



# Performanceoptimierung für SAP S/4HANA®

- › Das Standardwerk zu SAP-Performanceanalyse und -Tuning
- › Beste Leistung für SAP S/4HANA Cloud und on-premise
- › Neue Monitoring-Apps, Performance für SAP HANA und SAP Fiori u. v. m.

9., aktualisierte Auflage

Thomas Schneider

# Einleitung

Warum ist die Performance Ihrer betriebswirtschaftlichen IT-Anwendung wichtig? Nur bei guten Antwortzeiten können Benutzer motiviert und effizient mit der Anwendung arbeiten. Ein langsames System führt zu Ausfallzeiten und Frustration. Eskaliert die Situation, wird im schlimmsten Fall der zur Bewältigung der Geschäftsprozesse erforderliche Durchsatz nicht mehr erreicht. Mehrarbeit, Produktionsverzögerungen und finanzielle Verluste sind die Folge. Umgekehrt steigert eine systematische, proaktive Performanceoptimierung maßgeblich den Nutzen Ihrer betriebswirtschaftlichen Anwendung.

Die Performance eines Datenverarbeitungssystems ist als die Fähigkeit definiert, gegebene Anforderungen an Antwortzeiten und Datendurchsatz zu erfüllen. Solche Anforderungen können z. B. sein, dass innerhalb 1 Stunde ein Durchsatz von 10.000 gedruckten Rechnungen erreicht werden muss oder dass die Antwortzeit für das Erfassen eines Kundenauftrags unter 1 Sekunde liegen soll. Eine gute Performance ist keine absolute Eigenschaft einer betriebswirtschaftlichen Anwendung, sondern immer relativ zu den Anforderungen an diese zu sehen.

Unter *Performanceoptimierung* verstehen wir in diesem Buch einen Prozess, der immer fünf Phasen umfasst: Die ersten beiden Phasen sind, zu einem *Verständnis der Geschäftsprozesse* zu kommen sowie *Performanceziele festzulegen und zu quantifizieren*. Diese Schritte beziehen alle Beteiligten mit ein, d. h. Techniker und Anwendungsfachleute. Nur auf der Basis dieser Voraussetzungen kann eine Optimierung erfolgreich sein. Die Phasen drei bis fünf umfassen dann die *systematische Überwachung, Identifizierung und Analyse* von Problemen, die *Umsetzung von Optimierungsmaßnahmen* und die erneute Analyse zur *Verifizierung des Erfolgs* der eingeleiteten Maßnahmen (siehe Abbildung 1). Vor unreflektiertem »Herumschrauben« an Konfigurationsparametern und vergleichbaren »Tuningschnellschüssen« müssen wir warnen! Ziel dieses Buches ist es vielmehr, Ihnen die Möglichkeiten zu eröffnen, Performanceprobleme zu identifizieren und zu analysieren, um diese dann gezielt beheben zu können.

Performance-  
optimierung

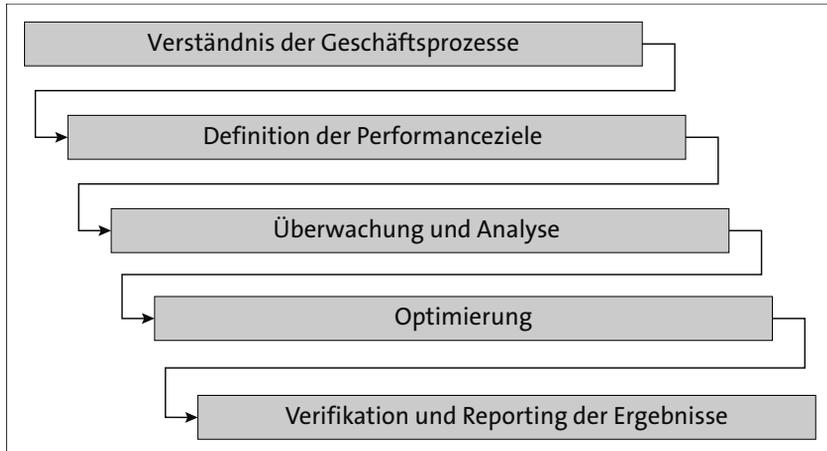


Abbildung 1 Performanceoptimierung in fünf Phasen

**Technische Optimierung**

Eine betriebswirtschaftliche IT-Anwendung besteht technisch gesehen aus zahlreichen Komponenten: aus logischen Komponenten, Prozessen wie Services, Threads oder Workprozessen, aus Speicherbereichen wie Puffern und aus Benutzerkontexten sowie aus physischen Komponenten wie Prozessoren (CPU), Hauptspeicher (RAM), Festplatten und Netzwerksegmenten. Diese Komponenten erlauben jeweils einen maximalen Durchsatz und eine optimale Antwortzeit. Ist das Zusammenspiel der Komponenten nicht ideal abgestimmt oder wird die Leistungsgrenze einzelner Komponenten erreicht, entstehen Wartesituationen, die sich negativ auf Durchsatz und Antwortzeit auswirken. Die Identifizierung, Analyse und Lösung solcher Probleme durch das Abstimmen der Komponenten aufeinander und die Verteilung der im System anfallenden Last sind die Aufgaben, die wir in diesem Buch als *technische Optimierung* bezeichnen.

**Optimierung von Anwendungen**

Der zweite wichtige Bereich der Performanceoptimierung ist die Vermeidung unnötiger Last. Die Performance kann ebenso von ineffizienten Programmen oder dem ineffizienten Einsatz von Programmen negativ beeinflusst werden. Die Optimierung einzelner Programme bezeichnen wir als *Anwendungsoptimierung*. Ziel der Optimierung ist es, zunächst die Systemeinstellung und die Anwendungen derart zu verbessern, dass auf der Basis der bereits vorhandenen Hardwareressourcen die gewünschte Performance erreicht wird. Genügt dies nicht, müssen die Ressourcen entsprechend den gewonnenen Erkenntnissen erweitert werden.

**Wie viel Tuning ist nötig?**

Wie viel Aufwand ist für Performanceanalyse und Tuning einer SAP-Lösung nötig? Die Antwort auf diese Frage hängt stark von der Systemgröße ab. Für eine kleine oder mittelgroße Installation ohne Modifikationen am SAP-

Standard und ohne Eigenentwicklungen genügt in der Regel eine Performanceoptimierung kurz vor und kurz nach dem Produktivstart sowie nach größeren Veränderungen, z. B. durch Upgrades, Produktionsstarts neuer SAP-Lösungen, die Erhöhung der Benutzerzahl, größere Datenübernahmen oder Mandantentransporte. Darüber hinaus muss man natürlich nach Bedarf eingreifen, wenn akute Performanceprobleme vorliegen. Das Tuningpotenzial und damit auch der notwendige Analyse- und Optimierungsaufwand wachsen mit der Systemgröße an. Die Praxis zeigt, dass insbesondere kundeneigene Entwicklungen und Modifikationen am SAP-Standard zu Performanceengpässen führen. Ursache dafür sind meist ungenügende Tests, aber auch Zeitdruck und mangelnde Erfahrung der Entwickler. Den Extremfall bildet eine ständig weiterentwickelte Großinstallation mit vielen hundert Benutzern, komplizierten Prozessketten, einem Dutzend oder mehr Entwicklern (die oft verschiedenen Beratungsfirmen angehören und zu unterschiedlichen Zeiten und an unterschiedlichen Orten am System arbeiten) und einem ausgelagerten Systemmanagement. In einer solchen Systemumgebung ist es unumgänglich, dass eine kleine Gruppe von Administratoren und Entwicklern die Übersicht über das Gesamtsystem behält und die Performance nicht aus den Augen lässt.

Hilfe bei Performanceanalyse und Tuning bieten die Remote-Services von SAP. Namentlich sind dies der SAP GoingLive Check, der Ihnen den Produktivstart Ihres Systems erleichtert, und der SAP EarlyWatch Alert, der Ihr produktives System überwacht und Ihnen weitere Optimierungen vorschlägt.

Wie trägt ein *proaktives Performancemanagement* dazu bei, Ihr Ziel zu erreichen, eine betriebswirtschaftliche Anwendung erfolgreich zu betreiben? Wenn Sie dieses Ziel erreichen wollen, müssen Sie zwei Einflussfaktoren im Blick halten: die Zufriedenheit der Benutzer und die Kosten für den Betrieb der betriebswirtschaftlichen Anwendung. Die Kosten für den Betrieb ergeben sich zum einen aus den Kosten für Hardware (Infrastruktur, CPU, Hauptspeicher, Festplatten und Netzwerke) und Betriebspersonal (Administration, Wartung, Fehleranalyse). Nicht vernachlässigen dürfen Sie aber die Kosten, die entstehen, wenn eine Anwendung nicht zur Verfügung steht oder nicht die geforderte Performance erreicht – der Schaden übersteigt in diesem Fall innerhalb weniger Stunden oder Tage den Betrag, der im Mittel in einem Jahr in die proaktive Performanceoptimierung investiert wird. Diese Risikokosten müssen ebenfalls den Kosten für ein proaktives Performancemanagement gegenübergestellt werden. Tabelle 1 führt Ihnen anhand zweier konkreter Beispiele den Nutzen des proaktiven Performancemanagements vor Augen.

Proaktives  
Performance-  
management

Proaktive Maßnahme	Effekt im System	Sofortiger Nutzen durch höhere Benutzerzufriedenheit	Sofortiger Nutzen durch geringere Betriebskosten	Vermindertes Eskalationsrisiko
Optimierung von SQL-Anweisungen	Reduktion der Datenbanklast	schnellere Antwortzeiten bei bestimmten Transaktionen	Hardwareinvestitionen (Datenbankserver, Speichersystem) können gestreckt werden.	Überlastung des Datenbanksystems wird vermieden.
proaktives Datenmanagement (Datenvermeidung, Archivierung, Reorganisation)	<ul style="list-style-type: none"> <li>■ Reduktion des Datenbankwachstums</li> <li>■ kürzere Zeiten für Wartungsarbeiten an der Datenbank (Backup/Recovery, Upgrade, Migration, Systemkopie)</li> </ul>	<ul style="list-style-type: none"> <li>■ schnellere Antwortzeiten bei bestimmten Transaktionen</li> <li>■ kürzere Downtime bei Wartungsarbeiten</li> </ul>	<ul style="list-style-type: none"> <li>■ Hardwareinvestitionen können gestreckt werden.</li> <li>■ geringerer Personalbedarf bei Wartungsarbeiten</li> </ul>	Datenbankgröße bleibt »beherrschbar«.

**Tabelle 1** Beispiele für den Nutzen eines proaktiven Performancemanagements

**Herausforderungen durch das Internet und mobile Geräte**

Mit dem Aufstieg des Internets, der Smartphones und Tabletcomputer vollzieht sich ein Paradigmenwechsel in der Welt der betriebswirtschaftlichen Software: Nicht mehr auf den hoch spezialisierten Angestellten an seinem PC zielt die Software ab, sondern auf Benutzer des Internets (extern oder firmenintern als Intranet) oder Benutzer eines mobilen Geräts. Der klassische Ansatz einer Prozessautomatisierung mit SAP R/3 basierte auf hoch spezialisierten Anwendern, die von festen Arbeitsplätzen über installierte SAP GUIs auf ihr ERP-System (Enterprise Resource Planning) zugreifen. Die Rolle dieses spezialisierten Vermittlers, der ausgebildet sein muss, um Software zu bedienen, wird allerdings an vielen Stellen zunehmend überflüssig. Stattdessen erhält der Endbenutzer durch Internet und Mobilgerät direkten Zugriff auf die ERP-Systeme der Unternehmen. Heute können in vielen Unternehmen die Mitarbeiter über das Intranet ihre Arbeits- und Fehlzeiten,

Reisekostenabrechnungen, Bestellungen etc. selbst in das System eingeben, wo dies früher über zentrale Benutzer geschah. Kunden bestellen heute ihre Produkte direkt über das Internet und nicht mehr über den Umweg eines Briefes, Faxes oder Telefonanrufes bei einem Vertriebscenter.

Die Erwartungen, die der Anwender einer Internet- oder mobilen Anwendung an die Bedienbarkeit und Performance hat, sind ungleich höher als die des klassischen Angestellten an sein ERP-System. Der Angestellte ist auf »sein« ERP-System angewiesen, und wenn es ihm normalerweise dabei hilft, seine tägliche Arbeit zu vereinfachen, wird er es akzeptieren und auch kleinere Fehler oder Performanceschwächen in Kauf nehmen. Ganz anders der Internetbenutzer: Funktionieren die Anwendungen, die ihm im Internet angeboten werden, nicht einfach und performant, kann er sofort zur Konkurrenzfirma wechseln und dort z. B. seine Einkäufe tätigen (»Die Konkurrenz ist nur einen Mausklick entfernt.«). Hinzu kommt: Das Internet macht nicht um 17 Uhr Feierabend – von einer E-Business-Lösung im Internet werden Verfügbarkeit und Performance an 365 Tagen 24 Stunden lang gefordert. Nutzer von mobilen Anwendungen legen die Maßstäbe in Bezug auf Bedienbarkeit und Performance an eine SAP-Anwendung an, die sie auch von anderen Mobile Apps gewohnt sind.

Mit SAP HANA und SAP S/4HANA ist es SAP gelungen, seine ERP-Lösung komplett neu zu erfinden. SAP HANA macht sich die Verfügbarkeit von riesigen Hauptspeichern und massiv parallelen Prozessorarchitekturen zunutze und setzt diese konsequent durch modernste Softwarearchitektur in Performance um. Vielleicht werden Sie sich fragen: Brauche ich überhaupt noch ein Buch über Performance, wenn ich SAP HANA habe? Oder löst SAP HANA alle Performanceprobleme? Die Antwort auf die zweite Frage ist – nach der festen Überzeugung des Autors – ein doppeltes Nein: Zum einen kann auch SAP HANA keine Wunder vollbringen, wenn Ihr Programm ganze Datenbanktabellen in den Applikationsserver liest und – schlimmer noch – die Daten weiter zum Webbrowser des Anwenders schickt. Der Löwenanteil der Laufzeit entfällt dann auf den Applikationsserver, das Netzwerk und den Browser. Die grundlegenden Regeln zur performanten Programmierung sind auch bei SAP HANA nicht außer Kraft gesetzt! Ein zweites Nein ergibt sich aus der Tatsache, dass jeder technischen Innovation »Begehrlichkeiten« oder, anders formuliert, Herausforderungen entgegenstehen.

SAP HANA und  
SAP S/4HANA

SAP S/4HANA nutzt die Optionen von SAP HANA auf drei Ebenen voll aus:

- Es profitiert von den Performanceverbesserungen, die insbesondere bei analytischen Anfragen durch SAP HANA entstehen.
- Durch die Optimierung der Datenmodelle und Programme für SAP HANA werden neue Prozesse möglich.

- Durch die Möglichkeiten von SAP HANA werden völlig neue Datenquellen erschlossen.

**Kleine »Fehler« –  
große Wirkung**

Durch die Umstellung der Prozesse werden Berechnungen und Analysen auf Daten möglich, die die Datenmenge traditioneller betriebswirtschaftlicher Daten um ein Vielfaches übersteigt. Übertragen auf unsere Performancefrage heißt das: Ein nicht performanceoptimiertes System oder Programm hat einen viel höheren Effekt als in einem »traditionellen« SAP-Business-Suite-System – als Konsequenz folgern wir: Auch in Zukunft wird Performance-Know-how ein wertvolles Gut sein.

**Cloud**

Bereits seit Jahren verfolgt SAP eine klare Cloud-Strategie. SAP S/4HANA wird als *das* Cloud ERP vermarktet. Mit dem Betrieb in der Cloud ist eine klare Trennung der Verantwortlichkeiten (*Separation of Concern*) zwischen dem Cloud-Anbieter (Provider) und dem Nutzer verbunden, die sich in den Administrationsaufgaben und Werkzeugen widerspiegelt. Damit ändern sich auch die Aufgaben der Performanceanalyse:

- Im Fall von SAP S/4HANA Cloud Public Edition werden der Systembetrieb und auch viele Aufgaben der Performanceüberwachung vom Cloud-Provider übernommen. Dennoch bleiben im Bereich der Anwendungsoptimierung noch Aufgaben, die Sie als Nutzer übernehmen müssen.
- Im Fall von SAP S/4HANA Cloud Private Edition liegt der Systembetrieb in den Händen des Cloud-Providers, allerdings sind Sie als Nutzer für die Performanceüberwachung und -optimierung verantwortlich.
- Bei der Nutzung als On-Premise-Produkt liegen alle Betriebsaufgaben bei Ihnen als Kunde.

In allen drei Varianten können Sie Erweiterungen und Kundenentwicklungen vornehmen. Die Berücksichtigung der Performance bei der Entwicklung, Performancetests und die Überwachung im Produktivbetrieb sind weitere Aufgaben der Performanceoptimierung, die in Ihrer Verantwortung als Kunde von SAP liegen.

**Zu diesem Buch**

**Grundlagen des  
Buches**

Die in diesem Buch dargestellten Methoden der Performanceanalyse und -optimierung entsprechen den Verfahren, die ursprünglich von den Experten des SAP EarlyWatch Alerts und des SAP GoingLive Checks verwendet und in den SAP-Basis-Trainings *Workload Analysis* und *Optimierung von ABAP-Programmen* vermittelt werden. Mit jeder neuen Auflage dieses Buches – mittlerweile der neunten – wird die Chance genutzt, aktuelle Trends

der Produktentwicklung bei SAP und – soweit relevant – Entwicklungen der IT-Welt allgemein adäquat zu behandeln.

Der Fokus des Buches liegt auf SAP S/4HANA. Es deckt die folgenden Varianten ab:

**Systembezug**

- SAP S/4HANA Cloud Public Edition
- SAP S/4HANA Cloud Private Edition
- SAP S/4HANA (on premise)

Sie profitieren von den in diesem Buch beschriebenen Methoden aber auch, wenn Sie eines der folgenden Produkte nutzen:

- SAP ERP (mit SAP HANA oder einer »klassischen« Datenbank)
- SAP BTP ABAP Environment (Entwicklung von Kunden- und Partnererweiterungen)
- SAP BW/4HANA (Optimierung von analytischen Anfragen)

In der neunten Auflage haben wir folgende Änderungen vorgenommen:

**Neuerungen in dieser Auflage**

- Das Kapitel zu SAP HANA wurde komplett überarbeitet. Dies trägt sowohl der strategischen Bedeutung dieses Themas als auch den gewachsenen Erfahrungen damit Rechnung. Deutlich erweitert wurde der Abschnitt zu Core Data Services. Neu hinzugekommen sind unter anderem die Themen SAP HANA Native Storage Extension (NSE) und der neue SQL Analyzer. Auch in allen anderen Kapiteln steht jetzt SAP HANA als Datenbankplattform im Fokus.
- Das Kapitel zu analytischen Anwendungen haben wir komplett neu geschrieben. Es behandelt SAP S/4HANA Embedded Analytics und analytische Anfragen in SAP BW/4HANA.
- Das Kapitel zu Web-UIs und Webservices wurde ebenfalls komplett überarbeitet. Insbesondere wurden die Abschnitte zu SAP Fiori, SAPUI5, OData und SAP Gateway aktualisiert.
- Aufgrund der steigenden Bedeutung der Public Cloud mit den Produkten SAP S/4HANA Cloud Public Edition und SAP BTP ABAP Environment haben wir das neue SAP-Fiori-basierte technische Monitoring (Technical Monitoring Cockpit, TMC), das auch in der Public Cloud genutzt werden kann, in allen Kapiteln berücksichtigt, ohne die »alten« Werkzeuge zu vernachlässigen.
- Alle weiteren Kapitel sind gründlich durchgesehen und überarbeitet worden.

Alle Inhalte zum SAP NetWeaver Application Server Java und auch das Java-Kapitel aus den Voraufgaben sind entfallen.

Weiterhin sind folgende Themen neu aufgenommen worden:

- Workprozess-Stichprobenanalyse (/SDF/SMON, siehe Kapitel 2)
- HTTP-Frontend-Statistiken (siehe Kapitel 3)
- individuelle Speicherquoten und Warnungen im Memory Management (siehe Kapitel 6)
- Pufferung von CDS-Views (siehe Kapitel 11)

Aufbau dieses Buches

Abbildung 2 stellt die Kapitel dieses Buches, orientiert an den fünf Phasen der Performanceoptimierung, im Überblick dar. **Kapitel 1**, »Performance-management einer SAP-Lösung«, wendet sich sowohl an SAP-Administratoren und SAP-Berater als auch an Anwendungsentwickler und SAP-Projektleiter. Es behandelt auf einem nicht technischen Niveau die folgenden grundlegenden Fragen zur Performanceanalyse:

- Welche Vorkehrungen müssen getroffen werden, um eine optimale Performance einer SAP-Lösung zu gewährleisten?
- Welche Maßnahmen zum Performancetuning kommen in Betracht?
- Welche Personen sind in den Tuningprozess involviert?

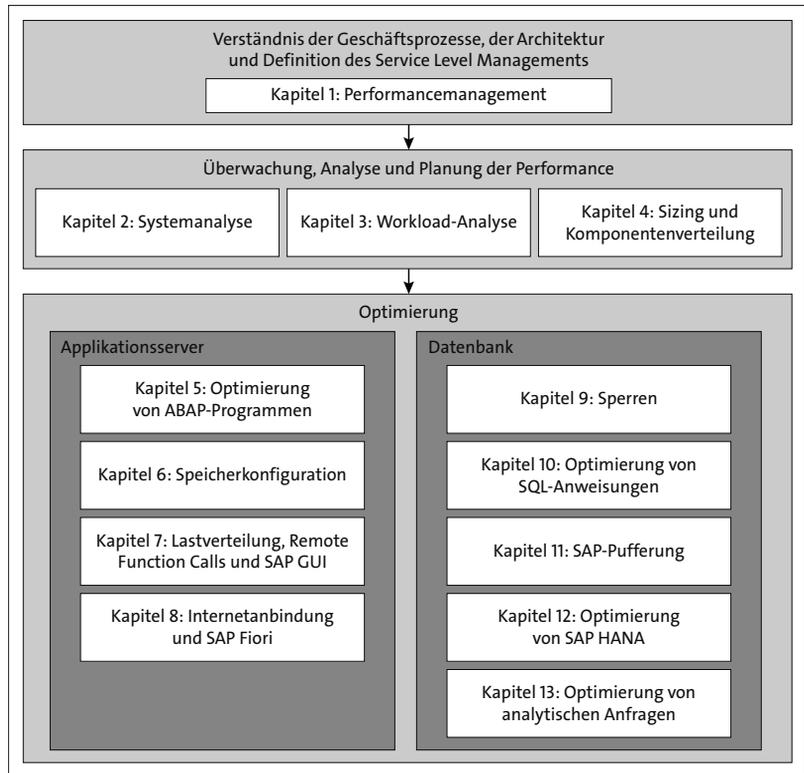


Abbildung 2 Die Kapitel dieses Buches

Hinter der dem Benutzer angebotenen Leistung verbirgt sich in der Praxis oft ein Netz von Partnern, die jeweils Teilleistungen erbringen. Viele Teile werden von unterschiedlichen, manchmal externen Service Providern erbracht. Um diese Komplexität in den Griff zu bekommen, führen die meisten Leistungsanbieter und Kunden ein *Service Level Management (SLM)* ein. Als SLM bezeichnet man eine strukturierte proaktive Methode, die das Ziel hat, den Benutzern einer IT-Anwendung ein adäquates Serviceniveau zu garantieren – in Übereinstimmung mit den betriebswirtschaftlichen Zielen des Auftraggebers und bei optimalen Kosten. Wir zeigen in diesem Buch, mit welchen Werkzeugen und Methoden Sie ein SLM für eine SAP-Lösung einführen.

In diesem Buch folgen wir zunächst der Strategie der Bottom-up-Analyse und beginnen in **Kapitel 2**, »Analyse von Hardware, Datenbank und ABAP-Applikationsserver«, mit den Untersuchungen von Betriebssystem, Datenbank, SAP-Speicherkonfiguration und Workprozessen. Gleichzeitig werden Lösungsvorschläge angeboten, die den Administrator oder Berater befähigen sollen, die wichtigsten Performanceprobleme zu lösen. Für kleine und mittelgroße Installationen wird diese Stufe des Tunings in vielen Fällen bereits ausreichen.

Anschließend wird in **Kapitel 3**, »Workload-Analyse«, die komplexere Workload-Analyse als Top-down-Analyse diskutiert. Nach Lektüre der ersten Kapitel sind Sie in der Lage, eine systematische Performanceanalyse für ein SAP-System mit ABAP-Server, Datenbank und Betriebssystem durchzuführen.

In **Kapitel 4**, »Hardware-Sizing und Komponentenverteilung«, stellen wir zum einen dar, was Sie beachten müssen, um eine optimal ausgelegte Hardware für Ihr SAP-System zu erreichen. Die Voraussetzung dafür ist, dass auf der einen Seite keine Hardwareengpässe entstehen, dass aber auf der anderen Seite nicht zu hohe Kosten durch unnötig vorgehaltene Hardware anfallen. Zum anderen stellen wir in diesem Kapitel die Themen Serverkonsolidierung, Virtualisierung und Skalierung dar.

Die weiteren Kapitel 5 bis Kapitel 13 vermitteln Wissen, das für eine umfassende Performanceanalyse notwendig ist. Sie wenden sich an SAP-Betreuer und Entwickler, die für das performante Funktionieren großer Systeme verantwortlich sind und die das Tuningpotenzial ihrer Systeme voll ausschöpfen wollen und müssen. Kapitel 5 bis Kapitel 13 sind weitgehend eigenständige Einheiten und können mit Kenntnis der ersten vier Kapitel in beliebiger Reihenfolge gelesen werden. Eventuelle Abhängigkeiten sind am Anfang jedes Kapitels vermerkt.

In Kapitel 5 bis Kapitel 8 werden die Themen behandelt, die sich auf den Applikationsserver und den Präsentationsserver beziehen:

- **Kapitel 5, »Optimierung von ABAP-Programmen«:** Zeigt Methoden zur Analyse einzelner ABAP-Programme auf, unter anderem mit den Hilfsmitteln Performance-Trace und ABAP-Laufzeitanalyse, Debugger, ABAP Test Cockpit, und gibt Hinweise zur Optimierung des ABAP-Codes.
- **Kapitel 6, »Speicherkonfiguration« (Memory Management):** Die Konfiguration der auf dem ABAP-Server allokierten Speicherbereiche hat wesentlichen Einfluss auf die Performance.
- **Kapitel 7, »Lastverteilung, Remote Function Calls und SAP GUI«:** Eine optimale Lastverteilung von Webanfragen, Dialog-, Verbuchungs- und Hintergrundaufträgen hilft dabei, Hardware optimal zu nutzen und Engpässe aufgrund falscher Konfiguration zu vermeiden. Die Performance von Schnittstellen zwischen Softwarekomponenten trägt ebenfalls maßgeblich zur Performance der gesamten Lösung bei. Im Detail werden hier auch die Performanceaspekte von SAP GUI und SAP GUI Controls behandelt.
- **Kapitel 8, »Internetanbindung und SAP Fiori«:** Der Anbindung von SAP-Systemen über das Internet (HTTP-Protokoll) als Web-UIs oder Webservices kommt immer größere Bedeutung zu. Im Detail werden hier die Performanceaspekte von SAP GUI for HTML (Internet Transaction Server, ITS), von Business Server Pages und von Web Dynpro ABAP sowie von SAP Fiori, SAPUI5 und OData/Gateway behandelt.

Kapitel 9 eröffnet die Reihe der Kapitel zu Datenbankthemen. Das zentrale Kapitel in diesem Block ist Kapitel 10, das Voraussetzung für die folgenden Kapitel 11 bis Kapitel 13 ist:

- **Kapitel 9, »Sperrungen«:** Datenbank- und SAP-Sperrungen gewährleisten die Datenkonsistenz. Durch eine optimierte Verwaltung von Sperrungen (z. B. durch effiziente Programmierung oder auch durch die Pufferung von Nummernkreisen) werden Durchsatzengpässe vermieden.
- **Kapitel 10, »Optimierung von SQL-Anweisungen«:** Ineffektive SQL-Anweisungen beanspruchen die Datenbank übermäßig und werden daher zum Problem für die Performance der gesamten Anwendung. In diesem Kapitel stellen wir die Detailanalyse für »teure« SQL-Anweisungen sowie Optimierungsmöglichkeiten durch Datenbankindizes und durch Programmoptimierung (fünf goldene Regeln) dar.
- **Kapitel 11, »Pufferung auf dem Applikationsserver«:** Die Pufferung von Tabellen auf den Applikationsservern beschleunigt den Zugriff auf oft gelesene Daten und dient der Entlastung der Datenbank.

- **Kapitel 12**, »Optimierung von Datenbankanfragen mit SAP HANA«: Dieses Kapitel behandelt umfassend die Grundlagen, Werkzeuge und Methoden der Performanceanalyse und -optimierung auf SAP HANA.
- **Kapitel 13**, »Optimierung analytischer Anfragen«: Analytische Anfragen sind spezielle SQL-Anweisungen, die in der Regel große Datenmengen bearbeiten – für diese Art von Anfragen gibt es spezielle Optimierungsmöglichkeiten.

Kenntnisse in der Performanceoptimierung von SAP-Systemen und -Anwendungen sind für SAP-Administratoren, SAP-Anwendungsbetreuer, SAP-Entwickler und SAP-Projektleiter von größtem Nutzen – an diese Zielgruppen richtet sich dieses Buch. Jedem Kapitel ist am Ende der Einleitung ein kurzer Abschnitt mit dem Titel »Wann sollten Sie dieses Kapitel lesen?« vorangestellt. Aus diesem ergibt sich die genaue Zielgruppe des jeweiligen Kapitels.

Zielgruppen

Dieses Buch setzt an den Stellen, an denen es um die konkrete Umsetzung von Empfehlungen geht, Kenntnisse in der Administration von SAP-Komponenten in Theorie und Praxis voraus. Als gute Vorbereitung kann z. B. das Buch und »SAP S/4HANA – Systemadministration« dienen (siehe Anhang D, »Informationsquellen«). Teile dieses Buches, z. B. Kapitel 5, setzen Wissen über die Programmiersprache ABAP, Kapitel 9 bis Kapitel 13 Kenntnisse über die Funktionsweise von relationalen Datenbanken und SQL voraus.

Voraussetzungen

In den folgenden Bereichen stößt das Buch an seine Grenzen:

Grenzen des Buches

#### ■ **Hardware- und Netzwerktuning**

Ein Engpass bei CPU, Hauptspeicher, I/O oder Netzwerk lässt sich zwar aus dem SAP-System heraus mit großer Wahrscheinlichkeit identifizieren, für eine detaillierte Analyse werden jedoch die Werkzeuge der Hardware- bzw. Netzwerkanbieter benötigt. Das gilt insbesondere für die Nutzung von Hyperscaler-Plattformen wie Amazon Elastic Compute Cloud (EC2), Microsoft Azure Virtual Machine Services und Google Cloud.

#### ■ **Datenbanken**

Der Schwerpunkt dieses Buches liegt im SAP-spezifischen Kontext des Datenbanktunings und in der Vermittlung der Konzepte von SAP HANA. Mit den Architekturen anderer Datenbanksysteme, die noch bei SAP ERP zum Einsatz kommen, beschäftigen wir uns nicht. Allerdings stellen wir mit dem DBA-Cockpit ein Werkzeug dar, das die Administration und Analyse der verschiedenen Datenbanksysteme so weit wie möglich vereinheitlicht.

■ **Anwendungstuning**

Viele Performanceprobleme lassen sich nur mit detailliertem Anwendungswissen und Kenntnis der einzelnen SAP-Lösungen lösen. Oft ist es eine Änderung im Customizing, die die Lösung des Problems bringt. Know-how zum Tuning einzelner SAP-Lösungen ist nicht Teil dieses Buches. Allerdings vermittelt es Ihnen Analysestrategien, sodass Sie Performanceprobleme auf bestimmte Anwendungen eingrenzen, mit anwendungsspezifischen SAP-Hinweisen arbeiten oder Probleme an den richtigen Entwickler oder Berater adressieren können.

■ **SAP Business Technology Platform (SAP BTP)**

Kein SAP-Kunde kommt ohne Erweiterungen von SAP S/4HANA aus. Diese Erweiterungen können Sie on stack (d. h. direkt auf dem SAP S/4HANA-System) oder side by side (mit den Services der SAP BTP) anlegen. Für On-Stack-Erweiterungen enthält dieses Buch alle Informationen, die Sie zur Performanceoptimierung benötigen. Auch für Side-by-Side-Erweiterungen, die Sie mit dem SAP BTP ABAP Environment anlegen, vermittelt dieses Buch das nötige Performancewissen. Alle anderen Services der SAP BTP werden in diesem Buch nicht behandelt.

**Versions-  
abhängigkeit**

Dieses Buch enthält zahlreiche zeitabhängige Informationen und Regeln. Nur so kann Ihnen dieses Buch als Nachschlagwerk für die tägliche Arbeit in der SAP-Administration dienen. Auf der anderen Seite ist klar: Eine neue Version, ein Patch (der SAP-Komponente, der Datenbank oder des Betriebssystems), eine neue Rechnergeneration – diese und andere Faktoren können mit einem Schlag alte Informationen wertlos machen; im schlimmsten Fall können sich veraltete Empfehlungen sogar kontraproduktiv auf die Performance auswirken. Dieses Buch ist also kein Gesetzbuch und wer Performanceoptimierung nur als stures Befolgen von Regeln betreibt, handelt fahrlässig. Die direkte Auseinandersetzung mit der Lösung, der SAP-Onlinehilfe und aktuellen SAP-Hinweisen im SAP Support Portal kann dieses Buch nicht ersetzen, sondern nur fördern.

Alle Angaben zu Menüpfaden, Bezeichnungen auf den Bildschirmen der Performancemonitore und Richtwerten für Performancekennzahlen beziehen sich – sofern nicht anders vermerkt – auf SAP S/4HANA 2023, SAP Basis 7.58 und SAP HANA Version 2, SPS08.

**Hinweise zur  
Lektüre**

In diesem Buch finden Sie mehrere Orientierungshilfen, die Ihnen die Arbeit mit dem Buch erleichtern sollen.

In hervorgehobenen Informationskästen sind Inhalte zu finden, die wissenswert und hilfreich sind, aber etwas außerhalb der eigentlichen Erläute-

rung stehen. Damit Sie die Informationen in den Kästen sofort einordnen können, haben wir die Kästen mit Symbolen gekennzeichnet:

- Die mit diesem Symbol gekennzeichneten *Tipps* geben Ihnen spezielle Empfehlungen, die Ihnen die Arbeit erleichtern können. 
- In Kästen, die mit diesem Symbol gekennzeichnet sind, finden Sie Informationen zu *weiterführenden Themen* oder wichtigen Inhalten, die Sie sich merken sollten. 
- Dieses Symbol weist Sie auf *Besonderheiten* hin, die Sie beachten sollten. Es *warnt Sie* außerdem vor häufig gemachten Fehlern oder Problemen, die auftreten können. 
- *Beispiele*, durch dieses Symbol kenntlich gemacht, weisen auf Szenarien aus der Praxis hin und veranschaulichen die dargestellten Funktionen. 

## Kapitel 2

# Analyse von Hardware, Datenbank und ABAP-Applikationsserver

*Mit der Analyse der Hardwareressourcen, der Datenbank sowie der Workprozesse und Speicherbereiche des ABAP-Servers steigen wir – bottom-up – in die Performanceanalyse ein. Verschaffen Sie sich so einen ersten Überblick über die aktuelle Situation im System!*

In diesem Kapitel erläutern wir die Grundlagen von Performanceanalysen der Workprozesse, der Hardware, der Datenbank und der Speicherkonfiguration des ABAP-Servers. Am Ende eines jeden Abschnitts finden Sie Flussdiagramme, die die wichtigsten Analysepfade zusammenfassen und die Abhängigkeiten zwischen den Analysen verdeutlichen. Im letzten Abschnitt gehen wir auf den zentralen Überwachungsmonitor ein, der Performanceindikatoren aus allen Bereichen integriert.

Zu diesen Analysen erhalten Sie sofort Optimierungsvorschläge, sofern diese ohne umfangreichere Erklärungen möglich sind. Um auch einem in der Performanceanalyse unerfahrenen Berater oder Administrator einen schnellen Einstieg zu ermöglichen, verzichten wir bewusst auf Hintergrundinformationen. So wird z. B. beschrieben, wie Sie den Extended Memory überwachen und einstellen, ohne den Begriff Extended Memory zu erklären. Umfassendere Informationen finden Sie anschließend in Kapitel 6 bis Kapitel 13. Dieser Aufbau trägt unserer Erfahrung Rechnung, dass es möglich ist, viele Performanceprobleme im Bereich von Betriebssystem, Datenbank und SAP-Basis anhand einfacher Handlungsanweisungen zu lösen, ohne sich vorher eingehend mit den Details der Architektur zu beschäftigen.

### **Wann sollten Sie dieses Kapitel lesen?**

Dieses Kapitel sollten Sie lesen, wenn Sie die Performance des SAP-Systems, der Datenbank oder des Betriebssystems vom SAP-System aus technisch überwachen und optimieren wollen.



## 2.1 Begriffsklärungen

Die Begriffe *Rechner*, *Server*, *Applikationsserver*, *ABAP-Instanz*, *ABAP-Server*, *Datenbank*, *Datenbankserver* und *Datenbankinstanz* werden in diesem Buch wie folgt verwendet:

- Rechner** Ein *Rechner* (oder Computer) ist eine physische Maschine mit CPU, Hauptspeicher, IP-Adresse etc.
- ABAP-Applikationsinstanz** Eine *ABAP-Applikationsinstanz* ist eine administrative Einheit: Sie besteht aus einem Satz von Workprozessen, die von einem Dispatcher verwaltet werden, sowie aus einem Satz von Puffern im Shared Memory des Rechners, auf den die Workprozesse zugreifen. Eine ABAP-Applikationsinstanz nennen wir in diesem Buch auch einfach *ABAP-Instanz* oder *ABAP-Server*. Es kann mehrere Applikationsinstanzen auf einem Rechner geben. In diesem Fall existieren mehrere Dispatcher und mehrere Puffersätze. Ein *Applikationsserver* ist ein Rechner, auf dem mindestens eine ABAP-Applikationsinstanz läuft.
- Datenbank** Jedes SAP-System besitzt genau eine *Datenbank*. Als Datenbank bezeichnen wir die eigentliche Datenbasis, die z. B. in Form von Dateien auf einem Dateisystem vorliegt, sowie den aktiven Teil der Datenbank, der den Zugriff auf die Daten regelt. Den aktiven Teil des Datenbanksystems bezeichnen wir als *Datenbankinstanz*. Eine Datenbankinstanz besteht aus Datenbankprozessen und einem Satz von Datenbankpuffern im Shared Memory eines Rechners. Ein *Datenbankserver* ist ein Rechner, auf dem mindestens eine Datenbankinstanz läuft. Ein Rechner kann zugleich Datenbank- und Applikationsserver sein, wenn eine Datenbankinstanz und eine ABAP-Applikationsinstanz darauf laufen.
- Bei SAP HANA kann die Datenbank horizontal skaliert werden, d. h., Daten können auf unterschiedliche Server verteilt werden. In diesem Szenario operieren also mehrere Datenbankinstanzen, die auch als *Knoten* bezeichnet werden.
- SAP-System** Als *SAP-Systeme* bezeichnen wir in diesem Buch Implementierungen einer SAP-Software mit einem ABAP-Server und einer Datenbank als Grundlage. Beispiele sind SAP-S/4HANA-, SAP-ERP-, SAP-BW- oder SAP-CRM-Systeme.
- Server** Der Begriff *Server* wird in der Dokumentation und der Literatur mehrdeutig verwendet. Er kann sowohl einen Rechner bezeichnen, z. B. im Begriff *Datenbankserver*, als auch einen logischen Service, z. B. in dem Begriff *Message-Server*. So verwenden wir *ABAP-Server* für eine ABAP-Applikationsinstanz.

## 2.2 Analyse der Workprozesse

Starten Sie nun die Analyse der aktuellen Systemperformance mit der Analyse der Workprozesse. Hierzu steht Ihnen die Workprozess-Übersicht zur Verfügung.

Die Workprozess-Übersicht für den aktuellen Applikationsserver (lokale Workprozess-Übersicht) rufen Sie über den Transaktionscode SM50 oder über den folgenden Menüpfad auf: **Werkzeuge • Administration • Monitor • Systemüberwachung • Prozessübersicht**.

Lokale  
Workprozess-  
Übersicht

Sie gelangen auf den Bildschirm **Prozessübersicht**. Um sich die Workprozesse eines beliebigen Applikationsservers anzeigen zu lassen, rufen Sie die Serverübersicht auf: **Werkzeuge • Administration • Monitor • Systemüberwachung • Server** (Transaktionscode SM51).

Markieren Sie anschließend den gewünschten Applikationsserver und wählen Sie im Menü **Springen • Workprozesse**.

Um einen Überblick über alle Workprozesse des SAP-Systems zu gewinnen, starten Sie die globale Workprozess-Übersicht mit **Werkzeuge • Administration • Monitor • Performance • Ausnahmen/Benutzer • Aktive Benutzer • Globale Prozessübersicht**. Geben Sie alternativ den Transaktionscode SM66 ein. Sie gelangen auf den Bildschirm **Globale Workprozess-Übersicht**.

Globale  
Workprozess-  
Übersicht

Wenn das zu untersuchende Performanceproblem so weit eskaliert ist, dass die Workprozess-Übersicht im SAP-System nicht mehr aufgerufen werden kann, starten Sie als Alternative auf Betriebssystemebene die SAP Management Console oder das Hilfsprogramm `dpmon`. Unter der Menüoption **L** finden Sie ebenfalls die Workprozess-Übersicht.

### Technisches Monitoring (TMC)

Für das Monitoring der Server und der Workprozesse stehen im Technischen Monitoring (TMC) die Monitoring-Sichten **Application Server Overview**, **Active Work Processes** und **Work Process Load Summary** zur Verfügung.



### 2.2.1 Felder der Workprozess-Übersicht

In Abbildung 2.1 sehen Sie die lokale Workprozess-Übersicht.

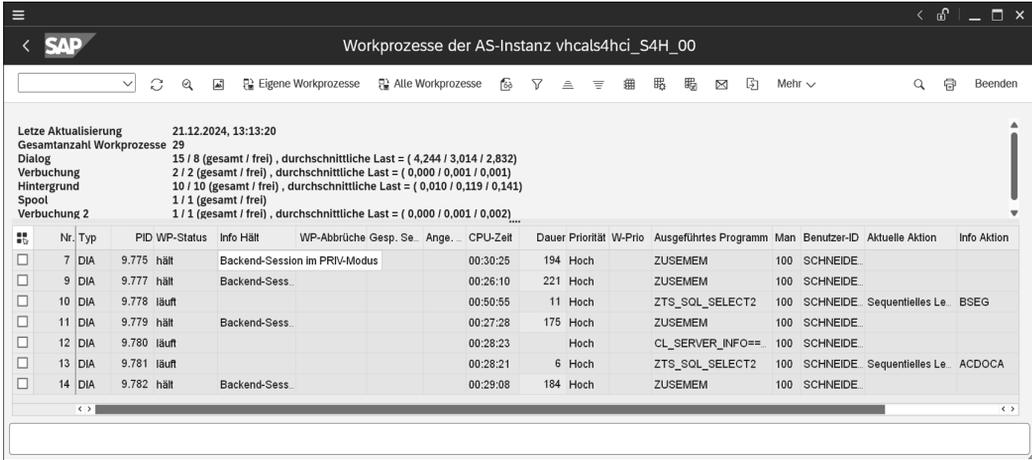


Abbildung 2.1 Workprozess-Übersicht

Die Beschreibung der wichtigsten Felder der lokalen Workprozess-Übersicht finden Sie in Tabelle 2.1.

Feld	Bedeutung
Nr.	Nummer des Workprozesses (eindeutig pro ABAP-Server). Durch die Kombination aus ABAP-Server und Workprozess-Nummer (oder die Kombination Rechnername und Prozess-ID) ist ein Workprozess in einem System eindeutig identifiziert.
Typ	DIA: Dialog, BTC: Hintergrund, UPD: Verbuchung, SPO: Spool
PID	Prozess-ID (Prozessidentifikation) des Betriebssystems (eindeutig pro Rechner). Mit dieser Zahl kann der Prozess mit Kommandos des Betriebssystems bearbeitet (z. B. abgebrochen) werden.
WP-Status	Dieses Feld zeigt den Zustand eines Workprozesses an. Der Status <b>wartet</b> zeigt an, dass der Prozess auf einen Auftrag eines Benutzers wartet. Im Normalfall sollten sich immer ausreichend Workprozesse in diesem Status befinden, ansonsten treten Wartezeiten für die Benutzer auf. Befindet sich der Workprozess im Status <b>läuft</b> , bearbeitet er einen Benutzerauftrag. In diesem Fall kann anhand des Feldes <b>Aktuelle Aktion</b> festgestellt werden, welche Aktion der Workprozess gerade ausführt (siehe unten). Im Status <b>beendet</b> wurde der Prozess durch einen Fehler im SAP-Kernel beendet und im Status <b>hält</b> wartet er auf eine Nachricht.

Tabelle 2.1 Felder der Workprozess-Übersicht

Feld	Bedeutung
Info Hält	Dieses Feld gibt den Wartegrund an, wenn sich der Workprozess im Zustand <b>hält</b> befindet. Eine Übersicht über die wichtigsten Wartegründe und ihre Ursachen finden Sie im nächsten Abschnitt.
Start	Dieses Feld gibt an, ob der Workprozess nach einem Prozessabbruch vom Dispatcher neu gestartet werden soll.
WP-Abbrüche	Anzahl der Abbrüche des Workprozesses
Gesperrte Semaphore	Nummern der Semaphore, die vom Workprozess gehalten werden
Angeforderte Semaphore	Nummern der Semaphore, auf die der Workprozess wartet
CPU-Zeit	bisher benötigte CPU-Zeit des Workprozesses im Format Minuten: Sekunden
Dauer	aktuelle Laufzeit eines Auftrags in Sekunden
Priorität und W-Prio	Priorität des Workprozesses und des Auftrages. Weitere Informationen zu Workprozessprioritäten finden Sie in Abschnitt 7.2.5, »Lastverteilungskonzept für Dialog-Workprozesse«.
Ausgeführtes Programm	Name des augenblicklich ausgeführten Programms
Man	Mandant
Benutzer-ID	Name des Benutzers, dessen Auftrag augenblicklich ausgeführt wird
Aktuelle Aktion	Wenn sich der Workprozess im Status <b>läuft</b> befindet, können Sie diesem Feld die momentane Aktion entnehmen.
Info Aktion	Informationen, z. B. zur Datenbanktabelle, auf die der Workprozess zuletzt zugegriffen hat

Tabelle 2.1 Felder der Workprozess-Übersicht (Forts.)

Die Kombination aus Rechner und *Prozess-ID* erlaubt es, einen Workprozess in einem SAP-System eindeutig zu identifizieren. Die Prozess-ID wird noch in folgenden Monitoren verwendet:

Prozess-ID

- In der Prozessübersicht des Betriebssystemmonitors (Transaktionscode ST06):

**Snapshot • Erste CPU-Prozesse**

Mithilfe dieses Monitors können Sie feststellen, wie stark ein Workprozess augenblicklich die CPU belastet.

- Im Datenbankprozessmonitor (Transaktionscode DBACOCKPIT):

**Performance • Threads** (als Beispiel für SAP HANA)

Dieser Monitor zeigt Ihnen an, welche SQL-Anweisung von einem Workprozess momentan auf der Datenbank prozessiert wird.

- Im Datenbanksperremonitor (Transaktionscode DBACOCKPIT):

**Diagnose • Sperren • Blockierte Transaktionen**

Mithilfe dieses Monitors können Sie ermitteln, ob ein Workprozess augenblicklich eine Datenbanksperre hält bzw. auf die Auflösung einer solchen wartet.

Die Kombination des Informationsgehalts dieser Monitore – Workprozess-Übersicht, Prozessübersicht des Betriebssystemmonitors, Datenbankprozessmonitor und Datenbanksperremonitor – erlaubt Ihnen einen umfassenden Überblick über den aktuellen Zustand des SAP-Systems.

**Hauptspeicherbedarf**

Um den aktuellen Hauptspeicherbedarf zu ermitteln, öffnen Sie die Detailsicht zum Workprozess mit einem Doppelklick auf die betreffende Zeile in der Workprozess-Übersicht.

**Semaphore**

Auf jedem Applikationsserver gibt es Ressourcen, die nur einmalig verfügbar sind. Diese Ressourcen können jeweils nur von einem Workprozess und nicht von mehreren zugleich verwendet werden. Wenn ein Prozess diese Ressource verwenden möchte, setzt er ein *Semaphor*. Andere Prozesse, die diese ebenfalls benötigen, müssen so lange warten, bis der Prozess, der das Semaphor hält, seine Aktion beendet hat. So muss z. B. immer, wenn ein Eintrag in einem SAP-Puffer geändert werden soll, ein Semaphor gesetzt werden, da nur ein einziger Prozess eine Änderung im Puffer vornehmen kann. Warten mehrere Prozesse auf ein Semaphor, bezeichnet man dies als *Serialisierung*. Eine weitere Operation, bei der ein Semaphor gesetzt wird, ist das Roll-in bzw. Roll-out. Dies bedeutet, dass immer nur ein Prozess nach dem anderen ein Roll-in bzw. Roll-out durchführen kann. Eine vollständige Liste der SAP-Semaphore erhalten Sie über die Werthilfe  zum Feld **Gesperrte Semaphore**.

Die globale Workprozess-Übersicht sollten Sie verwenden, wenn Sie ein System mit mehreren ABAP-Servern überwachen. Diese erreichen Sie entweder direkt über den Transaktionscode SM66 oder über SM50 • **Springen • Systemweite Liste**.

Mit der Schaltfläche **Aktive Workprozesse** schränken Sie die Selektion auf die aktiven Workprozesse ein, denn nur diese sind für die Analyse interessant.

### 2.2.2 Workprozess-Analyse durchführen

Wenn Sie die Workprozess-Übersicht für einige Minuten beobachten (Schaltfläche **Auffrischen**), können Sie relativ sicher feststellen, ob ein akutes Performanceproblem auf diesem ABAP-Server vorliegt, und dessen Ursache grob eingrenzen. Die Belegung aller Workprozesse eines Typs (d. h. Dialog, Verbuchung etc.) deutet auf ein solches Problem hin.

Wenn die Workprozess-Übersicht zahlreiche Workprozesse mit den Aktionen **Sequenzielles Lesen**, **Direktes Lesen**, **Update**, **Commit**, **Warten auf DB-Lock** oder anderen Datenbankaktivitäten zeigt, liegt ein Problem im Bereich der Datenbank vor. Öffnen Sie in diesem Fall zwei weitere Modi. Starten Sie den Datenbankprozessmonitor (siehe Abschnitt 2.4.3, »Identifizierung teurer SQL-Anweisungen«) und den Datenbanksperremonitor (Exclusive Lockwaits, siehe auch Abschnitt 2.4.4, »Weitere Analysen auf Datenbankebene«), um eventuell teure SQL-Anweisungen oder Datenbanksperren zu identifizieren.

Lange Datenbankzeiten

Ein spezielles Problem liegt vor, wenn alle Verbuchungs-Workprozesse (abgekürzt UPD vom englischen Update) belegt sind. Prüfen Sie, ob die Verbuchung deaktiviert wurde. Starten Sie dazu Transaktion SM13 und prüfen Sie, ob Sie die Information **Verbuchung ist deaktiviert** finden. Ist dies der Fall, existiert im SAP-Syslog (Transaktionscode SM21) ein Eintrag, aus dem hervorgeht, wann, von wem und aus welchem Grund die Verbuchung deaktiviert wurde. Lösen Sie das gemeldete Problem – dies kann z. B. ein Datenbankfehler sein – und aktivieren Sie die Verbuchung in Transaktion SM13 erneut.

Deaktivierte Verbuchung

Bei Problemen mit der SAP-Speicherkonfiguration trifft man in der Workprozess-Übersicht häufig auf die folgenden Situationen:

Probleme mit der SAP-Speicherkonfiguration

- Das Feld **Aktuelle Aktion** steht häufig auf **Roll In** oder **Roll Out** (auch in Verbindung mit dem Semaphor 6).
- Es befinden sich zahlreiche Workprozesse im **PRIV**-Modus (Feld **WP-Status** steht auf **hält**, Feld **Info Hält** steht auf **Backend-Session im PRIV-Modus**, siehe auch Abbildung 2.1).

#### Problem mit erschöpftem Extended Memory

Wenn Sie in der Workprozess-Übersicht sehen, dass sich praktisch alle Workprozesse im Zustand **PRIV** befinden, und im SAP-Speicherkonfigurationsmonitor feststellen, dass das Extended Memory vollständig erschöpft ist, dann ist das Extended Memory zu klein (oder der Speicherbedarf der Programme zu groß). Eine Vergrößerung des zu kleinen Extended Memorys



(Parameter `em/initial_size_MB`) oder eine Analyse der Programme mit hohem Speicherbedarf schafft in diesem Fall Abhilfe.

Über zwei Systemparameter können Sie Workprozesse beenden, die sich im **PRIV**-Modus befinden:

- `rdisp/max_priv_time`: Prozesse, die sich länger als die definierte Zeit im Zustand **PRIV** befinden, werden bei Erreichen des Zeitlimits beendet. Der Standardwert ist 0, d. h., Prozesse im **PRIV**-Modus besitzen mit diesem Wert kein Time-out.
- `rdisp/wppriv_max_no` definiert die maximale Anzahl an Prozessen, die sich im Zustand **PRIV** befinden dürfen. Bei Überschreitung dieses Limits wird der Prozess beendet, der sich am längsten im Zustand **PRIV** befindet.

**Haltende Prozesse** Wenn sich ein Prozess im Zustand **hält** befindet, liefert das Feld **Info Hält** die Ursache. Die Beschreibungen der einzelnen Haltegründe finden Sie in der Hilfe zu diesem Feld.

**Wartesituationen durch RFCs** Probleme mit Remote Function Calls (RFCs) erkennen Sie, wenn viele Workprozesse das Programm `SAPMSSY1` (RFC-Verwaltung) ausführen oder häufig der Wartegrund **RFC** oder **CPIC** auftritt. In diesem Fall ermitteln Sie anhand des ABAP-Codes, um welchen RFC-Aufruf es sich handelt. Anhand des Typs des Aufrufes und der Destination stellen Sie fest, ob es sich um einen Aufruf in ein entferntes System handelt oder um einen asynchrone RFC, der der Parallelisierung dient. Ist Ersteres der Fall, untersuchen Sie die Performance des aufgerufenen RFC-Bausteins auf dem entfernten System. Im zweiten Fall prüfen Sie, ob eine zu starke Parallelisierung zu einer Überlastung der Workprozesse führt. Weitere Informationen zur RFCs und Parallelisierung finden Sie in Kapitel 7, »Lastverteilung, Remote Function Calls und SAP GUI«.

**Wartesituationen durch SAP-Enqueues** Finden Sie in der Workprozess-Übersicht auffällig viele Prozesse, die das Programm `SAPLSENA` ausführen, deutet dies auf ein Problem mit der SAP-Enqueue-Verwaltung hin (siehe Kapitel 9, »Sperrungen«).

Im Prinzip ist es kein Problem, wenn sich einzelne Prozesse kurzzeitig im Status **hält** befinden. Übersteigt die Zahl der aus dem gleichen Grund haltenden Prozesse jedoch die 20 %-Marke oder befinden sich einzelne Prozesse lange Zeit in diesem Zustand, sollte das Problem detaillierter analysiert werden. Gerade bei haltenden Workprozessen ist oft ein einziger, leistungsschwacher oder gar fehlerhaft arbeitender Workprozess die Ursache dafür, dass in einer Kettenreaktion auch andere Workprozesse zum Halten gezwungen werden. Den fehlerhaft arbeitenden Prozess finden Sie anhand

des Feldes **Dauer**. In der Regel können Sie davon ausgehen, dass der Prozess mit der längsten Laufzeit das Problem verursacht hat. Oft kann das akute Problem dadurch behoben werden, dass der fehlerhafte Workprozess manuell beendet wird.

Wenn Sie in der Workprozess-Übersicht zahlreiche abgebrochene Workprozesse beobachten (Eintrag **beendet** im Feld **WP-Status**), die sich nicht wieder starten lassen, besteht höchstwahrscheinlich ein Problem mit dem SAP-Kernel oder bei der Anmeldung an die Datenbank. Überprüfen Sie in diesem Fall die Trace-Datei. Diese finden Sie in der Workprozess-Übersicht unter der Menüoption **Prozess • Trace • Anzeigen**. Sichern Sie diese Trace-Datei in einer lokalen Datei. Dies ist für die anschließende Fehlersuche wichtig, da die Trace-Datei beim Neustart des Workprozesses überschrieben wird und die Information über den Grund des Abbruchs damit für die Fehlersuche verloren ist. Suchen Sie im SAP Support Portal nach Hinweisen zu diesem Problem oder schalten Sie SAP bei der Fehlersuche ein.

Beendete Prozesse

In einem verteilten System mit mehreren Rechnern kann es dazu kommen, dass Sie auf mindestens einem Rechner einen Workprozess-Engpass feststellen, während andere Rechner noch über wartende, d. h. freie, Workprozesse verfügen. Untersuchen Sie in diesem Fall, wie viele Benutzer sich an den einzelnen ABAP-Servern angemeldet haben. Im Workload-Monitor (siehe Kapitel 3, »Workload-Analyse«) können Sie darüber hinaus feststellen, wie viele Dialogschritte auf den einzelnen Servern durchgeführt wurden. Stellen Sie eine extrem ungleichmäßige Lastverteilung fest, ist Ihre Log-on-Verteilung nicht richtig eingestellt. Sie sollten in Transaktion SMLG überprüfen, ob alle Server für die Log-on-Verteilung zur Verfügung stehen oder ob Fehler gemeldet werden. Optimieren Sie gegebenenfalls mithilfe der Transaktion SMLG Ihre Log-on-Verteilung.

Falsche  
Lastverteilung

Die aktuelle Benutzerverteilung finden Sie in der Anmeldegruppenverteilung unter diesem Menüpfad: **Werkzeuge • CCMS • Konfiguration • Anmeldegruppen** (Transaktionscode SMLG) • **Springen • Lastverteilung**.

Wie aus dem bisher Beschriebenen hervorgeht, kann es zahlreiche Gründe geben, weshalb alle Workprozesse eines Typs belegt sein können. Wenn Sie allerdings die bisher diskutierten Probleme ausschließen können und trotzdem ein Problem im Bereich der Workprozesse beobachten, kann es sein, dass Sie zu wenige Workprozesse konfiguriert haben. In diesem Fall sollten Sie die Anzahl der Workprozesse vergrößern. Vorher sollten Sie allerdings überprüfen, ob der Rechner über ausreichend Reserven an CPU und Hauptspeicher verfügt. Wenn die CPU schon zu 80 % ausgelastet ist, wird die Erhöhung der Anzahl der Workprozesse nicht zu einer Performanceverbesserung führen.

Zu wenige  
Workprozesse

**Weitere Informationen**

In Tabelle 2.2 finden Sie Verweise auf die Kapitel und Abschnitte dieses Buches, die die anhand der Workprozess-Übersicht identifizierten Probleme besprechen.

Problem	Kapitel oder Abschnitt	Titel/Thema
lange Datenbankantwortzeiten	2.4.3, 9, 10, 11, 12, 13	Identifizierung teurer SQL-Anweisungen
	9	Datenbanksperren
	10, 11, 12, 13	Optimierung von SQL-Anweisungen
SAP-Speicherkonfiguration	2.5	Analyse der SAP-Speicherkonfiguration
	6	Speicherkonfiguration
falsche Lastverteilung, zu wenige Workprozesse	7.2	Lastverteilung
Probleme mit Remote Function Calls (RFCs)	7.3	Remote Function Calls

**Tabelle 2.2** Kapitel und Buchabschnitte, die die in der Workprozess-Übersicht erkannten Probleme behandeln

**2.2.3 Überwachung der Dispatcher-Queue**

**Dispatcher-Statistik**

Gelegentlich kann es sinnvoll sein, die Queue des Workprozess-Dispatchers zu analysieren. Eine Statistik über die Dispatcher-Aktivität ist über die Serverübersicht verfügbar: **Werkzeuge • Administration • Monitor • Systemüberwachung • Server** (Transaktionscode SM51).

Markieren Sie anschließend einen ABAP-Server mit dem Cursor und wählen Sie **Springen • Informationen • Queue • Wartende Requests**. In der folgenden Liste werden für jeden Workprozess-Typ Informationen über die aktuell wartenden Anfragen, den Höchststand wartender Anfragen seit Start des ABAP-Servers, die maximal mögliche Anzahl der Anfragen pro Queue sowie die Anzahl der geschriebenen und der gelesenen Anfragen dargestellt.

Die Information über die Dispatcher-Queue ist insbesondere dann von Bedeutung, wenn das System »steht«, weil die Anzahl der Aufträge in der Queue deutlich größer ist als die Anzahl der Workprozesse und im SAP-System keine Workprozesse mehr zur Analyse zur Verfügung stehen. In diesem Fall kann die Queue-Info mit dem Hilfsprogramm `dpmmon` aufgerufen werden.

### 2.2.4 Workprozess-Stichprobenanalyse (/SDF/SMON)

Ein System durch wiederholtes Auffrischen der Workprozess-Übersicht zu überwachen und zu analysieren ist eine eintönige und zeitaufwendige Arbeit. Sie können diese Aufgabe jedoch dem System überlassen. Dazu dient die *Workprozess-Stichprobenanalyse* (Transaktionscode /SDF/SMON). Sie ruft die Workprozess-Übersicht regelmäßig im Hintergrund auf und speichert die Daten in Form von Stichproben (Snapshots), die Sie dann später systematisch auswerten können. Die Stichproben werden in sogenannten Stichprobensammlungen gespeichert. So umfasst typischerweise eine Stichprobensammlung die Stichproben eines Tages.

Die Idee hinter diesem Werkzeug ist, dass alle lang laufenden oder sehr häufig ausgeführten Aktionen durch die Stichprobenanalyse »erwischt«, werden und Sie damit einen systematischen Überblick über die Systemnutzung bekommen.

Um die Workprozess-Stichprobenanalyse zu konfigurieren, gehen Sie wie folgt vor:

**Stichprobenanalyse konfigurieren**

1. Starten Sie die Workprozess-Stichprobenanalyse mit dem Transaktionscode /n/SDF/SMON.
2. Wenn es noch keine Stichprobensammlungen gibt, wird Ihnen direkt der Konfigurationsbildschirm **Snapshot-Monitoring starten** angezeigt. Wenn schon Stichprobensammlungen vorliegen, dann wird Ihnen eine Liste der Sammlungen angezeigt. In diesem Fall gelangen Sie über **Springen • Neues Monitoring** zum Konfigurationsbildschirm.
3. Definieren Sie auf dem Konfigurationsbildschirm den Analysezeitraum, die Server oder `saposcol`-Instanzen, auf denen die Analyse laufen soll, und den Inhalt. Des Weiteren können Sie den Zeitrahmen angeben, nach dem Daten wieder gelöscht werden. Als Basisintervall, das definiert, in welchem Abstand die Stichproben erstellt werden sollen, empfehlen wir 1 Sekunde. Weitere Details, wie z. B. die Speicherverwendung, können etwa alle 10 Sekunden ermittelt werden. Sie können ebenfalls angeben, dass die CPU- und Paging-Daten aus dem Betriebssystemmonitor erfasst werden sollen. Als Standard empfehlen wir, eine Stichprobensammlung pro Tag zu erstellen. Sie können dies einfach über die Schaltfläche **Schedule Daily Monitoring** aktivieren. Weitere Details zur Konfiguration finden Sie in der in Anhang D, »Informationsquellen«, aufgeführten Dokumentation.

### Stichprobenanalyse auswerten

Um die Workprozess-Stichprobenanalyse auszuwerten, gehen Sie wie folgt vor:

1. Geben Sie den Transaktionscode /n/SDF/SMON ein. Wenn schon Stichproben vorhanden sind, dann wird Ihnen eine Liste der Stichprobensammlungen angezeigt.
2. Öffnen Sie eine der Sammlungen. In einem Auswahlbildschirm geben Sie an, welche Proben der Sammlung Sie sehen wollen. Folgende Selektionen können sinnvoll sein:
  - Wählen Sie die Option **Nur anzeigen alle 30 Snapshots** (im unteren Bereich des Bildschirms), um einen Überblick über den analysierten Zeitraum zu gewinnen.
  - Schränken Sie alternativ einen oder mehrere Messwerte ein. Wählen für Sie z. B. für den Wert der Dialog-Queue-Länge eine Schranke von größer 50, um auf Zeiträume einzuschränken, in denen eine gewisse Zahl von Anfragen im Dispatcher warteten. In diesem Fall können Sie im unteren Bereich des Bildschirms zusätzlich die Option **Anzeigen auch benachbarte 3 Snapshots** auswählen.

Wählen Sie anschließend **Ausführen**.

3. Das System zeigt Ihnen nun die Liste der Proben an, auf die die gewählten Einschränkungen zutreffen (siehe Abbildung 2.2). Sie können nun durch die Liste blättern oder nach Messwerten sortieren.
4. Wählen Sie zunächst einen Eintrag, den Sie näher betrachten möchten, und klicken Sie dann auf das Zahnradsymbol  (**Workprozess-Übersicht**). Das System zeigt Ihnen daraufhin die Liste der Workprozesse zum Zeitpunkt der Probenerstellung. Der Bildschirm ist derselbe, als ob Sie zu diesem Zeitpunkt die Workprozess-Übersicht (siehe Abbildung 2.1) aufgerufen hätten. Von einem Workprozess-Snapshot ist die Navigation in den zugehörigen Statistiksatz (siehe Abschnitt 3.2, »Einzelsatzstatistik«) oder ins ABAP-Coding des ausgeführten Programms möglich.

Im Falle einer Überlastsituation können Sie mit der Workprozess-Stichprobenanalyse sekundengenau die Situation im System verfolgen. Sie bekommen einen Einblick in die Hardwaredaten (CPU und Hauptspeicher), die Workprozesse und die Datenbankaktivitäten, wenn sie in der Workprozess-Übersicht zur Verfügung stehen.

Ein Ziel der Analyse ist es, Zeiträume mit einer Überlastungssituation herauszufiltern und diese zu bewerten. Insbesondere müssen Sie bewerten, ob Benutzer (im Sinne von Personen, die vor ihren Bildschirmen warten) von der Überlastungssituation betroffen sind oder ob die Überlastsituation ausschließlich technische Benutzer (d. h. Schnittstellen- oder Hintergrundver-

arbeitung) betrifft. In diesem Fall ist es vielleicht sogar gewünscht, dass das System vollständig ausgelastet ist.

Time	AS	Act. WPs	Dia.WPs	RFC Normal	RFC Low	CPU Usr	CPU Sys	CPU Idle	CPU	Ava.	Rea.	Ste.	Paging in	Paging out	F
01:19:04	L	197	110	118	38	73	6	21	63.20	80.00	0.0	0	0	0	1
01:19:05		220	134	118	23	73	6	21	63.20	80.00	0.0	0	0	0	1
01:19:06		202	116	118	36	73	6	21	63.20	80.00	0.0	0	0	0	1
01:19:07		237	150	118	16	73	6	21	63.20	80.00	0.0	0	0	0	1
01:19:09		233	147	102	8	73	6	21	63.20	80.00	0.0	0	0	0	1
01:19:10		195	109	118	32	73	6	21	63.20	80.00	0.0	0	0	0	1
01:19:11		174	88	118	66	73	6	21	63.20	80.00	0.0	0	0	0	1
01:20:57		185	100	115	43	72	6	22	62.40	80.00	0.0	0	0	0	1
01:20:59		157	71	118	68	72	6	22	62.40	80.00	0.0	0	0	0	1
01:21:00		170	86	118	66	72	6	22	62.40	80.00	0.0	0	0	0	1
01:21:01		182	98	118	54	72	6	22	62.40	80.00	0.0	0	0	0	1
01:21:02		183	95	117	53	72	6	22	62.40	80.00	0.0	0	0	0	1
01:21:03		172	86	118	59	72	6	22	62.40	80.00	0.0	0	0	0	1
01:21:04		177	90	118	62	72	6	22	62.40	80.00	0.0	0	0	0	1

Abbildung 2.2 Workprozess-Stichprobenanalyse (Transaktion /SDF/SMON)

### 2.2.5 Workprozess-Stichprobenanalyse (technisches Monitoring)

Die Monitoring-Sicht **Sampled Work Process Data** im technischen Monitoring ist eine Alternative zur SAP-GUI-Anwendung Workprozess-Stichprobenanalysen (/SDF/SMON), die im vorherigen Abschnitt beschrieben wurde.

Die Rohdaten der Anwendung sind hier ebenfalls Workprozess-Stichproben. Die Erstellung der Stichproben funktioniert allerdings anders als in der Transaktion /SDF/SMON. Inhaltlich entsprechen sich die erfassten Daten aber. Die Workprozess-Stichprobenanalyse im technischen Monitoring ist Teil der Basis und nicht wie der Monitor /SDF/SMON Teil des ST-PI-Add-ons (Solution Tools Plug-in).

Funktionsweise

Die Monitoring-Sicht bietet eine grafische Aufbereitung mit der Möglichkeit einer flexiblen Selektion der Daten und eines Eintauchens (Drill-down) in die Daten.



Abbildung 2.3 Monitoring-Sicht »Sampled Work Process Data« im technischen Monitoring

**Beschreibung der Anwendung**

Starten Sie die Monitoring-Sicht **Sampled Work Process Data** im SAP Fiori Launchpad unter **Administration • Technisches Monitoring • Workprozess-Stichprobendaten**. Alternativ wählen Sie im Navigationsmenü des technischen Monitorings den Eintrag **Information • Workload • Sampled Work Process Data**. Der Bildschirm **Sampled Work Process Data**, den Abbildung 2.3 zeigt, ist in drei größere Bereiche eingeteilt:

- 1 Im mittleren Hauptteil des Bildschirms sehen Sie eine Darstellung der gewählten Kennzahl als Funktion der Zeit als Balkengrafik.
- 2 Im oberen Bereich des Bildschirms finden Sie einen Bereich, in dem Sie den Analysezeitraum einschränken können.
- 3 Weiterhin können Sie im oberen Bereich die Balkengrafik konfigurieren. Dazu stehen folgende Optionen zur Verfügung:
  - Über die Schaltflächen **ABAP CPU Time**, **Main Memory**, **Dialog Work Process Utilization** und **Background Work Process Utilization** legen Sie fest, welche Kennzahlen Sie sehen wollen. In der Auswahlfeld **Drill Down** können Sie wählen, nach welcher Charakteristik die Balken unterteilt werden sollen.
  - Ihnen stehen die fünf Filter **Transaction Code**, **Request Entry Type**, **Request Entry Name**, **Tenant ID** und **WM Process Class ID** zur Verfüg-

gung, um die Selektion der Workprozess-Stichproben einzuschränken.

- ④ Der untere Bereich des Bildschirms enthält eine Liste der Workprozess-Stichproben. Jede Zeile ist eine Aggregation der Stichproben zu einer Anwendung im gewählten Zeitraum und den gewählten Filterbedingungen. Wenn Sie eine Zeile auswählen, zeigt die Anwendung im Detailbildschirm die einzelnen Stichproben an.

Mit der Monitoring-Sicht **Sampled Work Prozess Data** können Sie die in Abschnitt 2.2.2, »Workprozess-Analyse durchführen«, vorgestellten Analysen durchführen. Die folgende Anleitung gibt Ihnen einige Hinweise zur Anwendung:

**Analyse durchführen**

1. Abhängig von der konkreten Situation nehmen Sie folgende Vorselektion vor:
  - Wenn Ihnen konkrete Berichte über Performanceprobleme in einem bestimmten Zeitraum oder zu einer bestimmten Anwendung vorliegen, schränken Sie den Analysezeitraum und die Datenselektion entsprechend ein, um die Analyse zu beginnen.
  - Liegen Ihnen keine konkreten Informationen vor, selektieren Sie einen Analysezeitraum beispielsweise abhängig von Ihren erwarteten Hauptnutzungszeiten.
2. Wählen Sie zunächst die Option **ABAP CPU Time**, wie auch in Abbildung 2.3 dargestellt. Neben der ABAP-CPU-Zeit, die von den Workprozessen ermittelt wird, wird auch die CPU-Auslastung des entsprechenden Applikationsservers angezeigt. Sie können also auf einen Blick feststellen, ob eine hohe CPU-Zeit zu einer hohen CPU-Auslastung führt.
3. Analysieren Sie nun, ob es zu Lastspitzen mit hoher CPU-Auslastung gekommen ist. In Abbildung 2.3 erkennen Sie beispielsweise, dass von 6 bis 9 Uhr abends eine CPU-Auslastung bei 100 % vorliegt. Wählen Sie einen Zeitpunkt aus, den Sie als Nächstes analysieren wollen, und selektieren Sie den entsprechenden Balken. Im unteren Teil des Bildschirms können Sie nun die Anwendungen sehen, die zur Last beigetragen haben.
4. Anhand des Wertes von **ABAP CPU Time Since Last Sampling** identifizieren Sie die teuersten Anwendungen.

Haben Sie entschieden, dass Sie eine konkrete Anwendung analysieren wollen, selektieren Sie die Anwendung im unteren Teil des Bildschirms. Das System zeigt Ihnen einen Detailbildschirm mit allen Workprozess-Stichproben zur Anwendung im ausgewählten Zeitraum. Führen Sie nun die in Abschnitt 2.2.2 vorgestellten Analysen durch.

**Drill-down in die Statistiksätze**

Alternativ können Sie die Analyse auch mit der Option **Dialog Work Process Utilization** starten, um Zeiträume mit einer hohen Auslastung der Workprozesse genauer unter die Lupe zu nehmen.

### 2.2.6 Zusammenfassung

Workprozesse  
überwachen

Mit der Workprozess-Übersicht können Sie anhand der Felder **Dauer**, **WP-Status**, **Info Hält**, **Aktuelle Aktion** und **Info Aktion** die Aktionen der Workprozesse überwachen. Folgende Punkte sollten Sie beachten:

- Sind auf jedem ABAP-Server ausreichend freie Workprozesse (Status **wartet**) jeden Typs vorhanden?
- Gibt es Programme, die einen Workprozess sehr lange belegen (Feld **Dauer**)? In diesem Fall sollten Sie die Benutzer auf dieses Programm ansprechen und klären, ob das Programm fehlerfrei arbeitet. Eventuell sollten Sie es für eine detaillierte Programmanalyse vorsehen.
- Prüfen Sie anhand der Felder **WP-Status**, **Info Hält**, **Aktuelle Aktion** und **Info Aktion**, ob mehr als 20 % der Workprozesse die gleiche Aktion ausführen. Die wichtigsten Probleme sind dabei:
  - Befinden sich mehr als 20 % der Workprozesse im **PRIV**-Modus oder beim Roll-in oder Roll-out, deutet dies auf ein Problem mit der SAP-Speicherkonfiguration hin.
  - Befinden sich mehr als 40 % der Workprozesse in einer Datenbankaktion wie **Sequenzielles Lesen**, **Commit** etc., deutet dies auf ein Datenbankproblem hin.
  - Lesen mehr als 20 % der Workprozesse gleichzeitig von derselben Tabelle, deutet dies auf ein Problem mit einer teuren SQL-Anweisung, eventuell aber auch auf eine Sperrsituation in der Datenbank (*Exclusive Lockwaits*) hin.

Die Workprozess-Übersicht kann auf Betriebssystemebene mithilfe des Programms `dpmon` aufgerufen werden. Dies ist insbesondere dann nötig, wenn das Performanceproblem bereits derart massiv ist, dass keine Workprozesse für die Analyse herangezogen werden können.

Mithilfe der Workprozess-Stichprobenanalyse können Sie regelmäßige Stichproben der Workprozess-Übersicht nehmen und diese später systematisch auswerten.

Abbildung 2.4 zeigt Ihnen schematisch das Analyseverfahren im Bereich der Workprozesse.

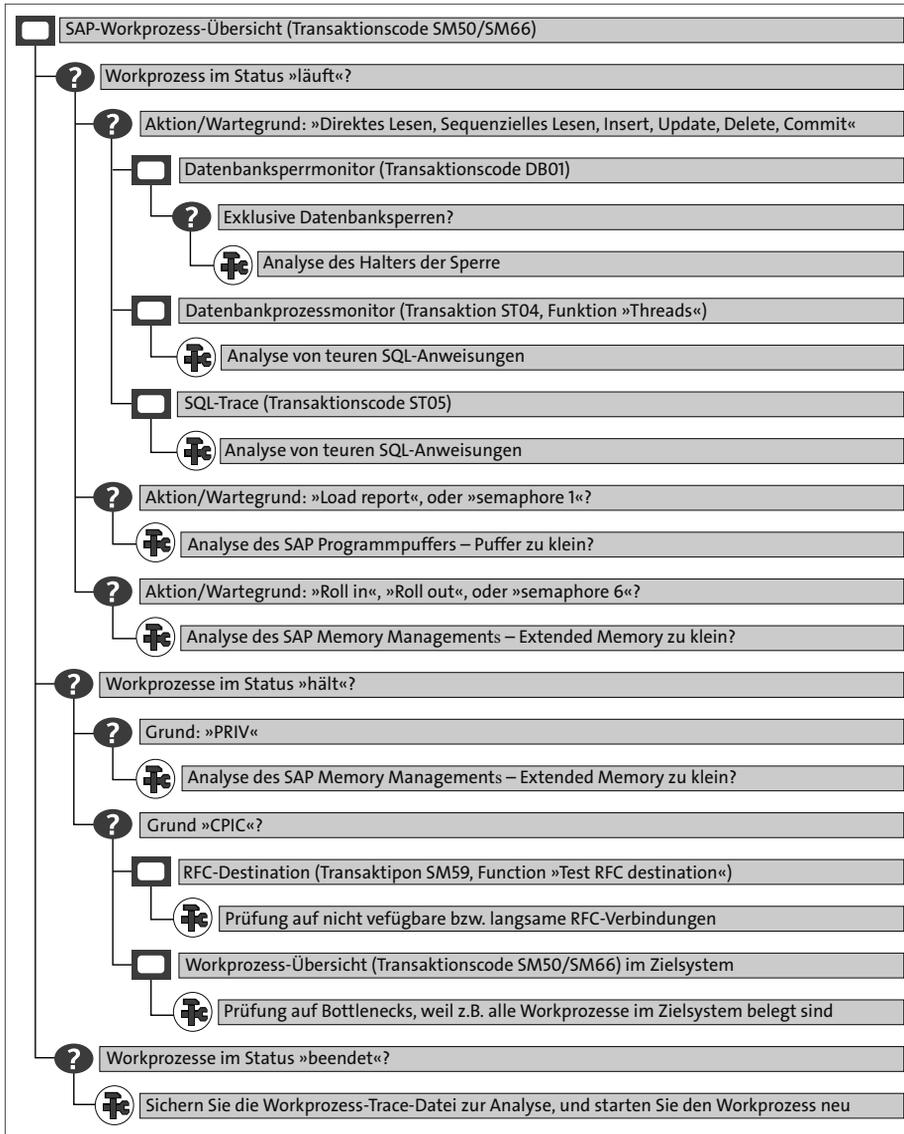


Abbildung 2.4 Detailanalyse der Workprozesse

## 2.3 Hardwareanalyse

Das Werkzeug zur Analyse von Hardwareengpässen und Betriebssystemproblemen ist der Betriebssystemmonitor. Um diesen Monitor für den Applikationsserver zu starten, auf dem Sie zurzeit angemeldet sind, wählen Sie: **Werkzeuge • Administration • Monitor • Performance • Betriebssystem • Lokal • Betriebssystemmonitor**. Alternativ können Sie den Transaktions-

**Betriebssystem-  
monitor**

# Inhalt

Vorwort und Danksagung .....	17
Einleitung .....	19

## **1 Performancemanagement einer SAP-Lösung** 33

---

<b>1.1 Die Architektur von SAP S/4HANA</b> .....	33
1.1.1 SAP S/4HANA .....	34
1.1.2 Client-Server-Architektur .....	41
<b>1.2 Grundlegende Begriffe und Aufgaben der Performanceoptimierung</b> .....	48
1.2.1 Die Parameter der Performanceoptimierung .....	48
1.2.2 Aufgaben, Rollen und Werkzeuge der Performanceoptimierung .....	52
<b>1.3 Das Überwachungs- und Optimierungskonzept für eine SAP-Lösung</b> .....	57
1.3.1 Anforderungen an ein Überwachungs- und Optimierungskonzept .....	57
1.3.2 Service Level Management .....	60
1.3.3 Werkzeuge für die Performanceüberwachung und Analyse .....	66
<b>1.4 Zusammenfassung</b> .....	72

## **2 Analyse von Hardware, Datenbank und ABAP-Applikationsserver** 75

---

<b>2.1 Begriffsklärungen</b> .....	76
<b>2.2 Analyse der Workprozesse</b> .....	77
2.2.1 Felder der Workprozess-Übersicht .....	77
2.2.2 Workprozess-Analyse durchführen .....	81
2.2.3 Überwachung der Dispatcher-Queue .....	84
2.2.4 Workprozess-Stichprobenanalyse (/SDF/SMON) .....	85

2.2.5	Workprozess-Stichprobenanalyse (technisches Monitoring) .....	87
2.2.6	Zusammenfassung .....	90
<b>2.3</b>	<b>Hardwareanalyse</b> .....	<b>91</b>
2.3.1	Analyse eines Hardwareengpasses (CPU und Hauptspeicher) .....	93
2.3.2	Identifizierung von Schreib-/Lese-(I/O-)Problemen .....	99
2.3.3	Parameteränderungen und Netzwerkcheck .....	100
2.3.4	Zusammenfassung .....	101
<b>2.4</b>	<b>Datenbankanalyse</b> .....	<b>103</b>
2.4.1	Performancemonitor im DBA-Cockpit .....	103
2.4.2	Analyse des Datenbank Hauptspeichers .....	106
2.4.3	Identifizierung teurer SQL-Anweisungen .....	110
2.4.4	Weitere Analysen auf Datenbankebene .....	124
2.4.5	Zusammenfassung .....	130
<b>2.5</b>	<b>Analyse der SAP-Speicherkonfiguration</b> .....	<b>131</b>
2.5.1	Analyse der SAP-Puffer .....	132
2.5.2	Analyse des Extended Memorys und des Heap Memorys .....	134
2.5.3	Anzeige des allokierten Speichers .....	136
2.5.4	Weitere Monitore zur detaillierten Analyse .....	138
2.5.5	Zusammenfassung .....	139
<b>2.6</b>	<b>Zusammenfassung</b> .....	<b>140</b>
<b>2.7</b>	<b>Übungsfragen</b> .....	<b>141</b>
<b>3</b>	<b>Workload-Analyse</b> .....	<b>143</b>
<hr/>		
<b>3.1</b>	<b>Einführung in die Workload-Analyse</b> .....	<b>144</b>
<b>3.2</b>	<b>Einzelsatzstatistik</b> .....	<b>146</b>
<b>3.3</b>	<b>Details zur Workload-Analyse</b> .....	<b>151</b>
3.3.1	Ablauf eines Anwendungsschrittes .....	151
3.3.2	Weitere Zeitkomponenten .....	155
3.3.3	Interpretation der Antwortzeiten .....	156
3.3.4	Aktivität, Durchsatz und Last .....	159
<b>3.4</b>	<b>Workload-Monitor</b> .....	<b>161</b>
3.4.1	Mit dem Workload-Monitor arbeiten .....	161
3.4.2	Technische Einstellungen zum Workload-Monitor .....	164
3.4.3	Profile im Workload-Monitor .....	165

<b>3.5</b>	<b>Workload-Analyse durchführen</b> .....	172
3.5.1	Liegt ein allgemeines Performanceproblem vor? .....	173
3.5.2	Liegt ein temporäres oder ein dauerhaftes Performanceproblem vor? .....	174
3.5.3	Liegt ein Hardwareengpass auf einem Rechner vor? .....	176
3.5.4	Liegt ein allgemeines Datenbankperformanceproblem vor? .....	177
3.5.5	Liegt ein Performanceproblem aufgrund falscher Lastverteilung vor? .....	178
3.5.6	Liegt ein Performanceproblem mit der SAP-Speicherverwaltung vor? .....	179
3.5.7	Liegt ein Performanceproblem mit einer Anwendung vor? .....	180
3.5.8	Liegt ein Netzwerkproblem zwischen Präsentationsserver und Applikationsserver vor? .....	181
<b>3.6</b>	<b>Workload-Analyse mit dem technischen Monitoring</b> .....	182
<b>3.7</b>	<b>Zusammenfassung</b> .....	188
<b>3.8</b>	<b>Übungsfragen</b> .....	190

## **4 Hardware-Sizing und Komponentenverteilung** 193

---

<b>4.1</b>	<b>Initiales Hardware-Sizing</b> .....	193
4.1.1	Übersicht über das Projekt zum initialen Sizing .....	195
4.1.2	Sizing-Methoden .....	200
4.1.3	Sizing-Projekt im Detail durchführen .....	205
4.1.4	SAP Standard Application Benchmarks .....	209
<b>4.2</b>	<b>Sizing bei Laststeigerung, Versionswechsel oder Migration</b> .....	212
4.2.1	Übersicht über ein Sizing-Projekt im Umfeld einer bereits produktiven Installation .....	213
4.2.2	Sizing im Umfeld produktiver Installationen im Detail durchführen .....	214
4.2.3	SAP-HANA-Migration .....	219
<b>4.3</b>	<b>Systemlandschaft planen</b> .....	221
4.3.1	Skalierung der Datenbankebene .....	222
4.3.2	Skalierung der Applikationsebene .....	225

<b>4.4</b>	<b>Zusammenfassung</b> .....	226
<b>4.5</b>	<b>Übungsfragen</b> .....	227
<b>5</b>	<b>Optimierung von ABAP-Programmen</b> .....	229
<hr/>		
<b>5.1</b>	<b>Performance-Trace</b> .....	229
5.1.1	Performance-Trace erstellen .....	230
5.1.2	SQL-Trace auswerten .....	233
5.1.3	Puffer-Trace auswerten .....	243
5.1.4	RFC-Trace auswerten .....	244
5.1.5	HTTP-Trace auswerten .....	244
5.1.6	Enqueue-Trace auswerten .....	245
<b>5.2</b>	<b>Performanceanalyse mit dem ABAP-Trace (Laufzeitanalyse)</b> .....	246
5.2.1	ABAP-Trace erstellen .....	247
5.2.2	ABAP-Trace auswerten .....	253
5.2.3	Zeitachsensicht verwenden .....	255
<b>5.3</b>	<b>Analyse des Speicherverbrauchs mit dem ABAP Debugger und im Memory Inspector</b> .....	257
<b>5.4</b>	<b>Statische Codeprüfungen mit dem ABAP Test Cockpit</b> .....	262
<b>5.5</b>	<b>Tipps und Tricks für performante ABAP-Programme</b> .....	266
<b>5.6</b>	<b>Zusammenfassung</b> .....	273
<b>5.7</b>	<b>Übungsfragen</b> .....	274
<b>6</b>	<b>Speicherkonfiguration</b> .....	277
<hr/>		
<b>6.1</b>	<b>Grundlagen der Speicherkonfiguration</b> .....	277
6.1.1	Begriffsklärungen .....	277
6.1.2	Speicherbereiche der ABAP-Instanz .....	279
6.1.3	Zero Administration Memory Management .....	282
6.1.4	Benutzerquoten .....	282
6.1.5	Technische Realisierung im Detail .....	285
6.1.6	Zusammenfassung .....	293
<b>6.2</b>	<b>Speicherbereiche konfigurieren und überwachen</b> .....	294
6.2.1	Auslagerungsspeicher überwachen .....	294

6.2.2	Speicherbereiche konfigurieren und überwachen .....	296
6.2.3	Hilfe zur Fehlerbehebung .....	299
<b>6.3</b>	<b>Zusammenfassung .....</b>	<b>306</b>
<b>6.4</b>	<b>Übungsfragen .....</b>	<b>308</b>

## **7 Lastverteilung, Remote Function Calls und SAP GUI** 311

---

<b>7.1</b>	<b>Services des ABAP-Servers .....</b>	<b>312</b>
<b>7.2</b>	<b>Lastverteilung innerhalb der ABAP-Instanzen .....</b>	<b>314</b>
7.2.1	Message- und Enqueue- Service verteilen .....	316
7.2.2	Dialog-, Hintergrund- und Spool-Workprozesse verteilen	317
7.2.3	Benutzer und Workprozesse auf CPU-Ressourcen verteilen .....	318
7.2.4	Dynamische Benutzerverteilung: Anmeldegruppen konfigurieren .....	320
7.2.5	Lastverteilungskonzept für Dialog-Workprozesse .....	322
7.2.6	Ressourcen pro Benutzer beschränken .....	325
7.2.7	Betriebsarten .....	326
7.2.8	Dynamische Workprozesse konfigurieren .....	327
7.2.9	Hintergrundverarbeitung .....	327
7.2.10	Verbuchung .....	328
<b>7.3</b>	<b>Remote Function Calls .....</b>	<b>336</b>
7.3.1	Grundlagen und Begriffe .....	336
7.3.2	RFC-Typen .....	339
7.3.3	Ablauf eines RFCs .....	341
7.3.4	Parallelisierung von Programmen .....	344
7.3.5	RFC-Verbindungen konfigurieren und testen .....	346
7.3.6	Serialisierung auswählen .....	350
7.3.7	Eingehende und ausgehende Last überwachen .....	351
7.3.8	Parallelisierung von Prozessen mit asynchronen RFCs konfigurieren .....	357
7.3.9	Datenübertragung mit transaktionalen und Hintergrund-RFCs überwachen .....	358
<b>7.4</b>	<b>SAP GUI .....</b>	<b>360</b>
7.4.1	Interaktionsmodell und Performancemessung .....	360
7.4.2	Performance der GUI-Kommunikation analysieren und optimieren .....	363

<b>7.5</b>	<b>Zusammenfassung</b> .....	367
<b>7.6</b>	<b>Übungsfragen</b> .....	370
<b>8</b>	<b>Internetanbindung und SAP Fiori</b> .....	373
<hr/>		
<b>8.1</b>	<b>SAP-Webanwendungen</b> .....	374
8.1.1	Webservices und der Internet Communication Manager .....	375
8.1.2	Einsatz von Web UI und SAP GUI planen .....	378
<b>8.2</b>	<b>Analysewerkzeuge für Webanwendungen</b> .....	382
8.2.1	Performancewerkzeuge der Internetbrowser .....	386
8.2.2	SAP-Statistiken in der HTTP-Anfrage .....	387
8.2.3	Webanwendungen kontinuierlich überwachen .....	389
<b>8.3</b>	<b>Pufferung von Webdokumenten</b> .....	390
8.3.1	Browserpuffer (Browsercache) .....	390
8.3.2	Puffer im ICM und im SAP Web Dispatcher .....	393
8.3.3	Content Delivery Network .....	395
8.3.4	Zusammenfassung: Nutzung von Puffern in Webanwendungen .....	396
<b>8.4</b>	<b>Business Server Pages, Web Dynpro ABAP und integrierter ITS</b> .....	396
8.4.1	Business Server Pages und Web Dynpro ABAP .....	397
8.4.2	SAP GUI for HTML und integrierter ITS .....	399
8.4.3	Performanceanalyse für eine Web-GUI-Anwendung durchführen .....	400
<b>8.5</b>	<b>SAP Fiori, SAPUI5 und OData-Services</b> .....	401
8.5.1	Grundlagen von SAP Fiori, SAPUI5 und OData .....	401
8.5.2	Aufbau einer SAP-Fiori-Systemlandschaft .....	404
8.5.3	SAP-Gateway-Statistiken .....	407
8.5.4	Performance von SAP Fiori, SAPUI5 und OData optimieren .....	409
8.5.5	Pufferung von SAPUI5- und OData-Metadatendokumenten .....	413
8.5.6	Performanceanalyse für eine SAP-Fiori-Anwendung durchführen .....	417
<b>8.6</b>	<b>Zusammenfassung</b> .....	419
<b>8.7</b>	<b>Übungsfragen</b> .....	421

---

## 9 Sperren 423

---

<b>9.1 Sperrkonzepte von Datenbanksystem und SAP-System .....</b>	<b>424</b>
9.1.1 Datenbanksperren .....	424
9.1.2 Enqueues .....	425
9.1.3 Lesekonsistenz .....	427
<b>9.2 Überwachung von Sperren .....</b>	<b>429</b>
9.2.1 Datenbanksperren .....	429
9.2.2 Enqueues .....	434
9.2.3 Lesekonsistenz .....	436
<b>9.3 Nummernkreispuﬀerung .....</b>	<b>437</b>
9.3.1 Grundlagen .....	437
9.3.2 Nummernkreispuﬀerung einschalten .....	442
9.3.3 Nummernkreispuﬀerung überwachen .....	443
<b>9.4 Zusammenfassung .....</b>	<b>444</b>
<b>9.5 Übungsfragen .....</b>	<b>445</b>

## 10 Optimierung von SQL-Anweisungen 447

---

<b>10.1 Teure SQL-Anweisungen identifizieren und analysieren .....</b>	<b>449</b>
10.1.1 Identifizierung und Voranalyse .....	449
10.1.2 Klassifizierung von teuren SQL-Anweisungen .....	453
<b>10.2 Die fünf goldenen Regeln der performanten SQL-Programmierung .....</b>	<b>457</b>
<b>10.3 SQL-Anweisungen im Programm optimieren .....</b>	<b>465</b>
10.3.1 Beispiel für die Optimierung eines ABAP-Programms .....	465
10.3.2 Vorbelegung von Feldern in Reporttransaktionen .....	473
<b>10.4 SQL-Anweisungen durch Sekundärindizes optimieren .....</b>	<b>479</b>
10.4.1 Ein wenig Theorie .....	479
10.4.2 Indizes anlegen .....	490
10.4.3 Allgemeine Regeln zum Anlegen oder Ändern von Sekundärindizes .....	494
10.4.4 Regeln für Sekundärindizes im SAP HANA Columns Store .....	497
10.4.5 Regeln für Sekundärindizes im SAP HANA Row Store und auf klassischen Datenbanken .....	500
10.4.6 Sekundärindizes bei der SAP-HANA-Migration .....	505

<b>10.5 Zusammenfassung</b> .....	507
<b>10.6 Übungsfragen</b> .....	510

## **11 Pufferung auf dem Applikationsserver** 513

---

<b>11.1 Pufferungsarten</b> .....	514
<b>11.2 Grundlagen der Tabellenpufferung</b> .....	515
11.2.1 Arten der Tabellenpufferung .....	515
11.2.2 Pufferzugriffe .....	518
11.2.3 Pufferinvalidierung .....	520
11.2.4 Puffersynchronisation .....	521
11.2.5 Pufferung einschalten .....	524
11.2.6 Welche Tabellen sollen gepuffert werden? .....	528
<b>11.3 Tabellenpufferung auf dem ABAP-Server überwachen</b> .....	531
11.3.1 Tabellenzugriffsstatistik .....	532
11.3.2 Analyse der gepufferten Tabellen .....	535
11.3.3 Analyse der bisher nicht gepufferten Tabellen .....	540
11.3.4 Detaillierte Tabellenanalyse .....	542
11.3.5 Monitor zur Puffersynchronisation (DDLOG-Einträge) .....	545
11.3.6 Teure SQL-Anweisungen durch falsche Pufferung .....	546
<b>11.4 Objektorientierte Anwendungspuffer überwachen</b> .....	548
<b>11.5 Zusammenfassung</b> .....	553
<b>11.6 Übungsfragen</b> .....	555

## **12 Optimierung von Datenbankabfragen mit SAP HANA** 557

---

<b>12.1 Grundlagen der Hauptspeicherdatenbank SAP HANA</b> .....	559
12.1.1 Spaltenorientierte Datenspeicherung .....	561
12.1.2 Datenkomprimierung .....	562
12.1.3 Datenhaltung im Hauptspeicher .....	566
12.1.4 Indizierung .....	569
12.1.5 Textsuche und Kategorisierung .....	572
<b>12.2 Architektur von SAP HANA im Überblick</b> .....	573

<b>12.3</b>	<b>SAP-HANA-Administrationswerkzeuge zur Performanceoptimierung</b> .....	576
<b>12.4</b>	<b>Code Pushdown</b> .....	582
12.4.1	Tabellen- und Indexdesign .....	583
12.4.2	ABAP SQL .....	585
12.4.3	Core Data Services .....	587
12.4.4	Allgemeine Regeln für performante CDS-Views und SQL-Anweisungen .....	594
12.4.5	Konkrete Optimierungsbeispiele .....	599
12.4.6	Design von schreibenden Anwendungen .....	611
<b>12.5</b>	<b>Analyse teurer SQL-Anweisungen</b> .....	612
12.5.1	Ausführungsplan auswerten .....	614
12.5.2	Ausführungs-Trace erstellen und auswerten .....	619
<b>12.6</b>	<b>SQL Plan Cache und Result Cache</b> .....	625
12.6.1	SQL Plan Cache analysieren .....	625
12.6.2	Result Cache (Cached Views) .....	627
<b>12.7</b>	<b>Management von Tabellen auf SAP HANA</b> .....	639
12.7.1	Tabellengröße, Lade- und Entladevorgänge überwachen .....	640
12.7.2	Deltafragmente überwachen .....	646
12.7.3	Partitionierung und Skalierung .....	651
12.7.4	Datenreduktion durch Löschen oder Auslagern .....	657
12.7.5	Replikation .....	664
<b>12.8</b>	<b>Zusammenfassung</b> .....	666
<b>12.9</b>	<b>Übungsfragen</b> .....	670
<b>13</b>	<b>Optimierung analytischer Anfragen</b> .....	673
<hr/>		
<b>13.1</b>	<b>Grundlagenwissen zu analytischen Anfragen</b> .....	674
13.1.1	Übersicht über die wichtigsten Begriffe .....	675
13.1.2	Das Star-Schema .....	677
13.1.3	Analytische Anfragen .....	681
<b>13.2</b>	<b>Architektur von SAP S/4HANA Embedded Analytics und von SAP BW/4HANA</b> .....	683
13.2.1	SAP S/4HANA Embedded Analytics .....	683
13.2.2	SAP BW/HANA .....	687

13.2.3	Datenmodellierung aus Performancesicht .....	691
13.2.4	Datenladeprozesse .....	692
<b>13.3</b>	<b>Analyse teurer analytischer Anfragen</b> .....	695
13.3.1	Teure Anfragen identifizieren .....	695
13.3.2	BW-Statistiken im Workload-Monitor .....	697
13.3.3	Testumgebung für analytische Anfragen (Query-Monitor) .....	698
<b>13.4</b>	<b>Optimierung teurer analytischer Anfragen</b> .....	703
13.4.1	Einstellungen analytischer Anfragen optimieren .....	705
13.4.2	OLAP-Cache überwachen .....	709
13.4.3	Data Tiering .....	714
<b>13.5</b>	<b>Zusammenfassung</b> .....	716
<b>13.6</b>	<b>Übungsfragen</b> .....	717
 <b>Anhang</b> .....		719
<b>A</b>	<b>Wichtige Transaktionscodes und Anwendungen</b> .....	721
<b>B</b>	<b>Fragen und Antworten</b> .....	727
<b>C</b>	<b>Glossar</b> .....	733
<b>D</b>	<b>Informationsquellen</b> .....	743
 Der Autor .....		761
Index .....		763

# Das Update für Ihr Performancewissen

## Architektur und Analysewerkzeuge

Verstehen Sie, welche Elemente der Architektur die Performance beeinflussen. Nutzen Sie klassische und neue Tools wie das Technical Monitoring Cockpit, die ABAP Development Tools und den neuen SQL Analyzer für SAP HANA, um Flaschenhälse zu identifizieren.

## Datenmodellierung und Anfragen

Lernen Sie Strategien kennen, um Datenmodelle und SQL-Anfragen performant zu gestalten. Zusätzlich zeigt das Buch, wie Sie die Leistung analytischer Abfragen mit Embedded Analytics, CDS-Views und SAP BW/4HANA verbessern.

## Cloud- und On-Premise-Betrieb

Erfahren Sie, was der Betrieb in der Public Cloud für die Performanceoptimierung bedeutet. Daneben geht der Autor auch auf die Nutzung von SAP S/4HANA in der Private Cloud und on-premise ein.

## Auf einen Blick

- Workload-Analyse
- Hardware-Sizing
- Performanceoptimierung in den ABAP Development Tools
- Speicherkonfiguration
- Lastverteilung
- SAP Fiori und SAPUI5
- Sperren und Pufferung
- SQL-Tuning
- SAP-HANA-Datenmodellierung
- Embedded Analytics und CDS-Views
- SAP BW/4HANA
- Technical Monitoring Cockpit

»Praxisnah und verständlich erklärt!«

Leser-Feedback zu Voraufgabe



### Der Autor

Dr. Thomas Schneider war von 2015 bis 2024 Chief Development Architect für die ABAP-Plattform der SAP SE. Hier arbeitete er insbesondere an Projekten für SAP S/4HANA. Seit 2025 ist er freiberuflich tätig.

