220, 390
Hauptspeicherauslastung 79
Hauptspeicherbedarf 324
Hauptspeicherengpass 79, 80
Hauptspeicherprofil 150
Hauptspeicherpufferung 381
Heap Memory 526
High availability 205
Hintergrundlast 146
Hintergrundservice 204, 205
Hintergrundverarbeitung 526
Hints 409
Hiperpool Effizienz 477
Hochverfügbarkeit 526
Hot Package 526
Hot Spots 470
HTML 526
HTTP 526
Hypertext Markup Language siehe
HTML
Hypertext Transport Protocol siehe
HTTP

I

I/O-Engpass 83, 98
IAC 527
IDES 527
IDoc 32, 527
IDoc-Typ 527
IMG 527
Implementation Guide 527
Index Range Scan 405
Index Unique Scan 404
Indizes administrieren 411
INFORMIX 478, 494
Initiales Sizing 221, 224
Instanz 527
Inter Process Communication 528
Interaktionsmodell 256
Internal Document 527
International Demo- and Education
Systemsiehe IDES
Interne Modi 304
Internet 229
Internet Application Component 266,
527
Internet Communication Manager
(ICM) 276
Internet Transaction Server 210, 265,
270, 507, 528, 539
Internetanbindung 255
Internetportal 229
Intranet 528
Introscope 199
Invalidierung 108, 346
IPC 528
iSeries 311
ITS siehe Internet Transaction Server
ITS-Administrationswerkzeuge 273
ITS-Grundlagen 265
ITS-Instanz 271
ITS-Workprozess 272, 275
iViews 33, 193

J

J2EE Engine 539
JARM 193, 292
JARM-Statistiken 194, 199
JARM-Trace 197
Java 528
Java Application Request Measurement
(JARM) 193
Java-Programm 165, 193

K

KAPOL 354, 367, 445
Karenz 344
Katalogpuffer (Catalog Cache) 89
Konditionstabellen 350, 367
Kontextwechsel 528
Kontinuierliche Performanceoptimierung 62
Kopplung
 harte 238
 weiche 238

L

Ladezeit 136
LAN 528
LAN-Check 84
Lange Datenbankzeit 118
Last 141, 151
Last pro SAP-Modul 155
Lastverteilung 147, 201, 537
Laufzeitanalyse 293
Lesezugriff 387
Local Area Network (LAN) 37, 528
Local Cache Hit Ratio 477
Local Memory 302
Lock List 89
Lock-Eskalation 472
Logical Unit of Work 528
Logische Analyse 338
Logische Lesezugriffe 87
Logon-Gruppe 209, 271
Logon-Gruppe siehe auch Dynamische Benutzerverteilung
Logon-Gruppe siehe auch SMLG
Lokale Verbuchung 218
Lokaler Speicher 302, 528
Low Speed Connection 262
LUW 528

M

Mandant 528
MaxDB 465, 466, 468, 472, 473, 487
MaxDB Command-Monitor 469
MaxDB Resource-Monitor 470
memlimits 323
Memory Management 311
Message-Service 203, 205
MiniApps 33
Missing Index 102, 412
Mittlere Antwortzeit 152

Mobile Client 33
Modus 304, 528
Modusliste (Mode List) 327
Monitor zur Puffersynchronisation 364
Monitoring-Agent 169
Monitoring-Agenten 58
Monitoring-Baum 56
Monitoring-Services 44
mySAP Business Suite 20
mySAP CRM 229
mySAP Enterprise Portal 229
mySAP Supplier Relationship Management (mySAP SRM) 31
mySAP Workplace 33
mySAP-Lösungen 29

N

Nested-Loop-Join 437
Netscape-Server-API 528
NetWeaver LifeCycle Management 42
Netzwerk 84, 177, 207
Netzwerkproblem 177
New Visual Design 263
Nicht-Dialog-Workprozess 309
NRIV 375, 380, 386
NRIV_LOKAL 382
NSAPI 528
Numernkreisstand 385
Nummernkreis 380
Nummernkreisintervall 380
Nummernkreisobjekt 380
Nummernkreispufferung 338, 380, 384, 385, 538

O

Object Linking and Embedding siehe OLE
OLAP 227, 528
OLE 528
OLTP 227, 528
Online Analytical Processing 528
Online Transaction Processing 528
Open 352
Optimierer siehe Datenbankoptimierer
Optimierung von SQL-Anweisungen 395
Optimierungskonzept 38
Optimizer 528
ORACLE 480, 485
OS siehe Betriebssystem

P

- Package-Puffer (Package Cache) 89
- Pages 87
- Paging 529
- Paging-Datei 315, 335
- PAI 529
- Parameteränderung 84
- Parsen 404
- Partieller Tabellenpuffer (TABLP) 337
- PBO 529
- Performance 529
- Performanceaspekte 308
- Performanceforum 73
- Performanceindikatoren 52
- Performancelimitierende Faktoren 273
- Performancemanagement 29
- Performancemessung 256
- Performancemonitor 538
- Performancerelevante Konfigurationsparameter 497
- Performance-Trace 171, 172, 259
- Performante SQL-Programmierung 423
- Pflichteingabefeld 439
- PHYS_MEMSIZE 110, 311, 326
- Physische Lesezugriffe 87
- Physischer Hauptspeicher (RAM) 301, 302, 315, 334, 335
- Physischer Speicher 112
- Popup 529
- Präsentationsebene 33
- Prepare 176
- Primärindex 402, 413
- PRIV-Modus 119, 308, 328
- Proaktives Performancemanagement 19
- Procedure Cache 481
- Procedure-Puffer 481, 482
- Process After Input 529
- Process Before Output 529
- Processing-Zeit 138, 140, 143
- Profilparameter 332
- Program Global Area (PGA) 480
- Prozess-ID 117
- Puffer 36, 498
- Puffereinstellung 108, 109, 445
- Pufferqualität 87
- Pufferstatus 354
- Puffersynchronisation 344, 364
- Pufferung einschalten 346
- Pufferungsart 339
- Pufferzugriff 342
- PXA_NO_SHARED_MEMORY 329

Q

- Q-API 529
- Queue Application Programming Interface 529
- Queued RFC (qRFC) 238
- Quick Sizer 222

R

- R/3 529
- RAID 529
- RBO 407, 409
- RDBMS 529
- rdisp/atp_server 205, 390
- rdisp/bufrefmode 345
- rdisp/bufreftime 345
- rdisp/enqname 205
- rdisp/max_wprun_time 210
- rdisp/mshost 205
- rdisp/PG_MAXFS 314
- rdisp/PG_SHM 314
- rdisp/ROLL_MAXFS 305, 331
- rdisp/ROLL_SHM 110, 305
- rdisp/vb_dispatching 215
- rdisp/vbstart 214
- Read Random Hit Ratio 476
- Rechner 75
- Rechnerprofil 148
- Recovery 47
- Recursive Call 481
- Redundant Array of Independent Disks 529
- Reisepass 158, 169
- Relational Database Management System 529
- Remote Function Call (RFC) 32, 210, 237, 529
- Reopen-Operation 175
- Reorganisation
 - von Indizes 444
- Reprepare Ratio 478
- Request 352, 360, 361
- RESB 65, 387, 444
- Ressourcenbedarf 538
- RFC siehe Remote Function Call
- RFC-Client-Profil 249
- RFC-Grundlagen 237
- RFC-Last 146
- RFC-Statistiken 249
- RFC-Trace 171, 180
- RFC-Verbindung 243
- RFC-Verbindungen

- überwachen 246
- RFC-Zeit 242
- Roll Memory 331, 529
- Roll-in 136, 305, 529
- Roll-out 136, 304, 529
- Roll-Puffer 326
- Roll-Wartezeit 241, 257
- Roundtrip 257
- Row Cache 480
- Row-ID 403
- rsdb/max_blocking_faktor 434
- rsdb/obj/buffersize 390
- rsdb/obj/max_objects 390
- rstr/file 173
- rstr/max_diskspace 173
- Rule-Based Optimizer (RBO) siehe RBO

S

- SAP Advanced Planner and Optimizer (SAP APO) 31
- SAP Application Benchmark Performance Standard 222
- SAP Application Benchmark Performance Standard siehe SAPS
- SAP Business Connector (SAP BC) 30, 524
- SAP Business Information Warehouse (SAP BW) 30
- SAP Customer Relationship Management (SAP CRM) 31
- SAP EarlyWatch Alert 146, 156, 248
- SAP Easy Access Menu 263
- SAP Enterprise Portal (SAP EP) 30
- SAP Exchange Infrastructure (SAP XI) 30
- SAP GoingLive Check 223
- SAP GUI 255, 529
- SAP GUI 4.6 255
- SAP GUI for HTML 32, 266, 268, 269
- SAP GUI for Java Environment 32, 33
- SAP GUI for Windows 33, 270
- SAP GUI for Windows Environment 32
- SAP Internet Transaction Server (SAP ITS) 30, 265, 528
- SAP J2EE Engine 30, 285
- SAP liveCache 30
- SAP Logical Unit of Work (SAP LUW) 373
- SAP NetWeaver 30
- SAP Online Store 229
- SAP Quick Sizer 335

- SAP R/3 20
- SAP R/3 Enterprise 31
- SAP Service Marketplace 335, 533, 537
- SAP Solution Manager 42, 50, 67, 68, 70
- SAP Standard Application Benchmarks siehe Benchmark
- SAP Strategic Enterprise Management (SAP SEM) 31
- SAP System Identifier 530
- SAP Web Application Server 33
- SAP-Applikationsinstanz 75
- SAP-Basis 29
- SAPCCMSR 169
- SAP-EG Memory 312
- SAP-Enqueue 372, 373, 378, 379
- SAP-Extended Memory 109, 303, 305, 308, 326, 334
- SAP-Heap Memory 109, 303, 306, 322, 331, 334
- SAP-Instanz 75, 231, 325
- SAP-Kernel 331
- SAP-Komponenten 29
- SAP-Online-Hilfe 533
- saposcol 77
- SAP-Paging Memory 312, 313, 334
- SAP-Parameteränderung 113
- SAP-Puffer 107, 156, 325, 334
- SAP-Pufferung 427
- SAP-Roll Memory 109, 303, 304, 334
- SAP-Roll-Datei 304
- SAP-Roll-Puffer 304
- Saprouter 529
- SAPS 222, 529
- SAP-Service 201, 204
- SAP-Speicherbereich 314, 324
- SAP-Speicherkonfiguration 106, 119
- SAP-Speicherkonfigurationsmonitor 106
- SAP-Speicherverwaltung 150
- SAP-System 76, 371
- SAP-Systemservice 529
- SAP-Tabellenpufferung 337, 350
- SAP-Transaktion 373
- SAP-Workprozess 35, 115, 326, 334
- SAP-Workprozessübersicht 115
- Savepoint 91
- Schaltfläche 530
- Schnittstellen 237
- Schreib/Lese-(I/O-)Probleme 82, 98
- Sekundärindex 402, 403, 411, 413
- Select-*-Klausel 432

Semaphoren 118
 Sequential Read 175
 Sequenzielles Lesen 484, 485, 486, 487, 488, 490, 491, 492, 493, 494, 495
 Server 75, 76, 530
 Server-Destination-Statistiksätze 249
 Serverkonsolidierung 231, 232
 Service Level Management (SLM) 45, 46, 530
 Service Level Reporting 50
 Session Manager 530
 SET UPDATE TASK LOCAL 218
 SET_PARAMETER_MEMORY_OVERFLOW 329
 Shared Cursor Cache 95, 467
 Shared Memory 302, 530
 Shared Pool 480
 Shared SQL Area 94, 95, 96, 366, 396, 397, 400, 422, 467, 480
 Shared SQL Cache 95, 467
 Shared-Sperren 389
 SID 530
 Single Activity Trace 197
 Sizing siehe Hardwaresizing
 Skalierbarkeit 37
 Skeleton 477
 SMLG 209
 Snapshot Information 195
 Solution Manager 60
 Sort Heap 89
 Speicherbelegung 307
 Speicherkonfiguration 110, 301
 Speicherverwaltung 500, 537
 Sperren 371, 387, 530
 Sperren mit Menge 205, 389, 390, 391
 Sperrkonzept 371
 Spezielles Performanceproblem 143, 150
 Spool-Service 204, 205
 SQL 530
 SQL-Anweisung 93, 94, 402, 418, 423, 431, 467
 SQL-Coding 431
 SQL-Server 481, 495
 SQL-Trace 171, 173, 178, 198, 366, 396, 400, 429
 Stammdaten 349
 Standard Application Benchmark 530
 Statistiksätze
 verteilte 158
 Statistischer Einzelsatz 165, 198
 Statistischer Satz 165
 Stillstand der Datenbank 103
 STORAGE_PARAMETERS_WRONG_SE
 T 322, 329, 331, 334
 Stored Procedure 481
 Structured Query Language 530
 SUBMIT 304
 Support Package 531
 Swap Space siehe Auslagerungsspeicher
 Synchroner Verbuchung 218
 synchroner RFC 238
 System Global Area (SGA) 480
 SYSTEM_NO_MORE_PAGING 314
 SYSTEM_NO_ROLL 329
 Systemkonfiguration 537
 Systemkonsolidierung 231
 Systemlandschaft 230, 531
 Systemverfügbarkeit siehe High Availability

T
 Tabellengröße 361
 Tabellenpuffer 168
 Tabellenpufferung 339
 Tabellensperren 378
 Tabellenstatistiken 101, 409, 413
 Tabellenstatistiken administrieren 411
 Tabellenzugriffsstatistik 351
 Task-Typ 134
 TCP/IP 531
 TCURR 363
 TDC 531
 Technische Analyse 338
 Technische Optimierung 18
 Technisches Tuning 65
 Temporäre sequenzielle Objekte 531
 Temporäres Performanceproblem 144
 TemSe 531
 Teure SQL-Anweisung 92, 96, 396
 TMS 531
 TO 531
 Trainings 533
 Transaktion 373, 531
 Transaktionaler RFC 238, 252, 531
 Transaktionscode 509, 531
 Transaktionsprofil 150
 Transaktionsschritt 132, 135
 Transaktionsvarianten 441
 Transmission Control Protocol/Internet Protocol 531
 Transport 531

Transport Domain Controller 531
Transport Management System 531
Transport Organizer 531
Transportdomäne 531
Trefferrate (hitratio) 87
tRFC-Tabelle 253
TSV_TNEW_PAGE_ALLOC_FAILED 329
TSV_TNEW_PG_CREATE_FAILED 314

U

Übertragene Datenmenge 257
Überwachung von Sperren 375
Überwachungskonzept 38
Überwachungsmonitor 53
Uniform Resource Locator 531
UNIX 28, 82, 323
UNIX-Betriebssystem 320
Unternehmens-IMG 531
URL 531
User Call 481

V

V1-Verbuchung 216
V2-Verbuchung 216
V3-Verbuchung 216
VBBE 387
VBDATA 215
VBHDR 215, 360, 415
VBMOD 215
Verbuchung 104, 213
Verbuchung-Dispatching 215
Verbuchungsart 216
Verbuchungsauftrag 214
Verbuchungsservice 204
Verbuchungsworkprozess 215
Verdrängungen (Swaps) 108, 346, 350
Verfügbare Adressraum 319
Verfügbare Speicher 317, 319
Verteilte Installation 204
Virtuell benötigter Speicher 317
Virtueller Hauptspeicher 335
Virtueller Speicher 301, 531
Vollständige Pufferung 340

W

WAN 532
WBO 532
Web-Anbindung 269
Web-Anwendung 269, 296
Web-Browser 524

Web-RFC 266, 268
WGate 265, 271
Where-Klausel 426
Wide Area Network (WAN) 37, 532
Windows 28, 321, 324
Windows NT 82
Workbench Organizer 532
Workload-Analyse 157
Workload-Analyse 129, 135, 142
Workload-Monitor 130, 134
 zentraler 157, 158
Workprozess 118, 120, 206, 532
Workprozess-Typen 132
Workprozessübersicht 115
World Wide Web 532
WP 532
WWW 532

X

XML 532

Z

Zeitflusshierarchie 187
Zentrale Einzelsatzstatistik 169
Zentrale Installation 204
Zentraler Überwachungsmonitor 42
Zentraler Workload-Monitor 43
Zentrales Monitoring 41
Zero Administration Memory Management 110, 311, 335
ztta/roll_area 305, 310
ztta/roll_extension 306, 310, 327, 331
ztta/roll_first 307, 308, 310