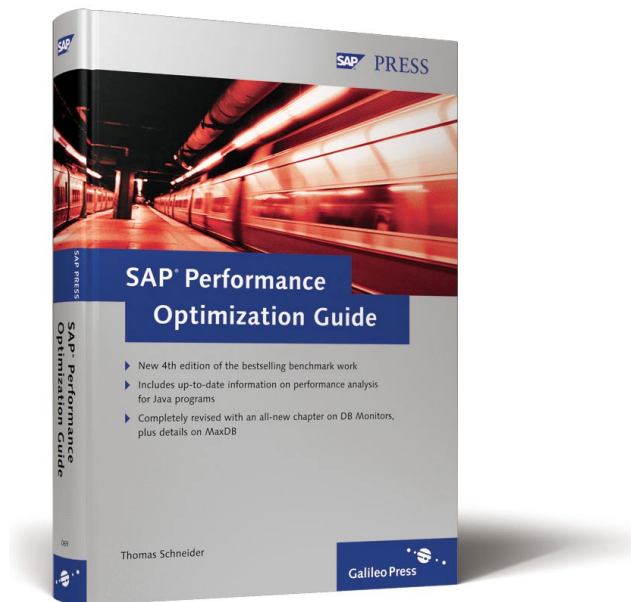


Thomas Schneider

# SAP® Performance Optimization Guide

Analyzing and Tuning SAP Systems



**SAP** PRESS

# Contents

<b>Foreword to the Series of Books</b>	<b>13</b>
<b>Acknowledgments</b>	<b>15</b>
<b>Introduction</b>	<b>17</b>
Proactive Performance Management .....	17
From SAP R/3 to mySAP Business Suite .....	20
About This Book .....	23
<b>1 Performance Management of a mySAP Solution</b>	<b>29</b>
1.1 mySAP Solution Architecture .....	29
1.1.1 mySAP Solutions and SAP Components .....	29
1.1.2 Client/Server Architecture .....	33
1.2 The Monitoring and Optimization Plan for a mySAP Solution .....	38
1.2.1 Requirements of a Monitoring and Optimization Plan .....	38
1.2.2 Tools and Methods for the Monitoring and Optimization Plan ..	41
1.2.3 Service Level Management .....	44
1.2.4 Continuous Monitoring .....	50
1.2.5 Plan for Continuous Performance Optimization .....	60
1.2.6 SAP Solution Manager .....	64
1.3 Summary .....	68
<b>2 Monitoring Hardware, Database and SAP Basis</b>	<b>71</b>
2.1 Basic Terms .....	71
2.2 Monitoring Hardware .....	72
2.2.1 Analyzing a Hardware Bottleneck (CPU and Main Memory) .....	73
2.2.2 Analyzing Read/Write (I/O) Problems .....	78
2.2.3 Other Checks With the Operating System Monitor .....	80
2.2.4 Summary .....	80

2.3	<b>Monitoring the Database</b> .....	81
2.3.1	Analyzing the Database Buffer .....	82
2.3.2	Identifying Expensive SQL Statements .....	87
2.3.3	Identifying Read/Write (I/O) Problems .....	93
2.3.4	Other Checks on the Database .....	94
2.3.5	Summary .....	100
2.4	<b>Analyzing SAP Memory Management</b> .....	101
2.4.1	Analyzing SAP Buffers .....	101
2.4.2	Analyzing SAP Extended Memory, SAP Heap Memory and SAP Roll Memory .....	103
2.4.3	Displaying the Allocated Memory .....	105
2.4.4	Other Monitors in the Memory Configuration Monitor .....	108
2.4.5	Summary .....	108
2.5	<b>Analyzing SAP Work Processes</b> .....	109
2.5.1	Fields in the Work Process Overview .....	110
2.5.2	Analyzing Work Processes .....	113
2.5.3	Systemwide Work Process Overview .....	115
2.5.4	Monitoring the Dispatcher Queue .....	117
2.5.5	Summary .....	117
2.6	<b>Summary</b> .....	119

### **3 Workload Analysis 123**

3.1	<b>The Workload Monitor</b> .....	124
3.1.1	Functions and Availability .....	124
3.1.2	Working With the Workload Monitor .....	124
3.1.3	Technical Settings for the Workload Monitor .....	128
3.2	<b>Workload Analysis</b> .....	128
3.2.1	Course of a Transaction Step .....	129
3.2.2	Interpreting Response Times .....	132
3.2.3	Activity, Throughput and Load .....	134
3.3	<b>Performing Workload Analysis</b> .....	136
3.3.1	Analyzing General Performance Problems .....	137
3.3.2	Analyzing Specific Performance Problems .....	144
3.4	<b>Application Monitor</b> .....	147
3.4.1	User Profile .....	147
3.4.2	Load per SAP Application Module .....	149
3.4.3	SAP Buffers .....	149
3.5	<b>Cross-Component Workload Analysis</b> .....	150
3.6	<b>Summary</b> .....	154

### **4 Identifying Performance Problems in ABAP and Java Programs 157**

4.1	<b>Single Record Statistics</b> .....	157
4.2	<b>Central Single Record Statistics (Functional Trace)</b> .....	161

4.2.1	Working with the Central Single Records Statistics .....	161
<b>4.3</b>	<b>Performance Trace .....</b>	<b>163</b>
4.3.1	Activating a Performance Trace .....	163
4.3.2	Evaluating an SQL Trace .....	165
4.3.3	Other Tools in the SQL Trace .....	169
4.3.4	Evaluating an RFC Trace .....	171
4.3.5	Evaluating an Enqueue Trace .....	172
<b>4.4</b>	<b>Performance Analysis With ABAP Trace (Runtime Analysis) .....</b>	<b>173</b>
4.4.1	Activating an ABAP Trace .....	173
4.4.2	Evaluating an ABAP Trace .....	174
4.4.3	Using Function Variations .....	175
4.4.4	Activating the Runtime Analysis for BSP Applications .....	177
4.4.5	Outlook: Single Transaction Analysis .....	177
4.4.6	Using the Single Transaction Analysis .....	179
<b>4.5</b>	<b>Analyzing Memory Usage with ABAP Debugger .....</b>	<b>180</b>
4.5.1	Common Performance Problems .....	181
<b>4.6</b>	<b>Performance Statistics for J2EE Applications (JARM) .....</b>	<b>183</b>
4.6.1	Working with JARM Statistics in SAP Enterprise Portal .....	185
4.6.2	JARM Trace (Single Activity Trace) .....	188
<b>4.7</b>	<b>Summary .....</b>	<b>188</b>

## **5 Workload Distribution 191**

<b>5.1</b>	<b>SAP Services .....</b>	<b>191</b>
5.1.1	Distributing Message, Enqueue and ATP Services .....	194
5.1.2	Distributing Dialog, Background and Spool Work Processes .....	195
5.1.3	Distributing Users and Work Processes Over CPU Resources .....	196
<b>5.2</b>	<b>Dynamic User Distribution (Logon-Groups) and Operation Modes .....</b>	<b>198</b>
5.2.1	Configuring Dynamic User Distribution .....	199
5.2.2	Planning Operation Modes .....	201
<b>5.3</b>	<b>Update Processing .....</b>	<b>202</b>
5.3.1	Monitoring Update Requests .....	202
5.3.2	Distributing Update Work Processes .....	204
5.3.3	Selecting Types of Update .....	205
<b>5.4</b>	<b>Hardware Sizing .....</b>	<b>208</b>
5.4.1	Planning Initial Sizing .....	210
5.4.2	SAP Standard Application Benchmarks .....	214
5.4.3	Planning Hardware Capacity to Deal With Increased Workload, Change of Release or Migration (Re-sizing) .....	216
5.4.4	New Challenges Arising From the Internet .....	217
<b>5.5</b>	<b>Planning the System Landscape .....</b>	<b>219</b>
5.5.1	Distribution of SAP Instances .....	220
5.5.2	Hardware Consolidation .....	221
5.5.3	Several SAP Systems on one Database .....	222
<b>5.6</b>	<b>Summary .....</b>	<b>222</b>

## **6 Interfaces 225**

6.1	RFC Fundamentals .....	225
6.1.1	Concepts .....	225
6.1.2	Course of an RFC .....	228
6.2	Interfaces to External Systems .....	230
6.2.1	Configuring and Testing RFC Destinations .....	231
6.2.2	Monitoring Inbound and Outbound Load .....	234
6.2.3	Configuring the Parallel Processes with Asynchronous RFCs .....	239
6.2.4	Monitoring Data Transfer With Transactional RFCs .....	239
6.3	Summary .....	241

## **7 SAP GUI and Internet Connection 243**

7.1	SAP GUI 4.6 .....	243
7.1.1	Interaction Model and Measuring Performance .....	244
7.1.2	Analyzing and Optimizing the Performance of GUI Communication .....	245
7.2	SAP Web Application Server (SAP Web AS) .....	251
7.3	SAP Internet Transaction Server (SAP ITS) .....	252
7.3.1	ITS Fundamentals .....	252
7.3.2	Planning a Web Connection .....	256
7.3.3	Configuring ITS .....	257
7.3.4	ITS Administration Tools .....	260
7.3.5	Performing a Bottleneck Analysis for the ITS .....	262
7.4	Business Server Pages (BSP) and Internet Communication Manager (ICM) .....	263
7.4.1	SAP Business Server Pages Fundamentals .....	263
7.4.2	Configuring and Monitoring ICM .....	266
7.4.3	Carrying Out a Performance Analysis on BSPs .....	269
7.5	SAP J2EE Engine .....	272
7.5.1	SAP J2EE Engine Fundamentals .....	272
7.5.2	Tools for Configuration and Monitoring .....	277
7.6	End-to-End Performance Monitoring .....	279
7.6.1	Performing Runtime Analyses in the Web Browser .....	280
7.6.2	Continuous Monitoring of Web Applications With the CCMS ...	282
7.6.3	Non-SAP Tools .....	283
7.7	Summary .....	283

## **8 Memory Management 287**

8.1	Memory Management Fundamentals .....	287
8.1.1	Basic Terms .....	287

8.1.2	SAP Roll Memory, SAP Extended Memory, SAP Heap Memory .	289
8.1.3	SAP EG Memory and SAP Paging Memory .....	298
<b>8.2</b>	<b>Configuring and Monitoring SAP Memory Areas .....</b>	<b>299</b>
8.2.1	Monitoring Swap Space .....	301
8.2.2	Address Space Restrictions (32- and 64-bit Architecture) .....	304
8.2.3	Configuring and Monitoring SAP Memory Areas .....	309
8.2.4	Assistance With Troubleshooting .....	312
<b>8.3</b>	<b>Summary .....</b>	<b>318</b>

## **9 SAP Table Buffering 321**

<b>9.1</b>	<b>Preliminary Remarks Concerning Chapters 9, 10 and 11 .....</b>	<b>322</b>
<b>9.2</b>	<b>Table Buffering Fundamentals .....</b>	<b>323</b>
9.2.1	Buffering Types .....	323
9.2.2	Buffer Accessing .....	325
9.2.3	Buffer Synchronization .....	327
9.2.4	Activating Buffering .....	329
9.2.5	What Tables Should be Buffered? .....	331
<b>9.3</b>	<b>Monitoring SAP Table Buffering .....</b>	<b>333</b>
9.3.1	Table Access Statistics .....	334
9.3.2	Analyzing Buffered Tables .....	337
9.3.3	Analyzing Tables That are Currently not Buffered .....	342
9.3.4	Detailed Table Analysis .....	344
9.3.5	Monitoring Buffer Synchronization (DDLOG Entries) .....	346
9.3.6	Shared SQL Area and SQL Trace .....	348
<b>9.4</b>	<b>Summary .....</b>	<b>348</b>

## **10 Locks 351**

<b>10.1</b>	<b>Database Locks and SAP Enqueues .....</b>	<b>351</b>
10.1.1	Database Lock Concept .....	352
10.1.2	SAP Enqueues .....	352
<b>10.2</b>	<b>Monitoring Database Locks .....</b>	<b>354</b>
10.2.1	Database Locks .....	354
10.2.2	SAP Enqueues .....	357
<b>10.3</b>	<b>Number Range Buffering .....</b>	<b>359</b>
10.3.1	Number Range Buffering Fundamentals .....	359
10.3.2	Activating Number Range Buffering .....	363
10.3.3	Monitoring Number Range Buffering .....	364
<b>10.4</b>	<b>ATP Server .....</b>	<b>365</b>
10.4.1	ATP Server Fundamentals .....	366
10.4.2	Configuring the ATP Server .....	367
10.4.3	Monitoring the ATP Server .....	370
<b>10.5</b>	<b>Summary .....</b>	<b>371</b>

## **11 Optimizing SQL Statements 373**

11.1	Identifying and Analyzing Expensive SQL Statements .....	373
11.1.1	Preliminary Analysis .....	373
11.1.2	Detailed Analysis .....	376
11.2	Optimizing SQL Statements Through Secondary Indexes .....	379
11.2.1	Database Organization Fundamentals .....	379
11.2.2	Administration for Indexes and Table Access Statistics .....	389
11.2.3	Rules for Creating or Changing Secondary Indexes .....	394
11.3	Optimizing SQL Statements in the ABAP Program .....	400
11.3.1	Rules for Efficient SQL Programming .....	400
11.3.2	Example of Optimizing an SQL Statement in an ABAP Program ..	405
11.3.3	Presetting Field Values in Report Transactions .....	414
11.4	Summary and Related Tuning Measures .....	419

## **A Performance Analysis Roadmaps and Checklists 425**

A.1	Roadmaps .....	425
A.2	Checklists .....	431

## **B Database Monitors, Buffers, and SQL Execution Plans 441**

B.1	Database Process Monitor .....	441
B.2	Shared SQL Area Monitor .....	443
B.3	Hard Disk Monitor .....	446
B.4	Database Lock Monitor .....	446
B.5	Monitoring Database Buffers .....	448
B.6	Execution Plans for SQL Statements .....	458

## **C Configuration Performance Parameters 471**

C.1	SAP Buffer Parameters .....	472
C.2	SAP Memory Management Parameters .....	474
C.3	Additional Parameters .....	475
C.4	Internet Communication Manager .....	479
C.5	Internet Transaction Server .....	480

<b>D</b>	<b>Selected Transaction Codes</b>	<b>483</b>
<b>E</b>	<b>Review Questions and Answers</b>	<b>485</b>
<b>F</b>	<b>Glossary</b>	<b>497</b>
<b>G</b>	<b>Information Sources</b>	<b>507</b>
<b>H</b>	<b>Selected SAP Service Marketplace Notes</b>	<b>509</b>
	<b>Index</b>	<b>513</b>

# Foreword to the Series of Books

At SAP, our first priority is to ensure that the SAP software solutions in your enterprise run successfully and at a minimal cost. This "Lowest Cost of Ownership" is achieved with fast and efficient implementation, together with optimal and dependable operation. SAP Active Global Support is actively and consistently there to help you, with the new SAP Solution Management strategy. Throughout the entire lifecycle of a solution, SAP offers customers all necessary services, first-class support, a suitable infrastructure, and the relevant know-how. The new strategy is backed up by three powerful support programs: *Safeguarding*, or, in other words, risk management; *Solution Management Optimization*, which aims to optimize the customer's IT solution; and *Empowering*, which ensures a targeted, effective transfer of knowledge from SAP to the customer.

The imparting of knowledge is also one of the key aims of this book—part of the line of *SAP Technical Support Guides*. This series gives you a detailed overview of technical aspects and concepts for managing SAP software solutions. The topics dealt with in these books range from a technical implementation project to running a software system and the relevant database system.

Whether you are new to SAP system management or wish to gain further qualifications, you will benefit from the wealth of practical experience and first-hand information contained in these books. With this line of books, SAP also endeavors to help prepare you for qualification as a "Certified Technical Consultant". Please note, however: These books cannot replace, nor do they attempt to replace, personal experience gained from working with the various SAP solutions! Rather, the authors offer suggestions to help in your day-to-day work with the software. Innovation in SAP solutions always brings with it new challenges and solutions for system management. The demands made on the customer's own or external support organizations also increase. The expertise and knowledge of these organizations can be a great help in avoiding problems when using the software. Therefore, one of the core tasks of this series of books is to teach problem-solving skills.

Even in this Internet age, books prove to be an ideal medium for imparting knowledge in a compact form. Furthermore, their content complements the new service and support platform, the SAP Solution Manager, and other new services offered by SAP. The series provides

background knowledge on the operation and functioning of new SAP solutions and contributes to customer satisfaction.



**Gerhard Oswald**

Member of the Executive Board of SAP



**Dr. Uwe Hommel**

Senior Vice President of SAP

SAP Active Global Support

Walldorf, November 2005

# Acknowledgments

As a result of the huge success of the first edition of this book—one can say without a doubt that not only has it become a cornerstone in performance training among customers but also for many SAP employees—once again the need has arisen to publish an updated and extended third edition. It is once again a great pleasure to write the third updated and revised edition. For the most part, the topics are selected and presented based on the experience my colleagues and I made in real-life situations, working with many SAP systems in production operations—whether in SAP's Early Watch and GoingLive Check services, in training courses for performance analysis or, not least of all, during on-site analysis of systems with critical performance problems. Based on this experience we are confident that this book covers a broad range of important performance-related topics.

This book would not have been possible without the support of many competent discourse partners. First, I would very much like to mention our colleague and mentor Augustinus Wohlfahrt, whose sudden and untimely death left us all deeply shaken. As one of the initiators of this series of books, he played a central part in putting together this publication. I would like to dedicate this book to him.

In particular, I would also like to thank the following colleagues: 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 and Liane Will.

I would like to express my gratitude to Jens Claussen, Guido Derwand and Mandy Krimmel for their help with the sections on Business Server Pages and Java applications that have been added to the third edition. I would like to thank Brigitte Huy, Anja Kerber, Christian Knappke, Claudia Langner, Wladimir Maljutin, Isolde Savelsberg-Walter, and Gienek Wajda for their support in updating the sections on the different databases for the fourth edition.

I would like to thank Tomas Wehren and Florian Zimniak from Galileo Press for their outstanding support.

**Dr. Thomas Schneider**

SAP Strategic Research & Development

# Introduction

Why is the performance of your e-business application important? Users will only be motivated and work efficiently with an application if response times are good. A slow system leads to downtime and frustration. Should the situation deteriorate further, in the worst case, you no longer have the throughput necessary for running business processes. The results are overtime, delays in production and financial loss. In contrast, the systematic, proactive optimization of performance considerably increases the value of your e-business application.

Performance

The performance of a data processing system is defined as the system's ability to fulfill given requirements concerning response time and data throughput. The system may, for example, be required to achieve a throughput of 10,000 printed invoices in one hour or a response time of under one second for the creation of a sales order. Good performance is, however, not an absolute characteristic of an e-business application. Rather, it should always be viewed as relative to the demands made on the application.

## Proactive Performance Management

By *Performance optimization* in this book we refer to a process that *always includes five phases*: The first two phases are *understanding the business processes* and *setting and quantifying performance goals*. These steps involve all participating parties, that is to say, technicians and application experts. Optimization can only be successful on the basis of these prerequisites. Phases three to five involve the *systematic monitoring, identifying and analysis* of problems, the *implementation of optimization measures* and further analysis to *verify the success of the measures* introduced (Figure 1). We warn against randomly tinkering with configuration parameters and similar impulsive tuning measures! Rather, the objective of this book is to enable you to identify and analyze performance problems in order to deal with them purposefully.

Performance  
Optimization

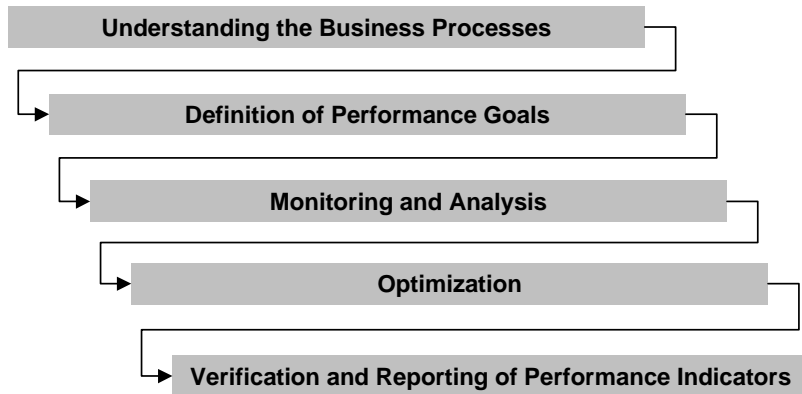


Figure 1 Performance optimization in five phases

**Technical Optimization**

From a technical point of view, an e-business application is made up of many different components. On the one hand there are the logical components: processes and services, threads or work processes and memory areas such as buffers and user contexts. On the other there are the physical components such as processors (CPU), main memory (RAM), hard disks and network segments. Each of these components allows for a maximum throughput and optimal response time. If the interplay between the components is not appropriately balanced or if an individual component has reached its performance limit, wait situations can occur which have a negative effect on throughput and response time. In this book, *Technical Optimization* refers to the identification, analysis and solution of such problems by tuning the components and distributing the workload that occurs within the system.

**Application Optimization**

The second important task of performance optimization is to avoid unnecessary workload. Performance can be weakened by inefficient programs or the inefficient use of programs. The optimization of individual programs is referred to as *application optimization*.

The goal of optimization is, first of all, to improve the system settings and the applications to achieve the desired performance on the basis of the existing hardware resources. If the existing resources are not sufficient, they must be extended according to the knowledge gained in the analysis.

**How much tuning is necessary?**

How much effort is involved in the performance analysis and tuning of a mySAP solution? The answer to this question depends largely on the size of the system. For a small or medium sized installation without modifications to the SAP standard and without customer developments,

it is normally sufficient to do performance optimization just before and shortly after the start of production and after large-scale changes, for example, after upgrades, large data transfers or client transports, or when new mySAP solutions or additional users are introduced in the system. Of course, it is also necessary to intervene when there are acute performance problems. The tuning potential and, with it, the effort involved in analysis and optimization increase in proportion to the size of the system. Experience shows that many performance bottlenecks are caused by customer developments and modifications to the standard SAP software. The most common reason for this is insufficient testing, but problems may also arise as a result of time constraints or a lack of experience on the part of the developer. The extreme case would be a large, constantly developing installation with several hundred users, complicated process chains, a dozen or more developers (often from different consulting firms, working on the system at different times and in different places) and outsourced system management. In such a system environment it is absolutely necessary that a small group of administrators and developers has an overview of the entire system and keeps an eye on performance.

SAP's remote services offer help with performance analysis and tuning. These are, namely, the GoingLive™ Check, which enables your system to make a smooth transition to production operation and the EarlyWatch® Service, which monitors your system and suggests additional optimizations.

How does *proactive performance management* help you attain your objective of successfully running of an e-business application? Two influencing factors should be borne in mind if this objective is to be achieved: the satisfaction of users and the costs of running the e-business application. Operating costs come, on one hand, from the cost of hardware (infrastructure, CPU, main memory, hard disks and networks) and personnel (administration, maintenance, fault analysis). However, the costs that arise if an application is not available or does not achieve the required performance should not be overlooked. In these cases losses incurred in a few hours or days can exceed the average amount invested in proactive performance optimization in one year. The cost of these risks must be compared to the cost of proactive performance management. The following table demonstrates the value of proactive performance management using two concrete examples.

**Proactive  
performance  
management**

Proactive measure	Effect on the system	Immediate value thanks to increased user satisfaction	Immediate value thanks to lower operating costs	Diminished risk of deterioration
SQL statement optimization	Reduction of the database load	Faster response times for certain transactions	Hardware investments (database server, memory system) can be stretched	Overloading of the database system can be avoided
Proactive data management (Data avoidance, archiving, reorganization)	Database growth reduced Shorter times for maintenance work on the database (backup/recovery, upgrade, migration, system copy)	Faster response times for certain transactions Shorter downtime during maintenance work	Hardware investments can be stretched Lower personnel requirements for maintenance work	Database size remains "manageable"

**Table 1** Examples of the value of proactive performance management

## From SAP R/3 to mySAP Business Suite

With the development of the Internet there has been a shift of paradigm in the world of business software: software is no longer aimed at highly specialized employees; rather, it is aimed at Internet or intranet users.

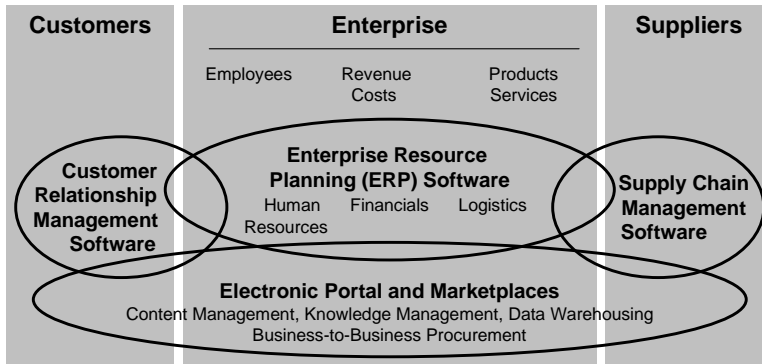
With SAP R/3, the classical strategy of process automatation was based on highly-specialized users accessing their ERP (Enterprise Resource Planning) system from fixed work centers via installed SAP GUIs. The role of these specialized agents, who had to be trained to use the software, is becoming unnecessary in many cases. Instead, the end user can get direct access to the enterprise's ERP systems via Internet or an Intranet. Today, for example, in many enterprises the employees can enter their work and absence times, travel expenses etc. into the system themselves via the Internet, where previously this would have been done by central users. Increasingly, customers are ordering their products directly on the

Internet and no longer by means of a letter, fax or telephone call to a sales center. Experts also speak of the "popularization" or even "democratization" of ERP software. The following statement from an analyst sums up the issue in a nutshell: "ERP software is made for clerks, but we aren't talking about clerks anymore." With the changeover from R/3 to the e-business platform mySAP Business Suite, SAP has completed the change of paradigm in the development of their software, with the consistent expansion of their Web technology on the one hand and more emphasis on solutions, on the other.

SAP describes the significance of the mySAP Business Suite e-business platform for their customers as follows: "In order to be able to survive profitably and competitively in the Internet influenced business world of today, successful companies must be put in a position that enables them to collaborate beyond traditional corporate boundaries and cooperate within virtual global networks. In mySAP Business Suite, SAP combines sound business and industry-specific know-how with a comprehensive e-business platform for solutions, services and technologies. By linking their business strategies with mySAP Business Suite, enterprises achieve a long-term competitive advantage, assessable added value and the best possible return on investment."

In an advertising slogan SAP makes the claim that their software does not merely execute "add to shopping basket", but also deals with "load to truck"—in allusion to the unsuccessful Internet connection attempts of many enterprises with "isolated applications", not linked with their business processes. The e-business platform mySAP Business Suite on the other hand, offers integrated solutions—from Customer Relationship Management to Enterprise Resource Planning through to Supply Chain Management.

What are the consequences of the changeover from SAP R/3 to mySAP Business Suite for system management and for performance management in particular? There are two specific consequences that we would like to deal with here in the first instance: firstly, the increasing user demands and secondly, the growing complexity of the system landscape linked with the growing need to recognize IT as a service.



**Figure 2** From SAP R/3 to mySAP Business Suite: SAP R/3 is typical ERP software covering the areas of human resources, financials and logistics. mySAP Business Suite includes solutions for the enhanced linking of the enterprise with customers and partners, for example, via the Internet.

**User expectations** The expectations of a user concerning the usability and performance of an e-business solution are disproportionately higher than the classical employee's expectations as regards their ERP system. The employee relies on his or her own ERP system, and if it normally helps to make his day-to-day work easier, he accepts it and even puts up with minor errors or weak points in performance. The Internet user is quite different: If the applications offered on the Internet do not work easily and effectively, users can immediately change over to the competition and, for example, make their purchases there ("the competition is only a mouse-click away"). In addition, the Internet does not finish work at 5 p.m.—an e-business solution on the Internet will be required to be available and working well 365 days a year, 24 hours a day.

**IT services** The demands for an open, flexible software architecture require specialized, independently running software components, which are linked via interfaces. That means that a business process involves several software components. The constantly growing number of solutions and components presents an administrative challenge for computer centers—the number has gone from the "manageable" SAP R/3 (with SAP instances, database, hardware/operating system) to a constantly growing range of technologies—including products that SAP does not produce itself, but that it offers as a reseller.

Consistent with this, the business process operator counters this trend by integrating more and more service partners into the running of the business process. This outsourcing may involve only the hardware (computer performance, hard disk memory, network resources etc.) or

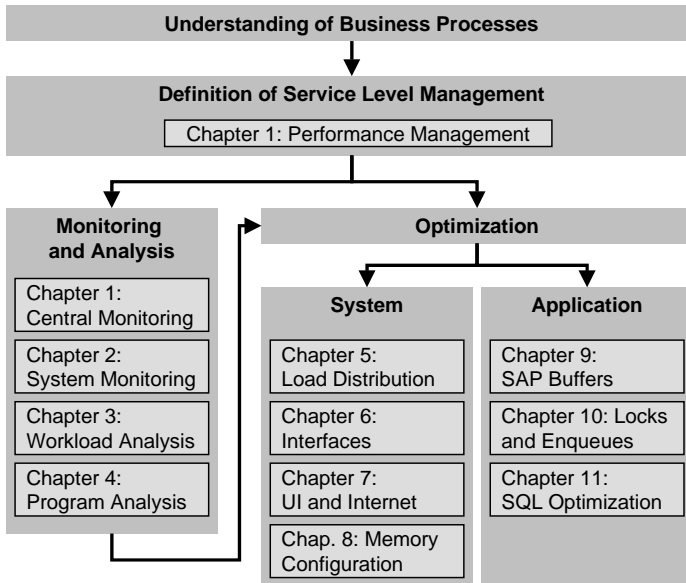
also the application itself (application service providing, ASP). For example the services of an Internet product catalog can be completely given over to a service provider instead of operating the catalog software in the enterprise itself. This means that it is not only necessary to monitor hardware and software components, but monitoring must also go beyond company and component boundaries.

Overall, completely new requirements arise for the administration and monitoring of mySAP solutions which cannot be dealt with using past concepts.

## About This Book

The methods for performance analysis and optimization presented in this book reflect the methods initially used by experts in the *EarlyWatch*<sup>®</sup> service and the *GoingLive*<sup>™</sup> Check and they are included in the SAP Basis training courses *BC315 Workload Analysis* and *BC490 Optimization of ABAP programs*. This is the fourth edition of this book, and with each new edition I take the opportunity to describe thoroughly the current trends in product development at SAP and, wherever relevant, to consider developments in the IT world in general.

Structure of the book



**Figure 3** The chapters in this book and how they correspond to the phases of performance optimization

**Chapter 1**, "Performance Management of a mySAP Solution", is directed at SAP administrators, SAP consultants, application developers and SAP project leaders. It deals with the following fundamental questions about performance analysis on a non-technical level:

- ▶ Which preventative measures must be taken to guarantee the optimal performance of a mySAP solution?
- ▶ What performance tuning measures should be taken into consideration?
- ▶ Who is involved in the tuning process?

The service provided to the user frequently turns out to be the combination of a number of different services carried out by a network of partners. Many parts are provided by different, sometimes external, service providers. In order to master this complexity, many service providers and customers implement *Service Level Management (SLM)*. The SLM concept calls for a structured, proactive method to ensure an adequate service level to the users of an IT application, taking into account both cost-efficiency and the business objectives of the customers. In this book, we'll describe the tools and methods you have to use in order to implement an SLM for a mySAP solution.

Performance analysis is presented in **Chapters 2 to 4**. After having read these chapters you should be in a position to carry out a systematic performance analysis.

In this book, we initially follow the bottom-up analysis strategy, starting in **Chapter 2**, "Monitoring Hardware, Database and SAP Basis", with an examination of the operating system, database, SAP memory management and SAP work processes. At the same time, solution proposals are provided which should enable the administrator or consultant to solve the most important performance problems. For small and medium sized installations this level of tuning is often sufficient.

Then in **Chapter 3**, "Workload Analysis", the more complex workload analysis is discussed as an example of top-down analysis. In **Chapter 4**, "Identifying Performance Problems in ABAP and Java Programs", you will find methods for analyzing individual programs, using the tools SQL trace and ABAP debugger, among others.

The remaining **Chapters 5 to 11** present knowledge necessary for a more indepth performance analysis. They are intended for SAP consultants responsible for the efficient functioning of large systems who seek and need to reach the full tuning potential of their system. Chapters 5 to 11

are independent units to a large extent, and they can be read in any order once you are familiar with the content of the first four chapters.

The topics are:

- ▶ Chapter 5, "Workload Distribution": Optimal workload distribution of dialog, update and background requests helps to ensure optimal use of hardware and avoids bottlenecks brought about by nonoptimal configuration. Server consolidation, i.e. the bundling of all services in a few powerful machines, has without doubt become an important trend in the IT market in recent years. Server consolidation usually takes place when the 64-bit memory-allocation technology is implemented. We'll describe what you must take into account in order to efficiently use these technologies.
- ▶ Chapter 6, "Interfaces": The performance of interfaces between software components contributes greatly to the performance of the entire solution. E-business solutions that consisted solely of monolithic R/3 systems were rarely used even in the past. It is, rather, open solutions comprising several components connected to each other via interfaces that represent the standard.
- ▶ Chapter 7, "SAP GUI and Internet Connection": Analysis and configuration recommendations demonstrate the optimization potential of linking GUIs (classical SAP GUI or Web browser) with the application.

With its EnjoySAP initiative born in 1997, SAP focused increasingly on user-friendly and intuitive software and developed personalized graphical user interfaces for SAP software. The entirely revamped software was released for the first time when SAP R/3 version 4.6 and other software components were brought to market. In terms of technology, the new design was based on a completely new interaction model for the communication between the presentation and the application layers, the so-called *controls*.

Since 1997 (SAP R/3 version 3.1) it has been possible to directly access an SAP R/3 system by means of a Web browser via SAP Internet Transaction Server (SAP ITS) and a Web server. In SAP R/3 4.6 the entire range of functionalities is web-enabled throughout. Other important SAP E-business solutions also use SAP ITS, which means that it will be a major strategic component for many customers and their mySAP solution landscapes in the coming years.

New SAP products are based on two new SAP technologies for the connection of Web front ends: the Business Server Pages (BSP) and the

use of Java-based front-end technologies. The third and fourth editions of this book contain separate sections dedicated to these technologies.

- ▶ Chapter 8, "Memory Management": The configuration of the memory areas allocated by the SAP component has a considerable influence on performance.
- ▶ Chapter 9, "SAP Table Buffering": Buffering tables on the application servers speeds up access to frequently read data and helps ease the load on the database.
- ▶ Chapter 10, "Locks": Database and SAP locks ensure data consistency. With an optimized administration of locks (for example, with the ATP server or by buffering number ranges) throughput bottlenecks can be avoided.
- ▶ Chapter 11, "Optimizing SQL Statements": Ineffective SQL statements make heavy demands on the database and thus become a problem for the performance of the entire application. An entire chapter is therefore devoted to optimizing SQL statements.

**Target groups** We have already differentiated between technical optimization and application optimization. Chapters 2, 5, 6, 7 and 8 deal with technical optimization and are mainly aimed at SAP system administrators and technical consultants. Chapters 4, 9, 10 and 11 deal with the analysis and tuning of individual programs (or applications) and as such form part of application optimization. This part of the analysis is also of interest to those responsible for SAP applications (employees in the department, application consultants and developers). Chapters 1 and 3 are relevant to both areas.

**Prerequisites** This book assumes both theoretical and practical knowledge of the administration of SAP components. You should be familiar with the use of *Computer Center Management Systems (CCMS)* in particular. "R/3 System Administration" (see appendix G, "Information Sources") should serve as good preparation. Parts of this book, e.g. Chapters 4, 9, 10 and 11 assume familiarity with the programming language ABAP, with the functioning of relational databases and with SQL.

The book does not cover the following topics:

► **Hardware and network tuning**

Although this book helps you to identify a bottleneck in the CPU, main memory, I/O or network, a detailed analysis requires the tools of the hardware or network provider. In view of the enormous number of products offered, this area (especially tuning the hard disk) cannot be included.

► **Databases**

In the Computer Center Management System (CCMS), SAP offers tools that standardize most administrative and analysis tasks for different database systems. However, for those who wish to do more in-depth database tuning, you need to know the different database system architectures. It is impossible to go into the fine points of all seven database systems that can be used in conjunction with mySAP Business Suite in sufficient detail in this book. In any case, this is not necessary because reference material on tuning is available for all database systems. This book cannot replace such material, nor does it endeavor to do so. Rather, the emphasis in this book is on the SAP-specific context of database tuning and on explaining concepts common to all database systems. The concrete examples used always refer to individual database systems. In the appendix you will find an overview of the most important monitors for analyzing all database systems.

► **Application tuning**

Many problems with performance can only be solved with detailed knowledge of the application and of the individual mySAP modules. A change to the customizing settings often solves the problem. This book does not provide know-how for tuning individual mySAP modules. However, this book does provide you with analysis strategies so that you can limit performance problems to certain applications and then consult the appropriate developer or consultant.

One question that was heatedly discussed before this book was published concerns the extent to which release-dependent and time-dependent information should be included; this would affect, for example, menu paths, recommendations for configuration parameters and guide values for performance counters. A new version, a patch (for the SAP component, the database or the operating system), a new generation of computers—these and other factors could render previous information useless overnight. In the worst case, obsolete recommendations could even have negative effects on performance. We are aware of this risk.

Nevertheless we have decided to include time-dependent information and rules in the book. Only in this way can the books of this series be used as works of reference for daily work in SAP administration. On the other hand, it is clear that this is not a book of fixed rules and regulations, and anyone who considers performance optimization to mean mechanically following rules is mistaken. This book cannot replace direct analysis of the solution, SAP online help or up-to-date SAP Notes on SAP's Service Marketplace—it only hopes to support them.

**SAP NetWeaver '04** All details on menu paths, references to performance monitor screens and guideline values for performance counters refer to SAP NetWeaver '04.



You will find important Notes and Tips in sections marked with this symbol.



This symbol indicates an Example.



Caution, sections marked with this symbol warn of potential errors and pitfalls!

**UNIX Windows** Sections referring to specific features of the UNIX and Windows operating systems will be marked as such.

## 3 Workload Analysis

Workload analysis gives reliable data on throughput, load and response times for the SAP system and its components. As described in the Introduction, an experienced performance analyst begins by using a workload analysis to reveal areas of the SAP system that have noticeable performance problems, and then proceeds with a more detailed top-down analysis.

Example: Assuming that you have systematically performed the analyses explained in the last chapter and have discovered several problems both in the database area and in SAP memory configuration. How can you determine which problem is the most serious and requires the most urgent attention? The workload analysis can provide the answer.



Workload analysis examines the various response times measured by the system. The kinds of performance problems identified by workload analysis are those that negatively affect throughput and response time and are known as *bottlenecks*. Bottlenecks can critically affect production operation and therefore require speedy removal. Workload analysis can also be used to prioritize performance problems.

In addition, workload analysis reveals the load distribution for each application's programs or transactions and indicates which of these are placing the greatest load on the SAP system. Workload analysis should therefore be the starting point for a detailed application analysis.

The first section of this chapter explains the basic concepts of workload analysis. After an introduction to the Workload Monitor, there is an explanation of which statistics are measured in units of time by the SAP system and how you can use these measurements to identify performance problems. The second section of the chapter provides recommendations on how to monitor your system performance regularly.

### When Should You Read this Chapter?

Read this chapter when you want to monitor, analyze, and interpret the response time of the SAP system or of individual programs and transactions. If you want to technically monitor and optimize the performance of the SAP system, you can read Chapter 3, "Workload Analysis", either before or after Chapter 2, "Monitoring Hardware, Database and SAP Basis". If you want to monitor and optimize the

performance of programs and transactions, you should read this chapter and then read Chapter 4, "Identifying Performance Problems in ABAP and Java Programs". Understanding workload analysis is a precondition for successful performance optimization.

## **3.1 The Workload Monitor**

This section describes the Workload Monitor. First we will deal with the functions and availability of the Workload Monitor, then we will describe how to work with the Monitor and, finally, we shall look at the technical settings.

### **3.1.1 Functions and Availability**

The Workload Monitor (transaction code ST03) has formed a part of SAP software since the first release of SAP R/3 (in the Computer Center Management System CCMS). With SAP Basis 4.6C the Workload Monitor is now available with a completely reworked interface, in EnjoySAP format (transaction code ST03N). In the latest version of the (central) Workload Monitor it is possible to display statistical data from all SAP components in complex system landscapes, even from "non-R/3" systems (transaction code ST03G). The statistical data from remote components can be accessed using RFC. The central Workload Monitor is available with SAP Basis 6.10. The system to be monitored must be at least SAP Basis 4.0 with corresponding support packages.

In the following section we shall refer to the mapping and menu paths in SAP Basis 4.6C and higher. Menu paths for older versions of Basis are also given.

### **3.1.2 Working With the Workload Monitor**

Statistics such as response times, memory use, and database accesses are collected and stored for all transaction steps for all SAP components. These statistics are organized in load profiles that can be displayed in the Workload Monitor. The Workload Monitor enables you to obtain a comprehensive overview of load distribution within an SAP component.

To access the initial screen of the Workload Monitor, use transaction code ST03N. The main screen of the Workload Monitor appears, **Load Analysis in System X**, as can be seen in Figure 3.1. The main screen is divided into three windows:

In the window on the upper left you will first of all find a button (in the top left-hand corner), with which you can choose a role. There are three roles:

► **Administrator**

This is the standard user mode. It offers fast access to the system load statistics of the current day and gives an overview of system load distribution. In addition, in this mode you can also display the functions related to the data collector.

► **Service Engineer**

This mode gives you the system load statistics for the current day and for the previous week as well as an overview of the history and distribution of the workload and a detailed analysis of system load. By default, the system displays all of the statistics for all application servers.

► **Expert**

This mode gives users all functions available in Transaction ST03N. You can display all available data on system load (daily, weekly and monthly data).

Furthermore, in the window in the upper left you can choose the SAP instance you wish to analyze or "Total", if you want to analyze the entire SAP system. You can also select the period of time to be included in the analysis.

You will find administration information in the top part of the window on the right. The **Instance** field displays the name of the SAP instance or "Total", if you are analyzing the entire SAP system. Using the data for **Period**, **First record** and **Last record** you can check to see if the collection and compression of data in the selected time period has been done accordingly.

Further down you will see statistical data on the performance of the SAP system and information on possible causes of performance problems. If you select the **Workload overview** analysis view in the window in the lower left, you will see a breakdown of statistics on response times and throughput, according to different task types. For the most part, the task types correspond to the *work process types* "Dialog", "Update", "Update2", "Background" and "Spool". The "Dialog" work process type is further subdivided into the task types "Dialog", "RFC", "AutoABAP" and "Buffer Sync".

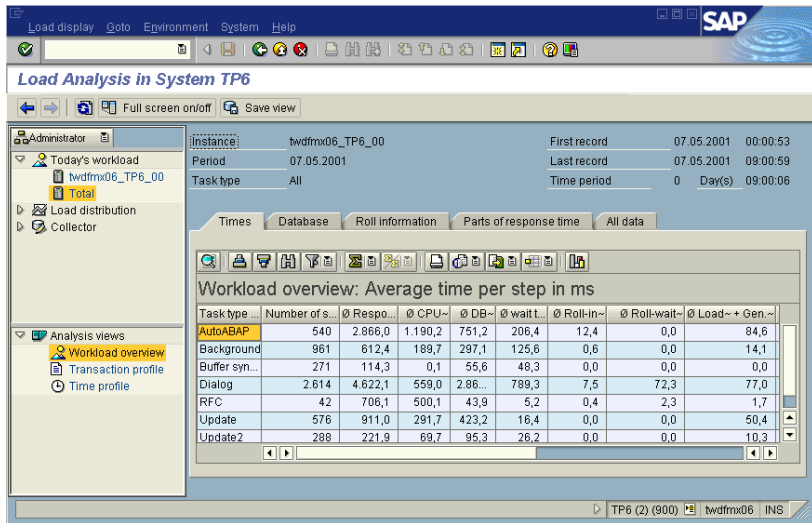


Figure 3.1 Main screen of the Workload Monitor

**Transaction step** The first important values are the number of *transaction steps* (**Number of steps** column) and the average response time (**Av. Response time**). In a dialog task, a transaction step corresponds to a screen change—that is, to a request that is executed for a user by the mySAP component. The term “dialog step” may be somewhat misleading, as dialog steps used in the Workload Monitor are performed not only in dialog tasks (immediate responses to input from online users), but also in background tasks, update tasks, and spool tasks. Processing an update request or a spool request is counted by the Workload Monitor as one dialog step. Similarly, a background program may involve one or more dialog steps. To avoid ambiguity, this book will use the term *transaction step* for the processing steps referred to in the Workload Monitor as dialog steps, to differentiate them from background dialog steps. The number of transaction steps per unit of time will be referred to as *system activity* or *throughput*.

**Response time** The *average response time for a transaction step* in a dialog task (**Av. Response time**) is seen by many SAP users as the criterion for the acceptable performance in an SAP component. Another generally accepted rule of thumb is that good performance in SAP R/3 is indicated by an average response time of 1 second or less. As we will see in this chapter, however, a broad generalization of this kind is not accurate when you consider the variety of different SAP components and the demands made on each.

In addition to the average response time, there are many other statistics, such as database time, CPU time, and so on, that enable you to understand performance problems and their possible causes. These statistics are explained in the next section.

Using tabs in the menu interface you can select other screens with information on database accesses etc. Using the **Save view** function you can save user-specific views. The next time you call up Transaction ST03N, the system automatically displays the view saved.

Load profiles, or analysis views, enable you to perform a detailed analysis of load distribution and response times. Apart from the **Workload overview** already mentioned, in the lower left window you can select other analysis views: Transaction profile and Time profile, which we will discuss in greater detail in later sections.

For older SAP Basis versions, enter transaction code ST03. The initial screen of the Workload Monitor appears, **Workload: Analysis of SAP System SID**. From this initial screen, you can access the main screen. First, choose the **Performance database** button. In the dialog boxes that appear you can select data for a specific server or for the entire system ("TOTAL") and a period to be analyzed. You are then shown the main screen of the Workload Monitor, **Performance: Workload summary of server**.

Older versions

If you are interested in the current statistics for the server you are connected to, from the initial screen of the Workload Monitor choose **This application server · Last minute load**. Then specify over how many minutes the response times should be given (e.g. the last 15 minutes). You are then brought to the main screen of the Workload Monitor, **Performance: Recent Workload for Server**.

The main screen of the Workload Monitor is divided into three sections. The first section, **Instance**, contains administrative information. The fields **SAP System**, **Server** and **Instance no** show the name of the SAP component and the name of the server selected (or "Total", if you are analyzing the entire SAP system). Using the data for **First record** and **Last record** you can check to see if the collection and compression of data in the selected time period has been done accordingly. The lower section, **Task types**, tells you which task type you have chosen (**Current** field), and the buttons that enable you to choose other task types. The central section of the screen (**Workload**) contains the statistical data.

From the main screen of the Workload Monitor you can go to the load profile by selecting **Goto • Profiles**. There, you can select profiles for the technical analysis, such as Task Type Profile, Time Profile, Computer Profile and Memory Profile and profiles for the application analysis, such as Transaction Profile, User Profile, Client Profile and Accounting Profile.

### 3.1.3 Technical Settings for the Workload Monitor

To ensure that individual statistical records, generated and recorded for each transaction step, are regularly collated in profiles, the `RSCOLL00` program must be scheduled to run every hour as a background job (generally under the name `SAP_COLLECTOR_FOR_PERFORMANCE`). You can display and modify the parameters affecting the creation of profiles in the Workload Monitor. In the "Expert" role, select

**Collector & Perf. Database • Parameter & Reorg • Collector & Reorg**

Under **Standard statistics** you can enter the retention periods for profiles—that is, the time before they are automatically deleted. Under **Time comparison data** you can enter the retention period for data displayed under the Workload Monitor menu option **Load history**. The **Cumulate server statistics to a systemwide total statistics** option determines whether or not a systemwide statistic should be generated. This option should always be selected. With the **Delete seq. statfile after ...** and **Max. number of records...** parameters you can specify when the individual statistical records are deleted and also the maximum number of records that are to be collated in each run of the `RSCOLL00` program.

Protocols are set for each run of the `RSCOLL00` program which you can use for troubleshooting. You can see the protocols in the Workload Monitor by selecting **Collector & Perf. Database • Perf. Monitor Collector • protocol** or **Collector & Perf. Database • Workload collector • protocol**. Explanations of the functions and settings of the data collector may be found in SAP Online Help and in the SAP Notes listed in the appendix.

## 3.2 Workload Analysis

To examine Workload Analysis more closely, in this section we will discuss the sequence of events in a transaction step and the times measured during it, with the help of Figures 3.2 and 3.3.

### 3.2.1 Course of a Transaction Step

Once an SAP user completes an entry, the presentation server sends the request to the dispatcher on the application server. The response time (**Av. response time**) is measured from the moment when the request from the presentation server reaches the dispatcher in the application server (step 1 in Figure 3.2). The response time ends when the request is processed and the last data is sent back to the presentation server.

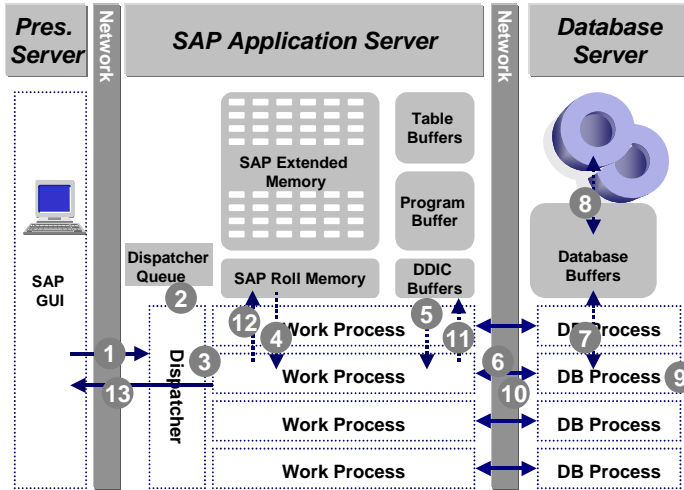


Figure 3.2 Course of a transaction step

When the dispatcher receives a processing request, it looks for a free SAP work process of the required type (dialog, update, and so on) and then sends the request to this work process, which begins the processing work. If all SAP work processes of the required type are busy when the request initially reaches the dispatcher, the request is placed in the dispatcher queue (2).

Dispatcher wait time

In the dispatcher queue, the request waits until a work process of the required type is free. As soon as a work process is free, the dispatcher sends the request to it (3). The time the request spends in the dispatcher queue is indicated as the **Av. wait time**. Note that there are many other kinds of wait time involved in the processing — for example, waiting for RFC calls, locks, CPU access, database access, and so on. To differentiate the wait time discussed here from others, it should be referred to as *Dispatcher wait time*.

- Roll-in, roll-out** An SAP transaction normally extends over several transaction steps (screen changes). During these steps, data such as variables, internal tables, and screen lists is built up and stored in the main memory of the application server. This data is known as *User context*. Different transaction steps are normally processed by different dialog work processes. For example, the first transaction step may be processed by work process number 3, the second transaction step by work process number 4, and so on. At the beginning of a transaction step, the user context is made available to the appropriate work process. This procedure is called *Roll-in* (4). The technical processes comprising a roll-in, such as copying data into the local memory of the work process, are described in detail in Chapter 8. The analogous process of *Roll-out* saves the current user-context data to the virtual memory at the conclusion of a transaction step (12). The duration of the roll-in is referred to as *Roll-in time* and the duration of the roll-out is known as the *Roll-out time*. The average roll times are indicated as **Time per roll-in** or **Time per roll-out** in the Workload Monitor in Release 3. To obtain the average roll times in Release 4, divide the values **Roll-in time** or **Roll-out time** by the number of "roll-ins" or "roll-outs" respectively. Please note that the roll-out time is *not* part of the response time of a transaction step. At roll-out, when the user context is copied from the local memory of the work process to the roll memory, the processed data has already been returned to the presentation server.
- Load time** All ABAP programs and screens that are required and are not yet available in the application server buffers must be loaded and possibly generated. The time it takes to do this is indicated as **Av. load+gen time**. Loading a program also entails accesses to database tables storing the ABAP programs—for example, the tables D010S and D010L.
- Database time** When data is read or changed in the database, the time required is known as *database time* and is indicated as **Av. DB request time**. Database time is measured from the moment of sending the database request to the database server and runs until the moment at which the data is returned to the application server (6–10). Because database time is measured by the application server, database time includes not only the time required by the database to produce the requested data, but also the time required for the network transfer of that data. Thus, a network problem between the database and the application server results in a greater database time.

Before accessing the database, the database interface of the work process checks whether the required data is already in the SAP buffers. If the data is already in the buffers, the buffers are accessed directly because using buffers is up to 100 times faster than a database access (5, 11). Buffer accesses do not contribute to database time.

Since release 4.6C it is possible to gather statistics separately on DB procedure calls. DB procedure calls are particularly relevant in APO systems, since communication between APO instances and liveCache are done via DB procedures (which can also be called *COM routines*). For analysis purposes, a new subrecord type, the *DB procedure subrecord*, has been introduced. This new subrecord type contains the name of a DB procedure and the name of the logical DB connection as a key field, along with the number of calls and the total execution time as the data section. By default, such subrecords are currently not written. To activate writing you must set the profile parameter **stat/dbprocrec**. This is also possible in a running instance in the Workload Monitor in expert mode. In the Workload Monitor the time needed to execute DB procedures is identified as *DB procedure time*. The database time in the Workload Monitor and in the single record statistics contains only times for calls to the "actual" database.

**DB procedure time**

*Roll wait time* occurs in connection with Remote Function Calls (RFC), in other words, when there is communication between software components or, from SAP Basis 4.6 on, when there is communication with the presentation level.

**Roll wait time**

Up until SAP Basis 4.5 the duration of communication between the presentation and application servers (network transfers and the creation of images on the presentation server) was not included in the workload analysis data. Starting with SAP Basis 4.6, these times are included in the response time, for the main part at least, as *GUI time*. Roll wait time and GUI time are explained in Chapters 6 and 7, "Interfaces", and "SAP GUI and Internet Connection".

**GUI time**

Enqueue time, indicated as **Av. enqueue time**, is the time during which a work process sets an enqueue request.

**Enqueue time**

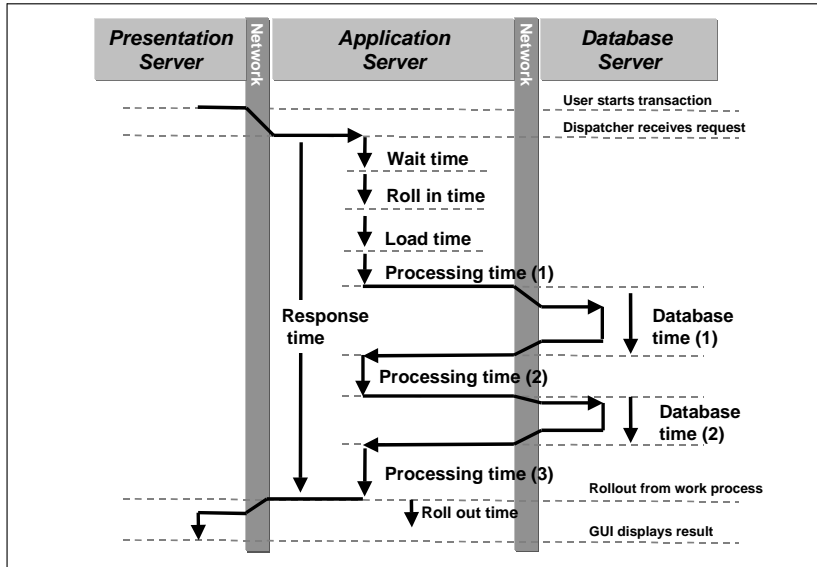
*Processing time* is the total response time minus the sum of all times mentioned above (except GUI time).

**Processing time**

All the statistics discussed above concern actions that form part of an SAP work process; that is to say, whenever the action in question runs, it is timed. With **Av. CPU time** on the other hand, at the end of a transaction

**CPU time**

step, the SAP work process asks the operating system how much CPU time has expired during that step. CPU time is not determined by the operating system. CPU time is not an additive component of transaction response time (like the times mentioned above), but is consumed during load time, roll time, and processing time (see Figure 3.3).



**Figure 3.3** Response time and its components: dispatcher wait time, roll-in time, roll wait time, load time, database time, processing time and CPU time

### 3.2.2 Interpreting Response Times

To analyze response times for dialog processing, use the guideline values in Table 3.1.

Time	Guideline value	Problem indicated	See chapter
Dispatcher wait time (»Wait time«)	< 10% of response time; < 50 ms	General performance problem with many possible causes	
Load time (Load+gen time)	< 50 ms	Program buffer too small or CPU bottleneck	2

**Table 3.1** Guideline values for analyzing the average response times for task type dialog. In the task type update the value can be around 50% higher. The "Problem indicated" column specifies what problem may arise if the given guideline values are significantly exceeded.

Time	Guideline value	Problem indicated	See chapter
Roll-in time, Roll-out time (in Release 4.0: "Roll-in time" / "Roll-Ins"; in Release 3.0/3.1: "Time per roll-in" etc.	< 20 ms	SAP roll buffer, SAP extended memory too small or CPU bottleneck	2, 8
Roll wait time	< 200 ms	Problem with frontend communication (together with higher GUI time) or with communication with external component	4, 6, 7
GUI time	< 200 ms	Problem with frontend communication (together with higher roll wait time)	4, 6, 7
Enqueue time	< 5 ms	Problem with enqueue, network problem	4
Processing time CPU time	Processing time < 2 xCPU time	CPU bottleneck or communication problem	5
Database time ("DB request time")	< 40% of (response time minus dispatcher wait time); Guideline value: 200–600 ms	Database problem, network problem or CPU bottleneck	3, 4, 11
Time per DB request	< 5 ms	database problem	3, 4, 11
Direct reads	< 2 ms	database problem	3, 4, 11
Sequential reads	< 10 ms	database problem	3, 4, 11
Changes and commits	< 25 ms	database problem	3, 4, 11

**Table 3.1** Guideline values for analyzing the average response times for task type dialog (contd.)

If the values you observe in the Workload Monitor are *significantly* outside the guideline range indicated here, there may be a performance problem in the relevant area (e.g. on the database). Note that these values are based on standard situations and may differ in some SAP components.



**"Lost time"** In addition to comparing your statistics with the guideline values, you should perform the following analysis, which could be referred to as the "search for time lost". As mentioned above, there are two different sources of time statistics. All times, apart from CPU time, are measured from the perspective of the SAP work process, whereas CPU time is measured from the perspective of the operating system. The lost time analysis aims to check whether the two statistics can be brought together. To do so, from the total average response time, subtract all times for which the SAP work process does *not* require any CPU time—namely, dispatcher wait time, database time, enqueue time and roll wait time. (As of Release 4, to find the average roll wait time from the Workload Monitor, divide the "Roll wait time" by the "Dialog steps".)

For the most part, programs are processed during processing time and as a result, CPU capacity is normally "consumed" during this time. As a result, processing time and CPU time should be more or less the same. As a rule of thumb, the difference between processing time and CPU time should not be more than 100%. Greater "lost times" indicate performance problems.

What are the possible causes for a significant difference between processing time and CPU time?

- ▶ The first possible cause is a CPU bottleneck. This would mean there is not enough CPU capacity available for the SAP work processes, which must therefore wait until CPU becomes available. In this case, processing time is being measured in the work process while no CPU time is used, and this processing time is considerably larger than CPU time.
- ▶ Another reason for a difference between processing time and CPU time concerns wait times in the SAP work process. Whenever the SAP work process has a status "stopped", processing time is measured without CPU time being used. This type of wait situation can be identified in the Work Process Overview.

### 3.2.3 Activity, Throughput and Load

Activity,  
throughput

The concepts of system activity, throughput and load can best be explained using the **Workload overview** analysis view (see Figure 3.1): The number of transaction steps can be seen in the second column (**Number of steps**). The number of transaction steps per unit of time can be defined as *system activity* or *throughput*. In our example we can see

the highest level of activity (2,614 transaction steps) corresponds to dialog processing, that is to say that the user has performed 2,614 screen changes in dialog processing mode during the period of time in question.

If two users have each executed 100 transaction steps in a given time period, they have created equal activity. This does not mean, however, that they have both produced the same *load* on the system. If, for example, the first user has entered 100 financial documents and has executed 100 transaction steps with an average response time of 500 ms, this user has occupied the system for 50 seconds. If a second user has created auditing reports and performs 100 transaction steps with of an average response time of 5 seconds, this user has occupied the system for 500 seconds. Evidently the second user has created a system *load* that is 10 times greater, with the same amount of *activity*. As can be seen from this example, the product of the number of transaction steps and the average response time is a way of measuring the load generated. (To be more exact, subtract dispatcher wait time and roll wait time from the total response time, because a request does not create system load while it waits in the dispatcher queue or while it waits for an RFC to be carried out.) Similarly, the database load created by the different task types can be determined using the total database time (transaction steps by average database time). CPU load on the application server can also be measured in this way. The distribution of times (database time, CPU time and so on) thus reflects the distribution of load on the system better than just the number of transaction steps.

**Load**

The simplest and most graphic measure for the size of an SAP component is the number of users. Unfortunately, the "number of users" is also an imprecise expression, and its meaning varies according to context. It can mean, for example, the number of licenses or the number of user master records, to mention just two possible meanings. For components that are mainly characterized by background or interface load, the number of users is no longer relevant as an indication of size.

**Active users**

To avoid confusion, this book distinguishes three types of users, as follows:

► **Occasional user**

On average a user of this type performs fewer than 400 transaction steps (screen changes) per week. For a 40-hour work week this corresponds to an average of one transaction step every 6 minutes. This is typically a user who only uses the SAP component now and again.

► **Transactional user**

On average a user of this type performs up to 4,800 transaction steps per week. This corresponds to less than one transaction step every 30 seconds. These are users who use the SAP component regularly and continuously.

► **Data entry/telesales/high volume "power user"**

Power users perform more than 4,800 transaction steps per week. They use the SAP component regularly and at a high volume.

In this book, *active users* are users who correspond to either the "transactional user" or the "power user" categories.

The user profile gives information about the activities of users. You can call the user profile in the Workload Monitor under **Load distribution · Users per instance**.

**Older versions** For older versions of SAP Basis call the user profile from the main screen of the Workload Monitor by choosing: **Goto · Profiles · User profile**. The profile provides information on the users logged on during a particular period of time and their activities.

### **3.3 Performing Workload Analysis**

In general, the first input for performance analysis comes in the form of observations made by the users. The Workload Monitor helps you to verify the subjective comments of users and narrow down the causes of performance problems. We can distinguish between two types of problem:

► **General performance problems**

A *general performance problem* results in poor response times and unsatisfactory throughput in *all transactions*. A problem of this type can have a negative impact on business processes and lead to financial loss.

► **Specific performance problems**

If the throughput or response time of *individual transactions* is unsatisfactory, then we can speak of *specific performance problems*. Specific performance problems can have a negative impact on business processes if the transaction in question is one of the key transactions in the business process (such as goods issue).

With the help of the seven questions included in the following sections, you can further limit performance problems. Guideline values and examples are provided to help you to answer these questions. You should, however, bear in mind that a simple yes or no answer is not always possible.

### 3.3.1 Analyzing General Performance Problems

#### Is There a General Performance Problem?

The users can usually identify a general performance problem. You can use the Workload Monitor to verify the users' observations and check whether response times affecting all transactions are high. The following criteria, which apply to dialog tasks, may help you to decide:

- ▶ Dispatcher wait time  $\gg 50$  ms: A significantly large dispatcher wait time always affects all transactions. It implies that programs are too slow and are blocking work processes for lengthy periods — or that too few work processes have been configured.
- ▶ Database time  $\gg 40\%$  (response time—dispatcher wait time) and database time  $> 400$  ms: A high database time slows performance for all transactions.
- ▶ Processing time  $> 2 \times$  CPU time: High processing time slows performance for all transactions. This can be caused by a CPU bottleneck or a problem with communication between systems.
- ▶ Average response time  $>$  system-specific guideline value: Average response time for a dialog task is seen by many SAP users as the decisive criterion for acceptable performance in an SAP component. A guideline value must be defined for each individual SAP component. A generally accepted rule of thumb is that good performance is indicated by an average response time of 1 second or less. A broad generalization of this kind is not always valid for all the different requirements of SAP components.

With SAP Basis 4.6 the response times displayed in the workload monitor include, for the first time, time data regarding communication between application and presentation levels (as part of GUI time, as time for network transfer and processing in the presentation server). This means that time elements are created that were not included in measurements in older versions. Due to the change in measuring techniques, as a rule of thumb the workload monitor for an SAP R/3 4.6 displays average response times of around 100 to 200 milliseconds higher than in the older versions, even though performance has not changed from the user's point of view. This should be borne in mind when negotiating service level agreements.



## Is the Performance Problem Temporary or Permanent?

After verifying that there is a general performance problem, try to find out how frequently the problem occurs. The following questions may help:

- ▶ Is the problem permanent or temporary?
- ▶ Does this problem occur at regular time intervals—for example at particular times of the day?
- ▶ Is it a non-recurring problem?
- ▶ What times (database time, CPU time or processing time) are high when the problem occurs?
- ▶ Does the problem occur following only specific system activities—for example, when background programs run on the system?

To examine these questions more closely, compare the workload statistics of recent days with each other.

In Workload Monitor expert mode, in the upper left window, select **Load history and distribution · Load history · Total**. Compare the performance values for several days to find out if the performance problem only occurs on certain days (see Figure 3.4).

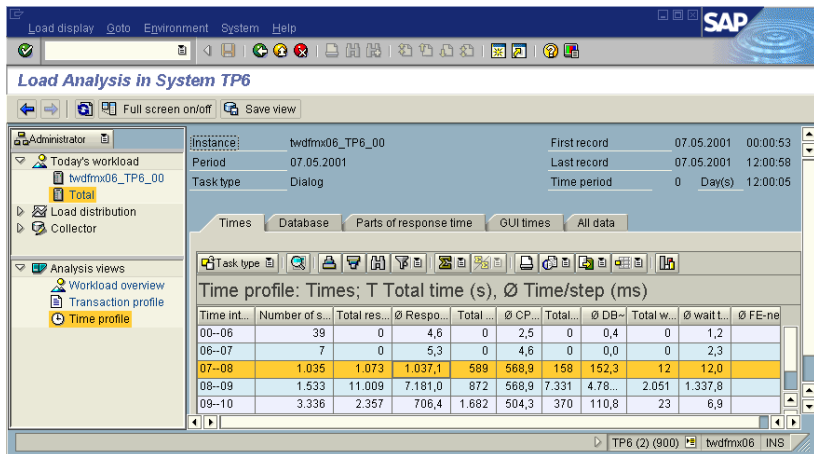


Figure 3.4 Time profile for dialog processing

Then generate the *day or time profile* by selecting the **Time profile** analysis view in the window in the lower left of the Workload Monitor. In a day or time profile the transaction steps and response times for all of the hours in one day are presented.

Using the time profile, you can analyze the daily loads on the system. If you find that the average response time increases dramatically only at particular periods of high load, you can infer that your system is overloaded at these times. If the average response times are also unsatisfactory at times of low system load, the performance problem is load-independent.

Example: You can diagnose a general performance problem using the workload analysis. By comparing the workload data for different days you can narrow the problem down to a particular period of time. In this period of time there are conspicuously high database times, especially *Changes and Commits*. On closer examination of the error log file in the database it emerges that an *Archiver stuck* occurred over night. (An *archiver stuck* occurs in an Oracle database if the directory for redo log files is full. This means that no further redo information can be written; the database and the SAP instances stop responding. Once the problem has been eliminated the database and SAP instances continue working without error.) The problem is resolved by the database administrator the next morning. In the middle of the day the *Archiver stuck* leads to higher database times for this day (especially for *Changes and Commits*). In an analysis on the Workload Monitor, this would suggest a poor database performance, although the performance is actually good, as becomes evident once this problem has been eliminated.



The time profiles enable you to determine whether excessive background processing during periods of peak system load has a negative impact on dialog processing. To create time profiles for dialog processing and background processing, using the **Task type** button, select the task types **Dialog** or **Background**. Use the **Total Response time (s)**, **Total CPU time total (s)** and **Total DB time total (s)** fields to determine at what time of the day the dialog or background load occurs. These profiles enable you to determine whether excessive use of background processing during periods of peak system load has a negative impact on dialog processing. You should try to ensure that the background processing load remains low during these peak periods, particularly if there are performance problems.

Dialog load vs.  
background load

You may also find it helpful to compare the time profile per day in the Workload Monitor with the time profile per day for CPU load and for paging (both indicated in the Operating System Monitor). This comparison enables you to determine whether deteriorating response times correlate with a large CPU load or a high paging rate. If so, a temporary hardware bottleneck is indicated (see the next question below).

**Older versions** For older versions of SAP Basis, in the initial screen of Workload Monitor select the **Performance database** button.

Then select **TOTAL, Previous Days** and one of the preceding days. Repeat this procedure for several days to find out if the performance problem only occurs on certain days.

Then create the *day or time profile*, using the following menu path in the main screen of the Workload Monitor:

**Goto • Profiles • Time profile**

**SAP EarlyWatch Alert** In the SAP EarlyWatch Alert report, in the section "DB Load Profile" you can get a weekly overview of the dialog, background and RFC load for each day. The tables in the SAP EarlyWatch Alert report are a compressed version of the daily profile.

**Is There a Hardware Bottleneck?**

A CPU bottleneck or main memory bottleneck can be detected as follows:

1. Find out if the hourly averages for the CPU load or paging rate are large. As a rule of thumb the risk of a hardware bottleneck is regarded as high when the hourly average of free CPU capacity (indicated as "CPU idle") is less than 20%, or the paging rate per hour exceeds 20% of the physical main memory. See also the section "Analyzing a Hardware Bottleneck (CPU and Main Memory)" on page 73 in Chapter 2.
2. In a second step, check whether the large CPU load or high paging rate really does negatively affect SAP component response times.
  - ▶ To check whether there is a hardware bottleneck on an application server, look at the processing time. If the processing time is much greater than double the CPU time, this indicates that the work processes are waiting for CPU. (However, an increased processing time may have other causes. See the section "Workload Analysis" on page 128.) A further indication of a hardware bottleneck on the application server is given by increased load times, roll-in times, and dispatcher wait time.
  - ▶ To check whether there is a hardware bottleneck on the database server, establish whether the database time is too large. Compare, for example, the average database times in the daily time profile at times of high and low load.

3. To check whether there is a main memory bottleneck, compare whether the virtually allocated memory is significantly greater than the physically available main memory. As long as the virtually allocated memory is smaller than 1.5 times the physical main memory, there is normally no risk of a main memory bottleneck. (See the section "Displaying the Allocated Memory" on page 105 in Chapter 2, for more on this subject.)

Only if all three of these checks (the first two apply to CPU and memory, the final one only to memory) indicate a hardware bottleneck, can you be fairly certain that there is, in fact, a hardware bottleneck.

The three possible causes of a hardware bottleneck are as follows:

► **Poor load distribution**

The load is not optimally distributed across the servers. There may be servers with free CPU or main memory capacity. Alternatively, load distribution may become nonoptimal at certain times of the day, for example, when several background processes are run in parallel during periods of peak system load. You should be able to reschedule these programs to run at times of low system load.

► **Individual processes causing a high CPU load:**

Individual processes with a high CPU load may be running at times of high system load. Such processes may include database processes (with expensive SQL statements), SAP work processes (with programs running as background jobs), or processes external to SAP. To improve performance, you may be able to tune, reschedule, or (in the case of external processes) cancel these processes.

► **Insufficient hardware capacity**

If the two previously mentioned causes of a hardware bottleneck do not apply, the hardware capacity may be too small for your system load.

If you have correctly identified a hardware bottleneck, proceed as described in Chapter 2 in the section "Analyzing a Hardware Bottleneck (CPU and Main Memory)" on page 73.

### **Is There a General Database Performance Problem?**

A general database performance problem is indicated by increased database times. The following guideline values for dialog tasks in the Workload Monitor indicate a general database performance problem:

- Database time >> 40% (response time minus dispatcher wait time);  
and  
database time > 400 ms

- ▶ Direct reads >> 2 ms
- ▶ Seq. Reads >> 10 ms
- ▶ Changes and Commits >> 25 ms

A database performance problem has many possible causes. Proceed as described in Chapter 2 in the section "Monitoring the Database" on page 81.

### Is Load Distribution Optimal?

A performance problem caused by nonoptimal load distribution can be detected by comparing the CPU load and the paging rates for the various servers (in the Operating System Monitor). You should also compare the response times for the various application servers in the Workload Monitor.

To display the *server profile* from the initial screen of the Workload Monitor, use the following menu path:

**Goto · Performance database · Analyze all servers · Compare all servers**

You can then enter the desired period of analysis with the menu option **Edit · Choose period type** and the buttons **Period+** and **Period-**.

The server profile shows the transaction steps and related response times for each server. If there are several SAP instances on one application server, the statistics indicated for the server are the totals for all instances on that server. To obtain details of the task types on individual servers, double-click a row in the list of servers.

#### Dispatcher wait time

In the server profile, check the load distribution across your servers. For example, if the *dispatcher wait time* occurs only on one server or on a small number of your servers, this implies either that too many users are working on these servers or that too few work processes are configured on these servers.

#### CPU time

Total CPU time (indicated as **CPU time total**) on all application servers should be roughly equal if all servers have the same CPU capacity. If you have servers with different CPU capacities, CPU time should differ proportionately.

One cause of a poor load distribution may be a nonoptimal configuration of log-on groups or work processes. To optimize load distribution, see Chapter 5, "Workload Distribution".

If the average *database times* (indicated as **DB time avg.**) for the various servers differ greatly, this may indicate a network problem. You can assume that application servers are configured with the same work processes and that users on the various application servers are, on average, using the same transactions. Thus, there is no obvious reason, apart from a network problem, as to why the database should serve one application server more slowly than another application server. This argument applies only to servers that are configured with the same work processes. For background servers, update servers or servers mainly used for reporting, the average database time will be greater than that for dialog servers.

Database time

For older versions of SAP Basis, to display the *server profile* from the initial screen of the Workload Monitor, use the following menu path:

Older versions

**Goto · Performance database · Analyze all servers · Compare all servers**

You can then enter the desired period of analysis with the menu option **Edit · Choose period type** and the buttons **Period+** and **Period-**.

### **Is There a Performance Problem Caused by SAP Memory Management?**

Performance problems caused by SAP memory management (see the section "Monitoring the Database" on page 81 in Chapter 2) may be the result of SAP buffers or SAP extended memory being too small. These problems would be seen in the Workload Monitor as follows:

- ▶ If the program buffer, CUA buffer or screen buffer are too small, there is an increase in average load time (average load time >> 50 ms).
- ▶ If the extended memory or the roll buffer are full the roll-in or roll-out times may increase (average roll-in or roll-out times >> 20 ms).

These guideline times apply to dialog tasks.

You should also monitor the *memory profile*. Up to SAP Basis 4.6, in the old Workload Monitor (transaction code ST03) this could be found under:

Memory profile

**Goto · Profiles · Memory profile**

The memory profile shows memory usage per program. Utilization of extended memory and heap memory (**Priv. mem.**) are indicated. The monitor also shows how often work processes entered PRIV mode (in the column **Workproc. Reservations**) and how often a work process was restarted after its use of heap memory exceeded the value of the

parameter **abap/heaplimit** (indicated in the column **Workproc. Restarts**). If you notice higher roll or load times, proceed with the analysis in Chapter 2 in the section "Analyzing SAP Memory Management" on page 101.

With SAP Basis 6.10 the main memory profile is integrated in the new Workload Monitor (transaction code ST03N).

### 3.3.2 Analyzing Specific Performance Problems

The Workload Monitor is an analysis tool used for both technical analysis and application analysis.

#### Is There a Performance Problem With a Transaction?

The *Transaction profile* is of primary importance for application analysis. To change over to the transaction profile, in the lower left window of the Workload Monitor, under **Analysis views**, select **Transaction profile**.

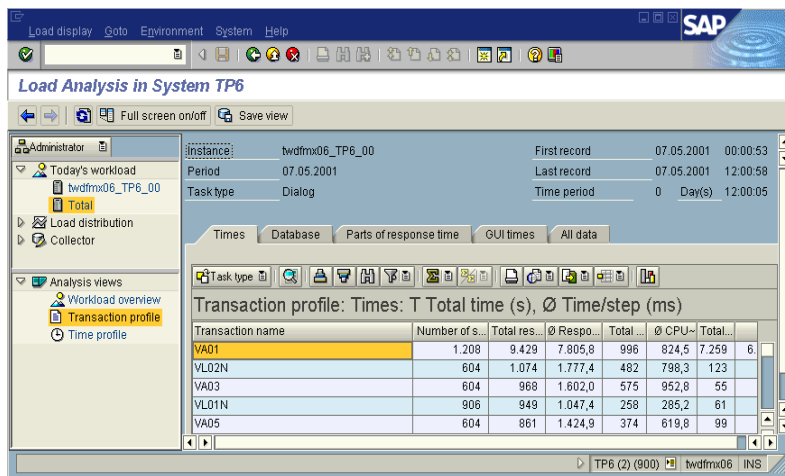


Figure 3.5 Transaction profile

The transaction profile contains a list of all transactions (or programs) started in the selected period. The number of transaction steps for each transaction is recorded (**Number of steps**), which is a measure of the activity of a transaction. Other columns in the transaction profile show the total response times and the average response times, as well as the proportion of CPU time, dispatcher wait time and database time. Using the tab pages in the menu interface you can select other screens with information about database accesses and so on.

From the number of transaction steps (**Number of steps**) you can estimate how frequently the transaction was executed if you know how many transaction steps (screen changes) each regular user requires on average per procedure. For example, if a regular user requires around 5 transaction steps to create a sales order (transaction code VA01) and the transaction profile shows 100,000 transaction steps for the selected time period, you can calculate that around 20,000 sales orders were created. If you wish to see which transaction had the most activity, sort by the column **Number of steps**.

System activity

The column **Total response time** provides a measure for the entire load on the system. (See the section "Activity, Throughput and Load" on page 134.) To find out which transactions produced the most load on the system, successively sort the list by the columns **Total response time**, **Total CPU time** and/or **Total DB time**. After each sort, the programs at the top of the list are likely candidates for performance optimization.

Load

In Figure 3.5, Transaction VA01 is executed twice as often as Transaction VA03 (1,208 steps compared to 604); however, due to the higher average response time, Transaction VA01 generates 10 times as much load as Transaction VA03 (9,429 seconds compared with 968 seconds).



For users, the *average response time* of the transactions they use is an important index of performance. Monitor and create guideline values for the average response times of core transactions—that is, transactions whose performance is central to business operations.

Average response time

In general, when analyzing the transaction profile, consider the following questions:

- ▶ Sort the transaction profile according to **Total DB time**. Which transactions cause the greatest database load?
- ▶ Sort the transaction profile according to **Total CPU time**. Which transactions cause the greatest CPU load?
- ▶ Are there transactions for which the proportion of database time or CPU time is significantly higher than 60% of the total response time? Analyze these transactions using an SQL trace or ABAP trace. The procedure for analyzing individual programs and transactions is described in Chapter 4, "Identifying Performance Problems in ABAP and Java Programs".
- ▶ Are there any customer-developed programs or transactions that produce a large load?

Monitor and save a copy of the transaction profile at regular intervals. This enables you to determine whether the response times of individual transactions grow continuously over time, or whether there is a sudden worsening of response times following a program modification. By recognizing such trends in the transaction profile *early*, you can initiate a detailed program analysis *before* a program causes a bottleneck for an entire process chain, or worse, reduces the performance of the entire SAP system through large CPU or database loads.

Table 3.2 provides explanations and guideline response times for common system transactions.

Transaction/ Program	Description/Comment	Acceptable response time
MainMenu	Actions in the menu. "MainMenu" frequently appears near the top of the list if sorted according to "Dialog Steps".	< 100 ms
Login_Pw/ Logoff	logon or logoff screen	
AutoABAP	The Auto ABAP runs periodically in the background and executes actions such as those required by Alert Monitor	< 1,000 ms
Buf.Sync	buffer synchronization	< 1,000 ms
Rep_Edit	Actions in the ABAP editor	
(B)SCHDL	The batch scheduler runs periodically and checks whether background programs are due to be started.	
RSCOLL00	The performance collector runs periodically and collects data on performance. If you sort the transaction profile according to "Response time total", this program is often near the top of the list. However, the columns <b>CPU time total</b> and <b>DB time total</b> , indicate that this program produces little CPU or database time. Most of the response time for this program occurs when this program is waiting in work processes to receive performance data.	
RSM13000	The update program is used to summarize all update module statistics that cannot be ascribed to a transaction.	< 3,000 ms

**Table 3.2** System programs in the transaction profile. In general these programs can be ignored during performance analysis.

For older versions of SAP Basis, transaction profiles are generated as follows: Older versions

1. Start the Workload Monitor (Transaction ST03), select **Performance database** and then select **Total** (or a server) and a period of time.
2. You are then brought to the main screen of the Workload Monitor. Using the buttons **Total**, **Dialog** and so on, located in the lower part of the screen, select the task type you require.
3. Then use the **Transaction profile** button to create the transaction profile.

### 3.4 Application Monitor

Another important instrument for workload analysis is the Application Monitor (transaction code ST07), which you can use to create a load profile for each SAP module. To call this monitor, select:

**Tools · Administration · Monitor · Performance · Workload · Application · Application monitor**

All screens of the Application Monitor show performance-relevant data according to SAP application module. The Monitor takes advantage of the fact that each transaction, each program and each table can be found in the SAP component hierarchy. For example, you can see statistics on the transaction "Create Sales Order" (transaction code VA01) under the SAP hierarchy branch **Sales & Distribution · Sales · Create sales order**. (To see an outline of the full SAP component hierarchy structure, use Transaction HIER.) Many screens in the Application Monitor show data that is also displayed in other monitors, such as the Workload Monitor, the SAP Storage Configuration Monitor (transaction code ST02), the Table Call Statistics (transaction code ST10), and so on. The advantage of using the Application Monitor is that the data is grouped according to the SAP application component hierarchy; from an initial overview, you can descend to an increasingly detailed view of the particular modules, sub-modules and transactions that are consuming the most system resources.

#### 3.4.1 User Profile

The initial screen of the Application Monitor shows the current number of users for the different SAP modules. The user profile for each listed application consists of the numbers of logged on users, active users, and

the users currently waiting for a request to be processed. To descend to a lower level of the SAP application component hierarchy and thus see the user profile for a more specific area, successively double-click the appropriate modules.



For example, double clicking on the **Sales & Distribution** module displays the distribution of users in the sub-modules "sales", "shipping", "billing" and "basic functions". A subsequent double click on **Sales** shows the user profile for transactions belonging to Sales, such as "VA01", "VA02".

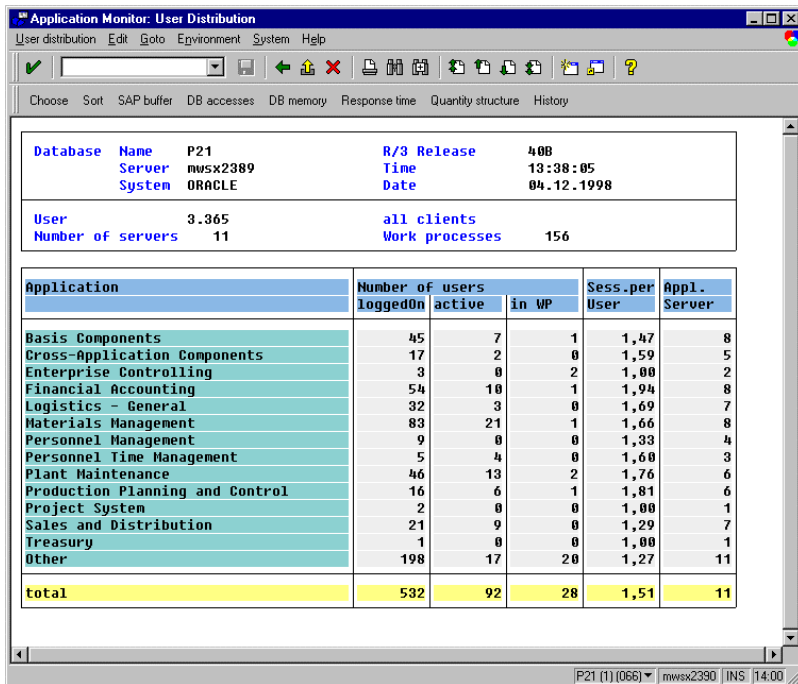


Figure 3.6 The user profile in the Application Monitor

Looking at the user profile in the Application Monitor is a convenient way of finding out the number of logged on and active users. An "active" user is one who has performed a transaction step in the last 30 seconds. You can change this time period by selecting the menu option **User distribution · Change active time**. To analyze the user distribution on the application servers, select **User distribution · Choose app. server**. The user profile in the Application Monitor is especially useful when you wish to check whether logging on with logon groups is functioning correctly, so that users are logged on to application servers independently of one another.

### 3.4.2 Load per SAP Application Module

1. From the Application Monitor initial screen choose **Response time** to display the load profile for each SAP application module.
2. In the dialog boxes that appear, specify the relevant server and time period—similar to the Workload Monitor.

The data on this screen matches that in the transaction profile and includes transaction steps, response times per transaction step and CPU times, wait times and database times. Use the **Total?↔Avg** button to toggle between the average response times per transaction and total response time for all transaction steps. All data on this screen is grouped according to SAP application module. By successively double-clicking at the appropriate module, you can descend to statistics on more specific component areas.

You can use the Application Monitor in the same way as you used the transaction profile in the Workload Monitor. The advantage of using the Application Monitor is that the data is grouped according to the SAP application component hierarchy; from an initial overview, you can descend to an increasingly detailed view of the particular modules and sub-modules that are consuming the most system resources.

In the Application Monitor, you can create graphs and diagrams illustrating load distribution. Use the button **Total?↔Avg** to show total response times (that is to say, "Resp. time total (s)", "CPU time total (s)" and so on) and then choose the **Graphic** button. The resulting pie charts illustrate the distribution of transaction steps and the response times according to SAP application module. The distribution of transaction steps on the module corresponds to the activity distribution, the distribution of total response time corresponds to the load distribution among all SAP components, distribution of database time corresponds to load distribution on the database, and distribution of CPU time corresponds to load distribution on the application servers.

In the SAP EarlyWatch Alert report, in the section "Workload by Application Module", you can get a weekly overview of the workload distribution per SAP component. The tables in the SAP EarlyWatch Alert report present the data from the Application Monitor in a condensed version.

SAP EarlyWatch  
Alert

### 3.4.3 SAP Buffers

From the initial screen of the Application Monitor, use the **SAP buffer** button to access an application of the SAP buffer, differentiated according

to SAP modules (program buffer, CUA buffer, screen buffer and table buffer for generic buffering and single record buffering). To access statistics for more specific areas within an SAP application module (following the SAP component hierarchy), successively double-click on the appropriate rows in the Application column, drilling down to individual programs and tables.

### 3.5 Cross-Component Workload Analysis

In a classic R/3 system, a transactional step usually consists of an action in a system. In a more complex system environment, a single transactional step can also involve actions in several systems. Examples of such transactional steps are:

- ▶ ITS applications such as SAP GUI for HTML, EWTs, etc. that involve actions on the ITS and in the ABAP runtime environment during a single transactional step.
- ▶ Applications such as CRM Internet Sales, Portal-iViews, Web dynpro applications, in which the front-end communication takes place via the SAP J2EE runtime environment and the back-end functionality is implemented via the ABAP runtime environment.
- ▶ Applications that involve two or more ABAP systems that are coupled via RFC, for instance sales transactions in R/3 that use the ATP function in APO for availability checks.

For all these cases, the NetWeaver technology enables you to carry out a cross-system workload analysis that links the performance statistics of the various components with each other.

**Availability** The central workload monitor is available in SAP Basis 6.20. The ABAP-based components to be monitored need at least a minimized version 4.0 of SAP Basis. In addition to ABAP-based components (R/3, BW Server, APO Server etc.) you can use the workload monitor to also analyze the SAP J2EE Engine, SAP ITS, and SAP Business Connector. Monitoring agents must be installed and must be running on the respective machines.

In order to use the central workload monitor you must set up an SAP component (with ABAP Basis 6.20 or later) as a central monitoring system. This system should be the same as the one you use for the central CCMS monitor. Typically, this is the Solution Manager. The components to be monitored must be made known to the central monitoring system in the CCMS System Component Repository (SCR) (see Section 1.2.4).

You can find more information on the CCMS SCR and the SAP monitoring agents and their enhancements in SAP Service Marketplace at <http://service.sap.com/systemmanagement>.

The central element of this SAP technology, also referred to as distributed statistics records (DSR), is the passport. The first component of a transactional step generates this passport (technically speaking, this is a GUID that is used to uniquely identify the transactional step) and transfers it to the next component that is part of the transactional step. This transfer is repeated for each component. For example, the first component that generates the passport could be the J2EE Engine (on which for instance an iView is being processed), while the component that follows is an R/3 system that provides the back-end logic for the iView.

Passport and distributed statistics records

For performance reasons, only the data of the passport is transferred in the communication flow during a transactional step. The actual statistical data is first saved locally by the respective components, then read out asynchronously via RFC or the monitoring agent (SAPCCMS), and then transferred to the monitoring system. Based on their passport, the statistical records that correspond to a transactional step can be identified and displayed in the central monitoring system.

The difference between the central workload monitor introduced here and the central single records statistics approach described in the following chapter (in the SAP Help these statistics are also referred to as *functional trace*) can be found in the way the data is displayed and the path through which the data reaches the display. While the global workload monitor displays aggregated data, central single-record statistics provides a more detailed view as it displays single statistics records and can thus trace actions that belong to a transactional step or a business process across system boundaries. In addition, single-record statistics displays traces of ABAP systems (SQL Trace and runtime analysis) as well as non-ABAP systems (DSR traces). The global workload monitor (ST03G) is thus an enhancement of the workload monitor (ST03N), while the functional trace (STATTRACE) is an enhancement of single-record statistics (STAD).

Central single records statistics and central workload monitor

## Working with the Central Workload Monitor

Call the initial screen of the workload monitor via Transaction ST03G: This brings you to the main screen of the workload monitor, "Global System Workload Analysis," as illustrated in Figure 3.7. The design of this

monitor is very similar to the design of the workload monitor for a single SAP system. The screen is divided into three window panes.

The top left pane contains the **Workload** node under which you can find the components for which load information is available. The load data is structured in the following hierarchy:

► **Component type**

Describes the type of components. For an ABAP-based component, "SAP R/3" is displayed; for the ITS, the short name "ITS;" for the SAP J2EE Engine, "SAPJ2ENode," and for the database to which the J2EE Engine is connected, "SAPJDBI.." For ABAP-based components, the SID is also displayed.

► **Component**

Describes an actual component such as the instance name of an ABAP-based component, or the instance name of an ITS.

► **Period**

You can view the workload data on a daily, weekly, or monthly basis. To do this, expand the relevant time unit and select the concrete period by double-clicking on it.

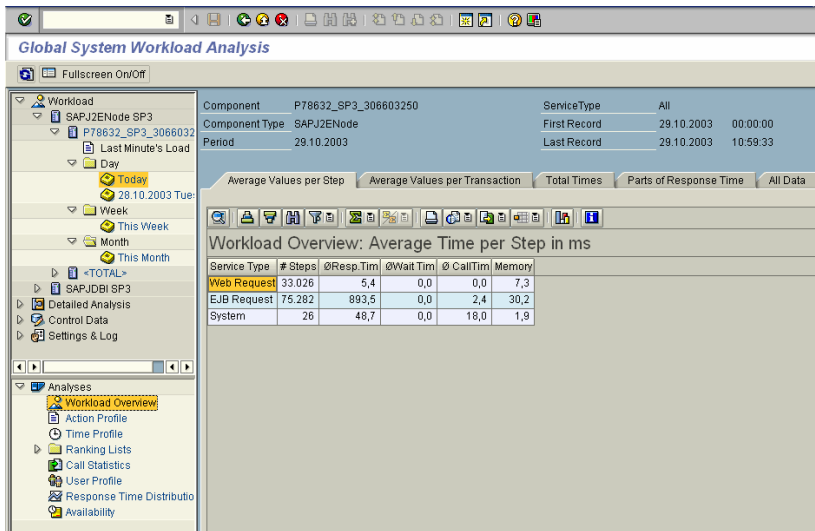


Figure 3.7 Main Screen of the Central Workload Monitor

**Analysis views** The lower left-hand pane contains a list of analysis views. These views are similar to the analysis views we already introduced in the context of the local workload monitor: Workload profile, time profile, transaction profile, and so on.

The right-hand pane that presents the analysis data also has the same structure as the one you already know from the local workload monitor: The upper part of the right-hand pane contains administration information, the instance name of the component, and the period for which statistical data is available.

The lower part of the right-hand pane displays the actual workload data, depending on the analysis view selected in the lower left-hand pane.

The lower part of the right-hand pane also displays the actual statistical records, depending on the analysis view selected in the lower left-hand pane. The statistical records that belong to a dialog step are displayed in a tree structure together with the most important performance information. This information involves the following data:

- ▶ Response time
- ▶ CPU time
- ▶ Call/roll-wait time
- ▶ Wait time of step

With this information, you can quickly find out in which component there were long wait times. A high CPU time means that the application on this component must be further analyzed. A high wait time in the component means that an overload situation exists in the component. Depending on the type of component (ABAP runtime environment, J2EE runtime environment or ITS), a component-based analysis must be carried out. We will describe this process in further detail in the following chapters. A high call/roll-wait time means that the performance problem cannot be found in this component but in a component that has been called by this one.

When you call the central workload monitor in your SAP system, you may find only statistical data for the component you are currently logged onto. In order to access the statistics data for other components from a specific system, you must make these components known centrally. To do this, proceed as follows:

1. Expand the **Setting & Log** subtree in the upper left-hand pane of the central workload monitor.
2. Select **System Selection**. The screen **Last saved dataset** is now displayed.
3. In order to display a list of systems, select the entry in the **Systems** dropdown menu.

4. If necessary, you can make changes to the list displayed.
5. To start the consistency analysis for a system list, click on the **Apply** button. The analysis is then performed for each activated entry. Prior to that the system carries out a consistency check of the destination to the monitoring system. If the consistency check fails for a specific entry, this entry is deactivated and a message for the application log is generated. The list for which you click on the **Apply** button thus contains the systems that are displayed by the global system-load monitor.

**Last minute's load** The central workload monitor determines the system-load data once per hour on the basis of the statistical data for the individual components and aggregates them into the corresponding load profiles.

However, under the **Workload** node in the upper left-hand pane, you can find the function **Last Minute's Load**. The workload monitor essentially determines the workload data once per hour on the basis of the statistical data for the individual components. This means that you cannot view any data that is less than one hour old, as it hasn't been written to the database yet. The function **Last Minute's Load**, however, enables you to request data that refers to a specific period within the last hour, such as for the last 15 minutes. Note, however, that this can take several minutes, depending on the size of the system.

**Technical settings for the workload monitor** For the global workload monitor, as for every system-monitoring tool, you must also find an optimal solution somewhere between the requirement for an exact monitoring of the system and the requirement for monitoring that doesn't affect overall system performance. The parameters that control the management of the statistics data are located under the **Control Data** and **Settings & Log** sub-trees. You can also find further information on this subject in the online help.

### 3.6 Summary

The Workload Monitor enables you to make detailed statements about the distribution of response times not only across different system components, such as the database, hardware, and SAP Basis components, but also across different transactions and programs. By performing a workload analysis, you can determine the system areas in which you require further analysis and tuning. Always remember to compare the results of your workload analysis with the observations of users. This helps you avoid jumping to the wrong conclusion if a superficial analysis

of the Workload Monitor indicates a performance problem where, in fact, there is no real problem. It also avoids the opposite situation of not noticing that the Workload Monitor is indicating a performance problem that is readily apparent to users.

Figure 3.8 summarizes a workload analysis for a general performance problem. You can find the corresponding detailed analyses for hardware, database, and SAP memory configuration in Chapter 2 in the sections "Monitoring Hardware", "Monitoring the Database", and "Analyzing SAP Memory Management".

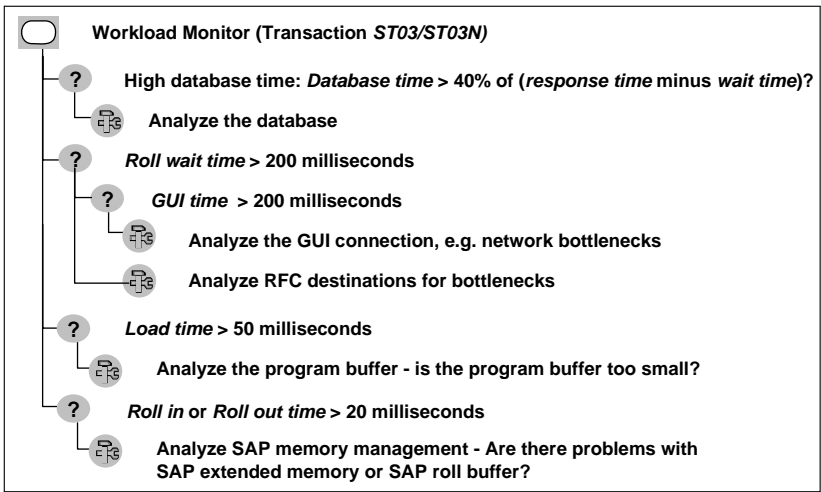


Figure 3.8 Summary of the most important steps of a workload analysis

## Important Terms in This Chapter

After studying this chapter you should be familiar with the following terms:

- ▶ Dispatcher wait time, load time, database time
- ▶ Roll-in time, Roll-out time
- ▶ Processing time and CPU time
- ▶ Activity, throughput and load

## Questions

1. Which of the following statements are correct?
  - ▶ CPU time is measured by the operating system of the application server.

- ▶ Database time is measured by the database system.
  - ▶ High network times for data transfers between the presentation server and the application server are reflected in an increased response time in the Workload Monitor.
  - ▶ High network times for data transfers between the application server and the database server are reflected in an increased response time in the Workload Monitor.
  - ▶ The roll-out time is not part of the response time because the roll-out of a user occurs only after the answer has been sent to the presentation server. Nevertheless, it is important for the performance of the SAP components to keep the roll-out time to a minimum, as during the roll-outs, the SAP work process remains occupied.
2. How is the term "load" defined in this book?
- ▶ "Load" is defined in this book as the amount of load on the CPU of a computer, expressed as a percentage. It can be monitored in the Operating System Monitor (**CPU utilization**).
  - ▶ In this book "load" means the sum of response times, that is to say that the term "total load" refers to the total response time (**Response time total**), CPU load refers to the total CPU time (**CPU time total**) and database load refers to total database time (**DB time total**).
  - ▶ The term "load" in this book refers to the number of transaction steps per unit of time.
3. The Workprocess Monitor displays increased dispatcher wait times ("Av. wait time" >> 50 ms). What does this tell you?
- ▶ There is a communication problem between the presentation servers and the dispatcher of the application server—for example, a network problem.
  - ▶ There is a general performance problem—for example, a database problem, a hardware bottleneck, or insufficient SAP extended memory; or there are too few SAP work processes. This statement does not provide enough information to pinpoint the exact problem.
  - ▶ An increased dispatcher wait time is normal for an SAP component. It protects the operating system from being overloaded, and can be ignored.

# Index

## Numerics

- 32-bit architecture 304
- 32-bit technology 288
- 46 463
- 64-bit architecture 305, 319
- 64-bit technology 288

## A

- A004 340
- ABAP 497
- ABAP coding 227
- ABAP debugger 180, 188
- ABAP Dictionary 497
- ABAP program 157, 400, 405
- ABAP program termination 313
- ABAP runtime analysis 173, 174, 188, 400
  - BSP-Anwendungen 177
- ABAP trace see ABAP runtime analysis 173
- ABAP Trace Summary 178
- abap/heap\_area\_dia 292, 296, 315, 317
- abap/heap\_area\_nondia 292, 296, 315, 317
- abap/heap\_area\_total 292, 303, 315
- abap/heaplmit 144, 292
- AcceleratedSAP 497
- ACID Principle 497
- activating buffering 329
- active users 135
- address space 288, 304, 497
- addressable memory 263
- administration for indexes 389
- administration for table access statistics 389
- administration tools 421
- Advanced Business Application Programming 497
- AGate 252, 258
- Agent Private Memory 84
- aggregate functions 401
- ALE 497
- Alert Monitor 51, 497
- allocated memory 105, 107, 454

- American National Standards Institute 497
- analyzing a hardware bottleneck 73
- ANSI 497
- application error 314
- application level 33
- Application Link Enabling 497
- application monitor 147
- application optimization 18
- application server 71, 254, 497
- application support layer 85
- application tuning 63
- ArchiveLink 497
- archiver stuck 98, 139
- archiving object 497
- array fetch 167
- ASAP 497
- asynchronous RFC 226, 239
- asynchronous update 202
- ATAB 348, 421
- ATP server 193, 194, 195, 365, 366, 367, 370, 510
- automatic alert messaging 58
- available address space 304
- available memory 302, 304
- average response time 145
- avoiding Select \* Statements 409

## B

- background load 139
- background processing 497
- background service 194, 195
- Backup 46
- BAPI 497
- Batch Input 498
- Benchmark 214, 498
- Bitmap Indexes 420
- blocks 83
- bottleneck analysis 123, 262
- browser 498
- BSP applications, runtime analysis 177
- buffer accessing 325
- buffer pool 84, 85
- buffer quality 83
- buffer settings 102, 103, 420

- buffer status 336
- buffer synchronization 327, 346
- Buffer trace 163
- buffered tables 337, 342
- buffering type 323
- buffers 36
- Business Application Programming Interface 497
- Business Connector 498
- Business Server Pages (BSP) 263
- button 498

**C**

- cache mechanisms 259
- caches 36
- CALL 290
- Catalog cache 448
- catalog cache 84
- CATT 498
- CBO 385, 387
- CCMS 26, 30, 393, 498
- CCMS System Component Repository (SCR) 150
- central control monitor 42
- central installation 194
- central monitoring 41, 123
- Central single record statistics 161
- central workload monitor 42
- Change and Transport Organizer 498
- changes 335, 343
- changes and commits 139
- Changing secondary indexes 394
- checkpoint 86
- client 498
- client/server architecture 33
- CO 498
- Common Programming Interface-Communication 498
- completed processes 114
- Component Overview 185
- Computer Aided Test Tool 498
- Computing Center Management System (CCMS) see CCMS
- condition tables 333, 348
- configuration performance parameters 471
- context switch 498

- continuous performance optimization 60
- Control Panel 498
- Controls 244
- Cost-Based Optimizer (CBO), see CBO
- coupling
  - hard 226
  - soft 226
- CPI-C 498
- CPU bottleneck 75
- CPU load 77, 141
- CPU resources 196
- CPU time 131, 134, 137, 142, 197
- CPU wait time 197
- creating secondary indexes 394
- CTO 498
- cursor 168
- Customizing 498
- customizing data 332
- Customizing Organizer 498

**D**

- D010S 421
- data archiving 499
- data buffer 83, 451, 455
- Data cache 448
- data cache 83, 456
- Data Control Language 499
- Data Definition Language 499
- Data Manipulation Language 499
- database 71, 499
- database analysis 71
- database buffer 82
- database consolidation 219
- database error log file 96
- Database Global Memory 84
- database heap 84
- database index 322
- database instance 71, 310, 499
- database level 35
- database load 403
- database lock 352, 354, 446, 499
- database monitors, buffers, and SQL execution plans 441
- database not responding 98
- database operations 167
- database optimizer 96, 382, 384, 399, 458, 499

- database performance monitor 82
- database performance problem 141
- database process monitor 88, 441
- database processor 86
- database server 71, 499
- database system 351
- database time 130, 137, 141, 143
- database view 413
- DB2 UDB for iSeries 449, 464
- DB2 UDB for Unix and Windows 84, 463
- DB2 UDB for zSeries 466
- DB2 Universal Database 449
- DB2 Universal Database for OS/390 451
- DB2 Universal Database for z/OS 451
- DBA 499
- DB-Analyzer 448
- DBIF\_RSQL\_NO\_MEMORY 318
- db/io\_buf\_size 167
- DBSTATC 392
- DCL 499
- DDL 499
- DDLOG 327, 346
- DDNTF 342, 421
- DDNTT 342, 421
- deactivated update service 113
- deadlock 356, 499
- detailed table analysis 344
- developer log 316
- Development Workbench 30
- DIAG protocol 499
- dialog load 139
- dialog service 194, 195
- dialog work process 499
- Direct read 460
- direct read 167, 459, 462, 464, 466, 468, 469
- disk space 263
- dispatcher 194, 499
- dispatcher queue 117, 129
- dispatcher wait time 129, 137, 142, 197
- dispatching update requests 204
- distributed installation 194
- Distributed statistics records, DSR 151
- DML 499
- dpmon 425
- dynamic statement cache 452

- dynamic user distribution 198, 199
- dynpro 499

## E

- EarlyWatch service 23
- Easy Web Transaction 253, 500
- EDI 499
- efficient SQL programming 400
- Electronic Data Interchange 499
- em/address\_space\_MB 297, 306
- em/blocksize\_KB 291
- em/initial\_size\_MB 104, 113, 291, 304, 311, 315
- em/max\_size\_MB 296
- enqueue service 193, 194
- enqueue table 357
- enqueue time 131
- Enqueue trace 163, 172
- enqueues 499, 510
- Enterprise IMG 499
- entity 499
- escalation procedure 48
- EWT 500
- exclusive database locks 95
- exclusive lockwait 94, 100, 118, 121, 354, 446
- execution plan 500, 382, 458
- expensive SQL statements 87, 91, 373
- expert monitors for performance analysis 42
- Explain plan, see execution plan.
- Explain, see execution plan.
- export/import buffer 367
- EXSORT\_NOT\_ENOUGH\_MEMORY 318
- extended global memory 298
- extended memory 315, 500
- extended memory area (EM) 314
- Extensible Markup Language 505
- external sessions 290
- external system 230

## F

- Failover solution 195
- FDDI 500
- fetch 335
- fetch operation 167
- fetch operations 409

- Fiber Distributed Data Interchange 500
- file system cache 78
- financial services 218
- firewall 258, 500
- fixed allocated memory (HEAP) 314
- FOR ALL ENTRIES 409, 412
- FOR ALL ENTRIES clauses 411
- fragmentation of indexes 420
- full buffering 324
- Full table scan 383, 395, 458
- Functional trace 151, 161
- fundamentals of RFC 225

## G

- general performance problems 136
- generating time 130
- generic buffering 324
- generic region 324
- generic table buffer (TABL) 321
- global cache hit ratio 452
- Global CCMS alert monitor 261
- global Operating System Monitor 262
- GoingLive check 23
- Graphical User Interface 500
- GUI 500
- GUI communication 245
- GUI time 131, 245, 247
- GUID 500

## H

- hard disk monitor 446
- hardware analysis 71
- hardware bottleneck 76, 140, 263
- hardware capacity 141
- hardware consolidation 221
- hardware landscape 194
- hardware partner 211
- hardware sizing 208, 209, 368
- heap memory 500
- high availability 195, 500
- hints 387
- hiperpool efficiency 452
- hit ratio 83
- Hot Package 500
- hot spots 446
- HTML 500
- HTTP 500

- Hypertext Markup Language, see HTML
- Hypertext Transport Protocol, see HTTP

## I

- I/O bottleneck 79, 93
- I/O problems 78
- IAC 500
- IDES 500
- IDoc 32, 500
- IDoc type 501
- IMG 501
- Implementation Guide 501
- inbound workload 234
- Index Range Scan 383
- Index range scan 461
- Index Unique Scan 382
- Informix 453, 468
- Informix data buffer 453
- initial sizing 210, 212
- instance 501
- Inter Process Communication 501
- interaction model 244
- interfaces 225
- Internal Document 500
- internal sessions 290
- International Demo and Education System, see IDES
- Internet 217
- Internet Application Component 253, 500
- Internet Communication Manager 263, 479
- Internet connection 243
- Internet portal 218
- Internet Transaction Server 200, 252, 257, 480, 511
- Intranet 501
- Introscope 189
- invalidation 103, 329
- IPC 501
- iSeries 297
- ITS administration tools 260
- ITS fundamentals 252
- ITS instance 258
- ITS work process 259, 262
- ITS, see Internet Transaction Server

iViews 33, 183

## J

J2EE Engine 511

JARM 183, 279

JARM statistics 184, 185, 189

JARM trace 188

Java 501

Java Application Request Measurement (JARM) 183, 502

Java program 157, 183

## K

KAPOL 337, 348, 421

## L

LAN 502

LAN check 80

load 135, 145

load distribution 509

load per SAP application module 149

load time 130

Local Area Network (LAN) 36, 502

local cache hit ratio 452

local memory 288, 502

local update 207

lock concept 351

Lock escalation 447

lock list 84

locking with quantity 195, 367, 368, 370

locks 351, 365, 502

logical analysis 322

logical read accesses 83

Logical Unit of Work 502

logon group 198, 258

logon group, see also dynamic user distribution

logon group, see also SMLG

long database response times 113

long-term performance problem 138

LUW 502

## M

main memory bottleneck 75, 76

main memory buffering 360

main memory requirement 309

main memory workload 75

master data 332

MaxDB 441, 442, 448

command monitor 444

resource monitor 445

measuring performance 244

memlimits 307, 308

memory allocation 292

memory area configuration 105

memory management 287, 297, 474, 509

memory profile 143

message service 193, 194

MiniApps 33

missing database indices 97

missing index 97, 390

mobile client 33

mode 502

mode list 311

Monitoring agent 161

monitoring buffer synchronization 346

monitoring database buffers 448

monitoring database locks 354

monitoring hardware 72

monitoring plan 38

monitoring SAP Basis 71

monitoring services 43

monitoring the database 81

monitoring tree 54

mySAP Business Suite 20

mySAP CRM 217

mySAP Enterprise Portal 33

mySAP solutions 29

mySAP Workplace 218

## N

nested loop join 414

Netscape Server API (Application Programming Interface) 502

NetWeaver LifeCycle Management 41

network 80, 169, 197

network problems 168

non-dialog work processes 295

nonoptimal workload distribution 76, 114, 142

NRIV 354, 359, 364

NRIV\_LOKAL 361

NSAPI 502

- number range 359
- number range buffering 322, 359, 363, 364, 510
- number range interval 359
- number range level 364
- number range object 359

## O

- Object Linking and Embedding, see OLE
- OLE 502
- OLAP 215, 502
- OLE 502
- OLTP 215, 502
- Online Analytical Processing 502
- Online Transaction Processing 502
- open 335
- operating system 287, 319, 502
- operating system limit 315
- Operating system monitor 73
- operating system monitor 73
- operating system restrictions 300
- operation mode 201, 502
- optimization plan 38
- optimizer 502
- Optimizer, see database optimizer
- Optimizing SQL Statements 373
- ORACLE 455, 460
- Oracle data buffer 455
- Oracle shared pool 455
- OS, see operating system
- outbound workload 234

## P

- package cache 84
- pages 83
- paging 502
- paging file 300, 319
- PAI 502
- parameter change 80
- parse 382
- partial table buffer (TABLP) 321
- Passport 161
- PBO 502
- pending period 328
- performance 502
- performance analysis checklists 425, 431
- performance analysis roadmaps 425

- performance aspects 294
- performance forum 70
- performance indicators 50
- performance management 29
- performance monitor 511
- Performance Trace 163
- Performance trace 163
- performance trace 247
- performance-limiting factors 260
- PHYS\_MEMSIZE 105, 296, 310
- physical main memory (RAM) 287, 300, 318
- physical memory 107
- physical read accesses 83
- planning hardware capacity 216
- pop-up window 502
- prepare 168
- presentation level 33
- primary index 380, 391
- PRIV mode 113, 312
- private mode 293
- proactive performance management 19
- procedure buffer 457
- procedure cache 457
- Process After Input 502
- Process Before Output 502
- process ID 111
- processing time 131, 134, 137
- profile parameters 316
- Program Global Area (PGA) 455
- PXA\_NO\_SHARED\_MEMORY 313

## Q

- Q-API 502
- Queue Application Programming Interface 502
- queued RFC (qRFC) 226
- Quick Sizer 211

## R

- R/3 502
- RAID 502
- RBO 385, 387
- RDBMS 503
- rdisp/atp\_server 195, 368
- rdisp/bufrefmode 328
- rdisp/bufreftime 328

- rdisp/enqname 195
- rdisp/max\_wprun\_time 199
- rdisp/mshost 195
- rdisp/PG\_MAXFS 299
- rdisp/PG\_SHM 299
- rdisp/ROLL\_MAXFS 291, 315
- rdisp/ROLL\_SHM 104, 291
- rdisp/vb\_dispatching 204
- rdisp/vbstart 203
- read access 365
- read random hit ratio 451
- read/write (I/O) problems 93
- recovery 46
- recursive call 456
- Redundant Array of Independent Disks 502
- Relational Database Management System 503
- Remote Function Call (RFC) 32, 131, 200, 225, 503
- reopen operation 167
- reorganization
  - of indexes 420
- reprepare ratio 453
- request 335, 342, 343
- required field 415
- RESB 63, 365, 420
- response time 132, 196
- RFC destination 231
- RFC time 230
- RFC trace 163, 171
- RFC, see Remote Function Call
- roll buffer 310
- roll memory 315, 503
- roll wait time 131, 230, 245
- roll-in 130, 290, 503
- roll-out 130, 290, 503
- roundtrip 245
- row cache 455
- Row ID 381
- rsdb/max\_blocking\_faktor 411
- rsdb/obj/buffersize 368
- rsdb/obj/max\_objects 368
- rstr/file 165
- rstr/max\_diskspace 165
- Rule-Based Optimizer (RBO), see RBO
- runtime analysis 280

## S

- SAP Advanced Planner and Optimizer (SAP APO) 31
- SAP Application Benchmark Performance Standard, see SAPS
- SAP application instance 71
- SAP architecture 29
- SAP Basis 29
- SAP buffer 101, 149, 310, 318
- SAP buffer parameters 472
- SAP buffering 404
- SAP Business Connector (SAP BC) 30, 498
- SAP Business Information Warehouse (SAP BW) 30
- SAP components 29
- SAP Customer Relationship Management (SAP CRM) 32
- SAP DB 462
- SAP Easy Access Menu 251
- SAP EG memory 298
- SAP enqueue 352, 357, 358
- SAP Enterprise Portal (SAP EP) 31
- SAP Exchange Infrastructure (SAP XI) 31
- SAP extended memory 103, 289, 291, 294, 310, 311, 318
- SAP GoingLive Check 212
- SAP GUI 243, 503
- SAP GUI 4.6 243
- SAP GUI for HTML 32, 255, 256, 257
- SAP GUI for Java environment 32, 33
- SAP GUI for Windows 32, 257
- SAP heap memory 103, 289, 291, 307, 315, 318
- SAP instance 71, 72, 220, 309
- SAP Internet Transaction Server (SAP ITS) 30, 252, 501
- SAP J2EE Engine 30
- SAP kernel 316
- SAP liveCache 30
- SAP Logical Unit of Work (SAP LUW) 353
- SAP memory area 299, 309
- SAP memory configuration monitor 101
- SAP memory management 101, 113, 143
- SAP NetWeaver 30

- SAP Notes on System Requirements 510
- SAP Online Help 507
- SAP Online Store 217
- SAP paging memory 298, 318
- SAP parameter changes 108
- SAP Quick Sizer 318
- SAP R/3 20
- SAP R/3 Enterprise 31
- SAP roll buffer 290
- SAP roll file 290
- SAP roll memory 103, 289, 290, 318
- SAP service 191, 194
- SAP Service Marketplace 318, 507, 509
- SAP Solution Manager 41, 49, 65, 66, 67
- SAP Standard Application Benchmarks, see benchmark
- SAP Strategic Enterprise Management (SAP SEM) 32
- SAP system 72, 351
- SAP System Identifier 504
- SAP system service 503
- SAP table buffering 321, 333
- SAP transaction 353
- SAP Web Application Server 33
- SAP work process 35, 109, 310, 318
- SAP work process overview 109
- SAPCCMSR 161
- saposcol 73
- SAProuter 503
- SAPS 211, 503
- savepoint 86
- scalability 37
- secondary index 379, 381, 389, 391
- semaphores 112
- Sequential read 460, 461, 463, 464
- sequential read 167, 458, 459, 462, 463, 464, 465, 466, 467, 468, 469
- server 71, 72, 503
- server consolidation 219, 221
- server profile 142
- Service Level Management (SLM) 24
- service level management (SLM) 44, 45, 503
- service level reporting 49
- session 289
- Session Manager 503
- SET UPDATE TASK LOCAL 207
- SET\_PARAMETER\_MEMORY\_OVERFLOW 314
- shared cursor cache 90, 443
- shared enqueues 367
- shared memory 288, 503
- shared pool 455
- shared SQL area 90, 92, 348, 374, 375, 378, 399, 443, 455
- shared SQL cache 90, 443
- SID 504
- Single activity trace 188
- single record buffering 321, 323
- single record statistics 157, 245
- Single statistics record 188
- Single transaction analysis 177
- sizing, see hardware sizing
- skeleton 452
- SMLG 199
- Snapshot Information 185
- Solution Management program 64
- solution manager 58
- sort heap 85
- specific performance problems 136, 144
- spool service 194, 195
- SQL 504
- SQL code 408
- SQL Server 456, 469
- SQL Server data cache 457
- SQL Server Procedure Cache 457
- SQL statement 88, 90, 379, 396, 400, 409, 443
- SQL Trace 188
- SQL trace 163, 165, 169, 348, 374, 377, 406
- Standard Application Benchmark 504
- Statistical record 504
- statistical record 157
- Statistics records
  - distributed 151
- stopped work processes 114
- STORAGE\_PARAMETERS\_WRONG\_SET 307, 313, 316, 317
- stored procedure 457
- Structured Query Language 504
- SUBMIT 290
- Support Package 504

- swap space 75, 287, 300, 301, 319, 504
- swaps 103, 329, 333
- synchronous RFC 226
- synchronous update 206
- system activity 126, 134, 145
- system availability, see high availability
- system configuration 509
- system consolidation 220
- System Global Area (SGA) 455
- system landscape 219, 504
- SYSTEM\_NO\_MORE\_PAGING 299
- SYSTEM\_NO\_ROLL 314
- systemwide work process overview 115

## T

- table access statistics 334, 387, 391
- table buffer 160
- table buffering 323
- table locks 357
- table size 343
- task type 127
- TCP/IP 504
- TCURR 345
- TDC 504
- technical analysis 322
- technical optimization 18
- technical tuning 63
- temporary performance problem 138
- temporary sequential objects 504
- TemSe 504
- think time 196
- throughput 126, 134
- Time flow hierarchy 178
- TMS 504
- TO 504
- Training Courses 507
- transaction 352, 505
- transaction code 483, 505
- transaction data 331
- transaction profile 144
- transaction step 126, 129
- transaction variants 417
- transactional RFC 226, 240, 505
- transferred data volume 245
- Transmission Control Protocol/Internet Protocol 504
- transport 505

- transport domain 505
- Transport Domain Controller 504
- Transport Management System 504
- Transport Organizer 504
- trRFC tables 241
- TSV\_TNEW\_PAGE\_ALLOC\_FAILED 313
- TSV\_TNEW\_PG\_CREATE\_FAILED 299

## U

- Uniform Resource Locator 505
- UNIX 28, 78, 304, 307
- update 99, 202
- update request 202
- update service 194
- update statistics 96
- update type 205
- update work process 204
- URL 505
- user call 456
- user connection 32
- user context 130, 289, 505
- user profile 147

## V

- V1 update 205
- V2 update 205
- V3 update 205
- VBBE 365
- VBDATA 203
- VBHDR 203, 342, 393
- VBMOD 203
- virtual main memory 318
- virtual memory 287, 505
- virtual memory required 302

## W

- WAN 505
- WB0 505
- web application 256, 282
- web browser 498
- web connection 256
- web RFC 253, 256
- WGate 252, 258
- WHERE clause 403
- Wide Area Network (WAN) 36, 505
- Windows 28, 306, 308
- Windows NT 78

- work process 113, 115, 196, 505
- work process overview 109
- work process types 125
- Workbench Organizer 505
- workload analysis 123, 128, 136
- workload distribution 141, 191
- Workload monitor
  - central 151
- workload monitor 124, 128
- Workloada analysis
  - cross-component 150
- World Wide Web 505
- WP 505
- WWW 505

## **X**

- XML 505

## **Z**

- zero administration memory management 105, 296, 319
- ztta/roll\_area 291, 296
- ztta/roll\_extension 291, 296, 312, 315
- ztta/roll\_first 293, 294, 296