



Workload-Manager Handbuch und Referenz



Workload-Manager Handbuch und Referenz

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen unter Anhang H, „Bemerkungen“, auf Seite 335 gelesen werden.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM DB2 Version 9.5 for Linux, UNIX, and Windows, Workload Manager Guide and Reference,
IBM Form SC23-5870-00,
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2007
© Copyright IBM Deutschland GmbH 2007

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
SW TSC Germany
Kst. 2877
Oktober 2007

Inhaltsverzeichnis

Zu diesem Handbuch vii

Teil 1. Einführung 1

Kapitel 1. Einführung in Workload-Management-Konzepte 3

Workload-Management-Phasen	3
Ermittlungsphase beim Workload-Management	5
Workloads	6
Arbeitsklassen	7
Verwaltungsphase beim Workload-Management	8
Überwachungsphase beim Workload-Management	14
Echtzeitüberwachung	15
Beispiel: Verwenden von Workload-Management-Tabellenfunktionen	17
Langzeitüberwachung	18
Aktivitäten	20
Workload-Management-Musteranwendung	20

Teil 2. Identifikation und Verwaltung 23

Kapitel 2. Serviceklassen 25

Standardservicesuperklassen und -unterklassen	27
Zuordnung von Aktivitäten zu Serviceklassen	29
CPU-Priorität und DB2-Serviceklassen	33
Vorablesezugriffspriorität bei Serviceklassen	33
Status von Verbindungen und Aktivitäten in einer Serviceklasse	34
Entitäten auf Systemebene, die nicht von Serviceklassen verfolgt werden	35
Arbeiten mit Serviceklassen	36
Erstellen einer Serviceklasse	36
Ändern einer Serviceklasse	38
Löschen einer Serviceklasse	41

Kapitel 3. Workloads 43

Workloadzuordnung	48
Standardworkloads	49
Zuordnen einer Verbindung zur Standardverwaltungsworkload	51
Arbeiten mit Workloads	52
Erstellen einer Workload	52
Ändern einer Workload	54
Erlauben des Datenbankzugriffs für Vorkommen einer Workload	55
Verhindern des Datenbankzugriffs für Vorkommen einer Workload	56
Aktivieren einer Workload	56
Inaktivieren einer Workload	57
Erteilen des Zugriffsrechts USAGE für eine Workload	57
Widerrufen des Zugriffsrechts USAGE für eine Workload	59

Löschen einer Workload 59

Kapitel 4. Schwellenwerte 61

Aktivitätsschwellenwerte und zusammengefasste Schwellenwerte	63
Schwellenwertzusammenfassung	63
Aktivitätsschwellenwerte	64
Schwellenwert CONNECTIONIDLETIME	64
Schwellenwert ESTIMATEDSQLCOST	64
Schwellenwert SQLTEMPSPACE	65
Schwellenwert SQLROWSRETURNED	66
Schwellenwert ACTIVITYTOTALTIME	67
Auflösung des Aktivitätsschwellenwertbereichs	67
Zusammengefasste Schwellenwerte	68
Schwellenwert TOTALDBPARTITIONCONNECTIONS	68
Schwellenwert TOTALSCPARTITIONCONNECTIONS	69
Schwellenwert CONCURRENTWORKLOADCURRENTENCES	70
Schwellenwert CONCURRENTWORKLOADACTIVITIES	71
Schwellenwert CONCURRENTDBCOORDACTIVITIES	73
Auswertungsreihenfolge für Schwellenwerte	74
Arbeiten mit Schwellenwerten	76
Erstellen eines Schwellenwerts	76
Ändern eines Schwellenwerts	82
Löschen eines Schwellenwerts	82

Kapitel 5. Arbeitsaktionssets, Arbeitsaktionen, Arbeitsklassensets und Arbeitsklassen 85

Arbeitsklassen und Arbeitsklassensets	85
Arbeitsaktionen und Arbeitsaktionssets	87
Beziehung zwischen Arbeitsklassen, Arbeitsklassensets, Arbeitsaktionen und Arbeitsaktionssets und deren Zuordnung zu anderen DB2-Objekten	89
Arbeitstypen und SQL-Anweisungen von Arbeitsklassen	91
Auswertungsreihenfolge von Arbeitsklassen in einem Arbeitsklassenset	93
Arbeitsaktionen und die Domäne des Arbeitsaktionssets	94
In Arbeitsaktionen verwendbare Schwellenwerte	97
Von Schwellenwerten unterstützte Arbeitsklassifikationen	97
Zuordnen von Aktivitäten zu Arbeitsklassen	98
Anwenden von Arbeitsaktionen auf Datenbankaktivitäten	99
Vergleich von Workload und Arbeitsaktionsset	101
Arbeiten mit Arbeitsaktionssets und Arbeitsaktionen	104
Erstellen eines Arbeitsaktionssets	104
Ändern eines Arbeitsaktionssets	105

Inaktivieren eines Arbeitsaktionssets	106
Löschen eines Arbeitsaktionssets	106
Erstellen einer Arbeitsaktion	107
Ändern einer Arbeitsaktion	110
Inaktivieren einer Arbeitsaktion	112
Löschen einer Arbeitsaktion	112
Arbeiten mit Arbeitsklassensets und Arbeitsklassen	113
Erstellen eines Arbeitsklassensets	113
Ändern eines Arbeitsklassensets	113
Löschen eines Arbeitsklassensets	114
Erstellen einer Arbeitsklasse	114
Ändern einer Arbeitsklasse	117
Löschen einer Arbeitsklasse	117

Teil 3. Überwachung und Steuerung 119

Kapitel 6. Überwachung und Steuerung 121

Überblick über Überwachungsdaten	121
Workload-Management-Funktionen für das Abrufen von Betriebsinformationen	124
Integration von Workload-Management-Tabellenfunktionen und des Überwachungsprogramms für Momentaufnahme	125
Gespeicherte Workload-Management-Prozeduren	126
Workload-Management-Ereignismonitore	127
Statistikmanagement	128
Statistikdaten für Workload-Management-Objekte	128
Histogramme beim Workload-Management	133
Erfassen von Workload-Management-Statistikdaten mit Hilfe eines Ereignismonitors für Statistikdaten	137
Workload-Management-Tabellenfunktionen für das Abrufen von Statistikdaten	140
Zurücksetzen von Statistikdaten für Workload-Management-Objekte	141
Überwachen von Schwellenwertverstößen	142
Erfassen von Daten für einzelne Aktivitäten	143
Importieren von Aktivitäten in den Designadvisor	145
Abbrechen von Aktivitäten	146
Richtlinien für das Erfassen von Informationen zu einer Rogue-Aktivität und Untersuchen dieser Aktivität	146
Modellierung der Workload-Management-Leistung	147
Arbeiten mit Histogrammen	147
Erstellen einer Histogrammschablone	147
Ändern einer Histogrammschablone	148
Löschen einer Histogrammschablone	149

Teil 4. Beispiele. 151

Kapitel 7. Workload-Management-Beispiele. 153

Beispiel: Verwenden von Serviceklassen	153
Beispiel: Workloadzuordnung	158
Beispiel: Workloadzuordnung, wenn Workloadattribute Einzelwerte haben	162

Beispiel: Workloadzuordnung für eine UOW, wenn mehrere Workloads vorhanden sind	164
Beispiel: Workloadzuordnung, wenn Workloadattribute mehrere Werte haben	167
Beispiel: Verwenden von Schwellenwerten.	169
Beispiel: Schwellenwerte CONCURRENTWORKLOADOCCURRENCES, TOTALDBPARTITIONCONNECTIONS und TOTALSCPARTITIONCONNECTIONS	170
Beispiel: Verwalten bestimmter Aktivitätstypen mit Hilfe von Arbeitsklassensets	171
Beispiel: Verwenden einer Arbeitsklasse mit dem Schlüsselwort ALL	172
Beispiel: Verwenden von Arbeitsaktionssets und Datenbankschwellenwerten.	174
Beispiel: Festlegen des Typs der auszuführenden Arbeit mit Hilfe von Arbeitsaktionssets.	176
Beispiel: Verwenden von Workload-Management-Tabellenfunktionen zum Überwachen des aktuellen Systemverhaltens auf verschiedenen Ebenen	177
Beispiel: Abrufen von Zeitpunktstatistikdaten aus Serviceklassen	180
Beispiel: Zusammenfassen von Daten mit Hilfe von Workload-Management-Tabellenfunktionen	181
Beispiel: Berechnen von Durchschnittswerten und einer Standardabweichung anhand von Histogrammen in einer Workload-Management-Konfiguration	182
Beispiel: Analysieren einer serviceklassenbezogenen Leistungsminderung des Systems.	184
Beispiel: Untersuchen einer workloadbezogenen Leistungsminderung des Systems.	186
Beispiel: Analysieren von Workloads nach Aktivitätstyp	187
Beispiel: Erkennen von blockierten Aktivitäten	188
Beispiel: Erfassen von Informationen zu einer Aktivität für die spätere Analyse	190
Beispiel: Untersuchen der Agentenauslastung nach Serviceklasse	192
Beispiel: Optimieren einer Workload-Management-Konfiguration, wenn Kapazitätsplanungsinformationen verfügbar sind	193
Beispiel: Optimieren einer Workload-Management-Konfiguration, wenn keine Kapazitätsplanungsinformationen verfügbar sind	195
Beispiel: Erkennen von Aktivitäten mit niedrigem geschätztem Aufwand und langer Laufzeit	200

Teil 5. Referenz 203

Kapitel 8. Prozeduren und Tabellenfunktionen. 205

WLM_CANCEL_ACTIVITY - Abbrechen einer Aktivität	205
WLM_CAPTURE_ACTIVITY_IN_PROGRESS - Erfassen von Aktivitätsinformationen für Aktivitätsereignismonitor	206
WLM_COLLECT_STATS - Erfassen und Zurücksetzen von Workload-Management-Statistiken	208

WLM_GET_ACTIVITY_DETAILS - Zurückgeben detaillierter Informationen zu einer bestimmten Aktivität	209
WLM_GET_QUEUE_STATS - Zurückgeben der Schwellenwertwarteschlangenstatistik	216
WLM_GET_SERVICE_CLASS_AGENTS - Auflisten von Agenten, die in einer Serviceklasse ausgeführt werden	221
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Liste der Workloadvorkommen	227
WLM_GET_SERVICE_SUBCLASS_STATS - Zurückgeben der Statistikdaten von Serviceunterklassen	231
WLM_GET_SERVICE_SUPERCLASS_STATS - Zurückgeben der Statistikdaten von Servicesuperklassen	236
WLM_GET_WORK_ACTION_SET_STATS - Zurückgeben von Statistikdaten zu einem Arbeitsaktionsset	238
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Zurückgeben einer Liste von Aktivitäten	240
WLM_GET_WORKLOAD_STATS - Zurückgeben von Workloadstatistikdaten.	244

Kapitel 9. Monitorelemente 249

Monitorelemente für Workload-Management	249
activate_timestamp - Zeitmarke für Aktivierung (Monitorelement)	249
activity_collected - Erfassen von Aktivitäten (Monitorelement)	249
activity_id - Aktivitäts-ID (Monitorelement)	250
activity_secondary_id - Sekundäre Aktivitäts-ID (Monitorelement)	250
activity_type - Aktivitätstyp (Monitorelement)	251
act_exec_time - Ausführungszeit für Aktivitäten (Monitorelement)	251
act_total - Gesamtzahl der Aktivitäten (Monitorelement)	252
arm_correlator - Korrelator zur Messung der Anwendungsantwortzeit (Monitorelement)	252
bin_id - ID des Histogramm-Bins (Monitorelement)	253
bottom - Untere Grenze von Histogramm-Bin (Monitorelement)	253
concurrent_act_top - Maximaler Wert für gleichzeitige Aktivitäten (Monitorelement).	254
concurrent_connection_top - Maximaler Wert für gleichzeitige Verbindungen (Monitorelement).	254
concurrent_wlo_act_top - Maximaler Wert für gleichzeitige Workloadaktivitäten (Monitorelement)	255
concurrent_wlo_top - Maximaler Wert für gleichzeitige Workloadvorkommen (Monitorelement)	255
coord_act_aborted_total - Gesamtzahl abgebrochener Koordinatoraktivitäten (Monitorelement).	255
coord_act_completed_total - Gesamtzahl abgeschlossener Koordinatoraktivitäten (Monitorelement)	256
coord_act_lifetime_top - Maximaler Wert für die Laufzeit von Koordinatoraktivitäten (Monitorelement)	256

coord_act_rejected_total - Gesamtzahl zurückgewiesener Koordinatoraktivitäten (Monitorelement)	257
coord_partition_num - Nummer der Koordinatorpartition (Monitorelement)	257
cost_estimate_top - Maximaler Wert für Kostenschätzung (Monitorelement)	258
coord_act_lifetime_avg - Durchschnittswert für die Laufzeit von Koordinatoraktivitäten (Monitorelement)	258
coord_act_queue_time_avg - Durchschnittliche Warteschlangenzeit für Koordinatoraktivitäten (Monitorelement)	259
coord_act_exec_time_avg - Durchschnittswert für die Ausführungszeit von Koordinatoraktivitäten (Monitorelement)	260
request_exec_time_avg - Durchschnittswert für die Ausführungszeit von Anforderungen (Monitorelement)	261
coord_act_est_cost_avg - Durchschnittswert für den geschätzten Aufwand für Koordinatoraktivitäten (Monitorelement)	261
coord_act_interarrival_time_avg - Durchschnittswert für die Ankunftszeit von Koordinatoraktivitäten (Monitorelement)	262
db_work_action_set_id - Set-ID für Datenbankarbeitsaktionen (Monitorelement).	263
db_work_class_id - Datenbankarbeitsklassen-ID (Monitorelement)	263
histogram_type - Histogrammtyp (Monitorelement)	264
last_wlm_reset - Zeitpunkt des letzten Zurücksetzens (Monitorelement)	265
num_threshold_violations - Anzahl der Schwellenwertverstöße (Monitorelement)	265
number_in_bin - Anzahl in Bin (Monitorelement)	266
parent_activity_id - ID der übergeordneten Aktivität (Monitorelement)	266
parent_uow_id - UOW-ID der übergeordneten Aktivität (Monitorelement)	266
prep_time - Vorbereitungszeit (Monitorelement)	267
queue_assignments_total - Gesamtzahl der Warteschlangenzuordnungen (Monitorelement)	267
queue_size_top - Maximaler Wert für Warteschlangengröße (Monitorelement)	268
queue_time_total - Gesamtwarteschlangenzeit (Monitorelement)	268
rows_fetched - Abgerufene Zeilen (Monitorelement)	268
rows_modified - Modifizierte Zeilen (Monitorelement)	269
rows_returned - Zurückgegebene Zeilen (Monitorelement)	269
rows_returned_top - Maximaler Wert für zurückgegebene Zeilen (Monitorelement)	270
sc_work_action_set_id - Set-ID für Serviceklassenarbeitsaktionen (Monitorelement)	270
sc_work_class_id - Arbeitsklassen-ID für Serviceklassen (Monitorelement)	271

section_env - Abschnittsumgebung (Monitorelement)	271
service_class_id - Serviceklassen-ID (Monitorelement)	272
service_subclass_name - Name der Serviceunterklasse (Monitorelement)	272
service_superclass_name - Name der Service-superklasse (Monitorelement)	272
statistics_timestamp - Zeitmarke für Statistikdatensatz (Monitorelement)	273
temp_tablespace_top - Maximaler Wert für Tabellenbereiche für temporäre Tabellen (Monitorelement)	273
threshold_action - Schwellenwertaktion (Monitorelement)	274
threshold_domain - Schwellenwertdomäne (Monitorelement)	274
threshold_maxvalue - Maximaler Wert für Schwellenwert (Monitorelement)	275
threshold_name - Name des Schwellenwerts (Monitorelement)	276
threshold_predicate - Vergleichselement für Schwellenwert (Monitorelement)	276
threshold_queue_size - Größe der Schwellenwertwarteschlange (Monitorelement)	276
thresholdid - Schwellenwert-ID (Monitorelement)	277
time_completed - Abschlusszeitpunkt (Monitorelement)	277
time_created - Erstellungszeitpunkt (Monitorelement)	278
time_of_violation - Zeitpunkt des Verstoßes (Monitorelement)	278
time_started - Startzeitpunkt (Monitorelement)	278
bottom - Obere Grenze von Histogramm-Bin (Monitorelement)	279
uow_id - UOW-ID (Monitorelement)	279
wlo_completed_total - Gesamtzahl abgeschlossener Workloadvorkommen (Monitorelement)	280
work_action_set_id - ID des Arbeitsaktionssets (Monitorelement)	280
work_action_set_name - Name des Arbeitsaktionssets (Monitorelement)	280
work_class_id - Arbeitsklassen-ID (Monitorelement)	281
work_class_name - Name der Arbeitsklasse (Monitorelement)	281
workload_id - Workload-ID (Monitorelement)	282
workload_name - Name der Workload (Monitorelement)	282
workload_occurrence_id - ID des Workloadvorkommens (Monitorelement)	283
Kapitel 10. Befehle	285
SET WORKLOAD (Befehl)	285
Kapitel 11. Konfigurationsparameter	287
wlm_collect_int - Workload-Management-Erfassungsintervall (Konfigurationsparameter)	287

Kapitel 12. Katalogsichten	289
SYSCAT.HISTOGRAMTEMPLATEBINS	289
SYSCAT.HISTOGRAMTEMPLATES	289
SYSCAT.HISTOGRAMTEMPLATEUSE	289
SYSCAT.SERVICECLASSES	290
SYSCAT.THRESHOLDS	292
SYSCAT.WORKACTIONS	293
SYSCAT.WORKACTIONSETS	296
SYSCAT.WORKCLASSES	297
SYSCAT.WORKCLASSESETS	298
SYSCAT.WORKLOADAUTH	298
SYSCAT.WORKLOADCONNATTR	299
SYSCAT.WORKLOADS	299

Teil 6. Anhänge und Schlussteil **301**

Anhang A. Aspekte von DDL-Anweisungen für Workload-Management **303**

Anhang B. Integration von DB2-Workload-Management und AIX Workload Manager **305**

Anhang C. Verarbeitung von gespeicherten Prozeduren in einer Workload-Management-Lösung **315**

Anhang D. Namenskonventionen **317**

Anhang E. Rollen **319**

Anhang F. Gesicherte Kontexte und Verbindungen **321**

Anhang G. Übersicht über die technischen Informationen zu DB2 **325**

Bibliothek mit technischen Informationen zu DB2 im Hardcopy- oder PDF-Format	326
Bestellen gedruckter DB2-Bücher	328
Aufrufen der Hilfe für den SQL-Status über den Befehlszeilenprozessor	329
Zugriff auf verschiedene Versionen der DB2-Informationenzentrale	329
Anzeigen von Themen in der gewünschten Sprache in der DB2-Informationenzentrale	329
Aktualisieren der auf Ihrem Computer oder Intranet-Server installierten DB2-Informationenzentrale	330
DB2-Lernprogramme	332
Informationen zur Fehlerbehebung in DB2	333
Bedingungen	333

Anhang H. Bemerkungen **335**

Index **339**

Zu diesem Handbuch

Dieses Handbuch enthält Informationen zu den DB2-Workload-Management-Features und -Funktionen, mit denen Sie eine stabile, Ausführungsumgebung mit vorhersagbaren Ergebnissen, die Ihre Geschäftsanforderungen erfüllt. Über das Workload-Management werden sowohl Anforderungen als auch Ressourcen verwaltet. In diesem Handbuch finden Sie außerdem Informationen zur Überwachung der Workload auf Ihrem Datenserver und zur Fehlerbehebung.

Teil 1. Einführung

Kapitel 1. Einführung in Workload-Management-Konzepte

Ein gutes Workload-Management-System trägt dazu bei, die gewünschten Ziele in der Umgebung, in der Arbeit anfällt, effizient zu erreichen. Sie können Beispiele für die Notwendigkeit eines guten Workload-Management-Systems überall in Ihrer Umgebung feststellen.

Sehen Sie sich z. B. ein Lebensmittelgeschäft an. Unterschiedliche Aktivitäten müssen berücksichtigt werden: Bedienung von Kunden, Auffüllen von Waren, Pflege des Lagerbestands usw. Außerdem müssen einige einfache Ziele festgelegt werden. Geschäftseigner wollen sowohl die Anzahl der Kunden im Laden als auch die Beträge für die Kundeneinkäufe maximieren und beide Ziele auf eine Weise erreichen, dass die Kunden zufrieden sind und gerne wiederkommen. Außerdem müssen Geschäftseigner sicherstellen, dass sie einen ausreichenden Warenbestand für ihre Kunden haben (wobei der Bestand aber nicht zu groß sein sollte, damit nicht zu viele Waren verderben). Geschäftseigner wollen auch verfolgen, was ihre Kunden kaufen, und diese Informationen zur Erstellung von Werbung nutzen, mit der die Kundschaft zum Wiederkommen bewegt werden soll. Überwachungsmechanismen verfolgen den Lagerbestand und senden Benachrichtigungen bei niedrigen Beständen. Zur Erkennung von Ladendiebstahl sind entsprechende Sicherungsvorrichtungen vorhanden. Spezielle Schnellkassen werden eingerichtet, so dass Käufer, die nur wenige Artikel kaufen wollen, dies tun können, ohne hinter Kunden mit vielen Waren warten zu müssen. Wenn alle diese Ziele erfüllt sind und alle Betriebsprozesse gut funktionieren, sind die Kunden zufrieden und kommen gern wieder, statt in ein anderes Geschäft zu gehen. Diese Ziele und Betriebsprozesse sind alles Aspekte von Workload-Management.

In einer Datenserverumgebung ist die Notwendigkeit für eine effektive Arbeitsverwaltung noch wichtiger, insbesondere jetzt, da Datenserver wie nie zuvor ausgelastet sind. Elektronische Kassensysteme generieren Tausende von Dateneinfügungen, es werden konstant Berichte generiert, um zu ermitteln, ob die Vertriebsziele erreicht werden, Stapelanwendungen werden ausgeführt, um erfasste Daten zu laden, und Verwaltungsaufgaben wie Sicherungen und Reorganisationen werden ausgeführt, um die Daten zu schützen und den Server optimal arbeiten zu lassen. Alle diese Operationen nutzen dasselbe Datenbanksystem und konkurrieren um dieselben Ressourcen.

Ein effizientes Workload-Management-System ist daher entscheidend, um die Ziele für den Betrieb eines Datenservers bestmöglich zu erreichen.

Workload-Management-Phasen

Das Workload-Management erfolgt in drei klar definierten Phasen: die Ermittlung der Arbeit vor dem Eintritt in den Datenserver, die Verwaltung der Arbeit während der Ausführung sowie die Überwachung, mit der sichergestellt wird, dass der Datenserver effizient genutzt wird.

Für ein erfolgreiches Workload-Management muss eine Reihe von Aspekten berücksichtigt werden, beginnend mit der Kenntnis Ihrer Ziele. Im Beispiel des in Kapitel 1, „Einführung in Workload-Management-Konzepte“ beschriebenen Lebensmittelgeschäfts wären mögliche Ziele, die Beträge für die Kundeneinkäufe zu maximieren, Ladendiebstahl zu minimieren und sicherzustellen, dass Kunden den Laden so zufrieden verlassen, dass sie wiederkommen.

Auch in einer Datenserverumgebung müssen Sie Ziele definieren. Manchmal sind die Ziele eindeutig, insbesondere wenn sie durch Service-Level-Agreements begründet sind. Beispielsweise dürfen Abfragen von einer bestimmten Anwendung höchstens 10% der gesamten CPU-Ressourcen verbrauchen. Ziele können auch an eine bestimmte Tageszeit gebunden sein. Beispielsweise muss ein über Nacht laufendes Stapeldienstprogramm bis 8 Uhr morgens das Laden von Daten abgeschlossen haben, damit die täglichen Verkaufsberichte rechtzeitig fertig sind. In anderen Situationen sind die Ziele möglicherweise nur schwer quantitativ zu bestimmen. Ein Ziel könnte sein, dass die Datenbankbenutzer zufrieden sind und dass keine anormale Datenbankaktivität sie in ihrer täglichen Arbeit behindert. Unabhängig davon, ob die Ziele quantitativ bestimmt werden können oder nicht, ist die Kenntnis der Ziele entscheidend, wenn die folgenden Phasen des Workload-Managements betrachtet werden:

- **Ermittlung.** Wenn Sie ein Ziel für eine bestimmte Art von Arbeit erreichen wollen, müssen Sie zuerst die Details zur Arbeit angeben können. Im Lebensmittelgeschäft können Sie Käuferinformationen über Kredit- und Kundenkarten oder nicht bezahlte Artikel über ein aktives Sicherheitsetikett am Artikel ermitteln. Beim Datenserver müssen Sie entscheiden, wie Sie die Arbeit angeben wollen, die im System eintrifft. Sie können den Namen der Anwendung, die die Arbeit übergibt, oder die Berechtigungs-ID, mit der die Arbeit übergeben wird, oder eine Kombination von Elementen verwenden, die eine Form von Identifikation bietet.
- **Verwaltung.** Die Verwaltungsphase umfasst Mechanismen, mit denen Sie Ihr Ziel immer besser erfüllen können, sowie Maßnahmen, die zu ergreifen sind, wenn das Ziel nicht erreicht wird. Ein Beispiel für einen Mechanismus ist die Handhabung von Preisprüfungen an Schnellkassen. Schnellkassen sollten zu einem höheren Durchsatz und zufriedeneren Kunden führen. Wenn jedoch eine Milchpackung falsch ausgezeichnet ist und eine Preisprüfung erforderlich wird, könnte das den Durchsatz an der Schnellkasse verringern. Eine Lösung für dieses Problem ist eine schnelle Preisprüfung verbunden mit dem Öffnen einer anderen Kasse (wenn möglich) und dem Versuch das Preisproblem so zu beheben, dass es nicht wieder auftritt. Beim Datenserver stellen Sie eventuell fest, dass die Gesamtleistung leidet, wenn einige schlecht geschriebene SQL-Anweisungen ausgeführt werden, dass in Stoßzeiten das Volumen sich stark vergrößert oder dass eine zu große Konkurrenz um dieselben Ressourcen durch unterschiedliche Anwendungen auftritt. Die Verwaltungsphase umfasst Mechanismen für die Zuordnung von Ressourcen, damit Sie Ihre Ziele erreichen können, sowie Maßnahmen für den Fall, in dem ein Ziel nicht erreicht wird.
- **Überwachung.** Die Überwachung ist aus den folgenden Gründen wichtig. Um zu ermitteln, ob Sie ein Ziel erreichen, müssen Sie einen Mechanismus haben, um den Fortschritt zu diesem Ziel hin zu verfolgen. Außerdem hilft die Überwachung Ihnen bei der Ermittlung der Probleme, die das Erreichen des Ziels verhindern können. In einem Laden kann der Filialleiter den Kundenfluss beobachten, sich automatisch Probleme wie Ladendiebstahl oder einen extrem niedrigen Bestand eines Artikels melden lassen oder eine Analyse vergangener Kaufmuster durchführen, um die optimale Produktplatzierung im Laden zu ermitteln. Bei einem Datenserver gibt es oft explizite Ziele für Antwortzeiten von Datenbankaktivitäten, und es ist wichtig, eine Möglichkeit zu haben, diese Messgrößen zu ermitteln und auf Trends hin zu untersuchen.

Die folgende Abbildung stellt die Phasen des Workload-Managements dar:

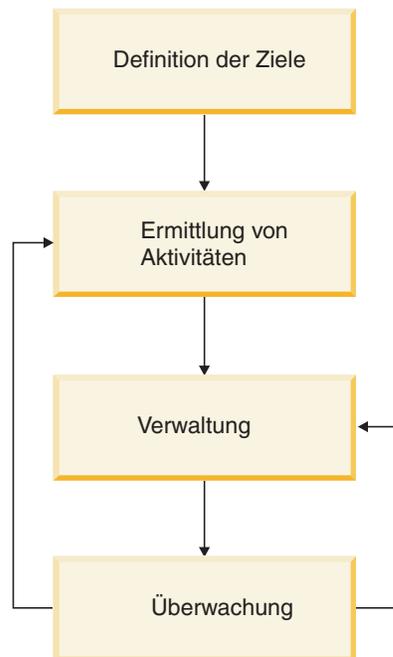


Abbildung 1. Phasen des Workload-Managements

Ermittlungsphase beim Workload-Management

Die erste Phase bei der Implementierung einer Workload-Management-Lösung ist die Ermittlung der Arbeit, die auf dem Datenserver ausgeführt wird.

Für die Ermittlung von Datenbankaktivitäten können unterschiedliche Methoden eingesetzt werden. Beispielsweise können Sie Aktivitäten nach ihrer Quelle angeben (d. h. danach, wer die Arbeit übergeben hat). Die folgende Abbildung zeigt eine Reihe unterschiedlicher Quellen von Arbeit: von unterschiedlichen Benutzern, Gruppen und Anwendungen. Aktivitäten können auch nach ihrem Typ angegeben werden.

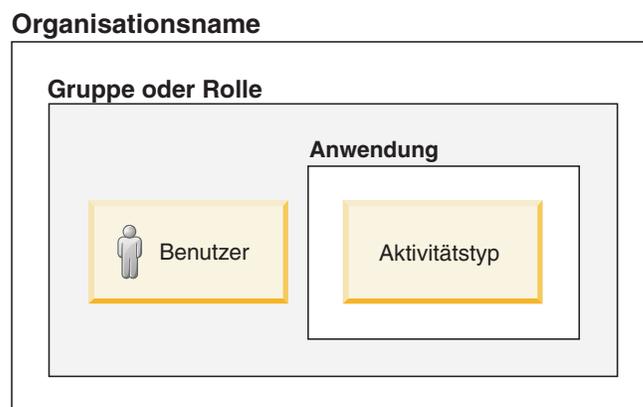


Abbildung 2. Verschiedene Quellen und Typen von Datenbankaktivitäten auf einem Datenserver

Zur Implementierung der Ermittlungsphase stehen zwei neue Datenbankobjekte zur Verfügung: Workloads und Arbeitsklassen.

Workloads

Eine *Workload* ist ein Objekt, das zur Angabe von eingehender Arbeit basierend auf ihrer Quelle verwendet wird, so dass diese später verwaltet werden kann. Die Quelle wird anhand der Attribute der Datenbankverbindung ermittelt, unter der die Arbeit übergeben wird.

Die Verbindungsattribute werden beim Herstellen der Verbindung ausgewertet, und die Verbindung wird einer Workloaddefinition zugeordnet. Zu diesem Zeitpunkt wird ein neues Vorkommen dieser Workload, ein so genanntes Workloadvorkommen, gestartet. Wenn sich eines der Verbindungsattribute während der Verbindung ändert, wird die Workloadzuweisung am Anfang der nächsten UOW (Unit of Work - Arbeitseinheit) nach der Änderung erneut ausgewertet. Wenn eine neue Workloaddefinition zugeordnet werden soll, wird das alte Workloadvorkommen für die zuvor zugeordnete Workload beendet, und ein neues Vorkommen für die neu zugeordnete Workloaddefinition wird gestartet. Während jede Verbindung jeweils nur einer einzigen Workload zugeordnet ist, können ein und derselben Workload mehrere Verbindungen gleichzeitig zugeordnet sein. Dies hat zur Folge, dass gleichzeitig mehrere Workloadvorkommen für diese Definition vorhanden sind.

Informationen zu den für eine Workload unterstützten Verbindungsattributen finden Sie in Kapitel 3, „Workloads“, auf Seite 43.

Die Workloadneuauswertung erfolgt am Anfang jeder UOW, falls der Wert eines Verbindungsattributs oder der Workloaddefinition selbst sich innerhalb der UOW ändert. Diese Neuauswertung kann dazu führen, dass die Verbindung einer neuen Workload zugeordnet wird und so ein anderes Workloadvorkommen erstellt wird.

Erstes Beispiel für das Erstellen einer Workload

```
CREATE WORKLOAD "REPORTING" APPLNAME('Accounts')  
SERVICE CLASS Marketing
```

Dieser Befehl erzeugt ein Workloadobjekt namens REPORTING. Alle Verbindungen mit einem Anwendungsnamensattribut von Accounts werden dieser Workload zugeordnet und in der Serviceklasse Marketing ausgeführt (siehe folgende Abbildung):



Abbildung 3. Workload REPORTING

Zweites Beispiel für das Erstellen einer Workload

```
CREATE WORKLOAD "SUMMARY" SESSION_USER_GROUP('Deptmgr') APPLNAME('Accounts')
SERVICE CLASS HumanResources
```

Dieser Befehl erzeugt ein Workloadobjekt namens SUMMARY. Alle Verbindungen mit einem Anwendungsnamen von Accounts, die zur Sitzungsbenutzergruppe Deptmgr gehören, deren Sitzungsbenutzerberechtigungs-ID zur Gruppe DeptMgr gehören, werden der Workload SUMMARY zugeordnet und in der Serviceklasse HumanResources ausgeführt (siehe folgende Abbildung):



Abbildung 4. Workload SUMMARY

Verbindungen, die bei der Workloadauswertung keiner benutzerdefinierten Workload zugeordnet werden, werden der Standardworkload zugeordnet. Mit der Standardworkload wird sichergestellt, dass alle Datenbankverbindungen mit einer Workload verknüpft sind.

Arbeitsklassen

Zusätzlich zur Verwendung von Verbindungsattributen, die sich auf die Quelle der Aktivitäten beziehen, können Sie Aktivitäten basierend auf dem Typ von Arbeit identifizieren, indem Sie eine optionale *Arbeitsklasse* erstellen.

Ein *Arbeitsklassenset* stellt eine allgemeine Definition von Arbeitstypen dar, die in unterschiedlichen Teilen des DB2-Datenservers in Verbindung mit einem Arbeitsaktionsset verwendet werden können.

Tabelle 1. Arbeitstypen

Arbeitstyp	Beschreibung
READ	Umfasst alle SELECT- und XQuery-Anweisungen, bei denen nur Daten abgerufen werden (d. h. Tabellen nicht aktualisiert werden).
WRITE	Umfasst alle Anweisungen, die Dateninhalte auf dem Datenserver ändern (d. h. INSERT, UPDATE, DELETE und MERGE, selbst wenn sie in einer SELECT-Anweisung eingebettet sind).
CALL	Umfasst alle Aufrufe von Prozeduren mit einer Anweisung CALL.
DML	Kombiniert Arbeit, die in den Arbeitstypen READ und WRITE gefunden wird.
DDL	Umfasst Anweisungen, die Datenbankobjekte erstellen oder ändern (d. h. CREATE, ALTER, DROP, COMMENT, DECLARE GLOBAL TEMPORARY TABLE, REFRESH TABLE, RENAME, GRANT, REVOKE, SET INTEGRITY).

Tabelle 1. Arbeitstypen (Forts.)

Arbeitstyp	Beschreibung
LOAD	Umfasst die gesamte Arbeit, die vom Dienstprogramm LOAD auf dem Datenserver eingeleitet wird.
ALL	Umfasst alle Typen von Arbeit.

Arbeitsklassen führen auch die Funktionalität für die Verwendung prädiktiver Elemente bei der Ermittlung für DML-Arbeit (oder READ- und WRITE-Anweisungen) ein. Prädiktive Elemente sind sehr nützlich, da sie Informationen zu Datenbankaktivitäten bieten, anhand derer Maßnahmen ergriffen werden können, bevor diese Aktivitäten beginnen, Ressourcen auf dem Datenserver zu verbrauchen. Die folgende Tabelle bietet Informationen zu prädiktiven Elementen:

Tabelle 2. Merkmale für prädiktive Identifikation

Prädiktives Element	Beschreibung
Geschätzter Aufwand	Verwendet den verfügbaren geschätzten Aufwand vom DB2-Compiler, um den gesamten DML-Code in einem gegebenen Timerbereich einzuschließen (z. B. Erstellen eines Arbeitsklassensets für alle großen Abfragen mit einem geschätzten Aufwand von über 1 000 000 Timerons)
Geschätzte Kardinalität	Verwendet die geschätzten zurückgegebenen Zeilen (Kardinalität) vom DB2-Compiler, um DML-Code in einem gegebenen Bereich zurückgegebener Zeilen einzuschließen (z. B. Erstellen einer Arbeitsklasse für große Abfragen, die geschätzt mehr als 500 000 Zeilen zurückgeben)

Sie können auch Aktivitäten angeben, indem Sie den Schemanamen der Prozedur verwenden, den eine Anweisung CALL aufruft. Basierend auf den Workloadattributen und Arbeitsklassentypen können Sie Arbeit identifizieren und für die nächste Phase, die Verwaltung der Arbeit, vorbereiten.

Mit Arbeitsaktionen können Sie auch teilweise definieren, wie die Aktivitäten in der Arbeitsklasse verwaltet werden sollen. Damit eine Arbeitsklasse aktiv ist, müssen Sie mindestens eine Arbeitsaktion für die Klasse definieren. Weitere Informationen finden Sie in der Beschreibung von Arbeitsaktionssets in „Verwaltungsphase beim Workload-Management“.

Verwaltungsphase beim Workload-Management

Nach der Ermittlung der Arbeit ist die nächste Phase die aktive Verwaltung der Arbeit, bei der Sie Ressourcen zuordnen und Steuerungen auf diese Arbeit anwenden.

Serviceklassen

Mit einer Serviceklasse wird eine Ausführungsumgebung definiert, in der die Arbeit ausgeführt werden kann. Diese Umgebung kann verfügbare Ressourcen und diverse Ausführungsschwellenwerte umfassen.

Beim Definieren einer Workload müssen Sie die Serviceklasse angeben, in der die mit dieser Workload verknüpften Arbeit ausgeführt wird. Darüber hinaus gibt es die Standardworkload, die sicherstellt, dass die gesamte Datenserverarbeit in einer Serviceklasse ausgeführt wird.

Priorisierung und Ressourcensteuerung

Beim Erstellen oder Ändern eines Serviceklassenobjekts können Sie eine Reihe von Ressourcensteuerungen definieren:

Tabelle 3. Ressourcensteuerung durch Serviceklassen

Steuerung	Beschreibung
Agentenpriorität	Diese Steuerung legt eine CPU-Prioritätsstufe für die Agententhreads in einer Serviceklasse fest. Diese Priorität wird als relative (Delta-)Priorität zu anderen Threads und Prozessen auf dem Datenserver an das Betriebssystem übergeben. Anmerkung: Nicht aktiv, wenn ein abgehender Korrelator verwendet wird.
Vorablesezugriffspriorität	Diese Steuerung ordnet Vorablesezugriffsanforderungen eine Priorität zu, die sich auf die Reihenfolge auswirkt, in der der Datenserver sich mit ihnen befasst.
Abgehender Korrelator	Diese Steuerung erlaubt es einer Workload, einige ihrer Ressourcen von einem externen Workload-Manager wie z. B. AIX Workload Manager (derzeit der einzige unterstützte externe Workload-Manager) steuern zu lassen. Die Kennung wird durch den Agenten an den externen Workload-Manager übergeben und nimmt eine Zuordnung zu einer mit dem Manager definierten Ressourcengruppe vor. Durch Kombination des DB2-Workload-Managements mit AIX Workload Manager kann das Leistungsspektrum erweitert werden. Mit Hilfe von AIX Workload Manager können Sie den CPU-Anteil steuern, der den einzelnen Serviceklassen zugeordnet wird. Es gibt u. a. die Möglichkeit, für jede Serviceklasse den minimalen, maximalen oder relativen CPU-Anteil festzulegen.

Serviceunterklassen

Die Servicesuperklasse ist zwar die höchste Arbeitsebene, Aktivitäten werden jedoch nur in Serviceunterklassen ausgeführt. Dies ist ein wichtiges Unterscheidungsmerkmal. Jede Servicesuperklasse enthält eine Standardserviceunterklasse, in der die Aktivitäten ausgeführt werden, die nicht explizit einer definierten Unterklasse zugeordnet sind. Die Standardunterklasse wird beim Erstellen der Servicesuperklasse erstellt. Sie können weitere Unterklassen in einer Serviceklasse erstellen, wenn Sie diese zur weiteren Eingrenzung von Arbeit oder von Ressourcen benötigen.

Sie können nur eine einzige Ebene von Unterklassen definieren (d. h. Sie können keine Unterklasse in einer anderen Unterklasse, sondern nur in einer Servicesuperklasse definieren).

Die folgende Abbildung ist ein Beispiel einer angepassten Workload-Management-Konfiguration mit Workloads und Serviceklassen:

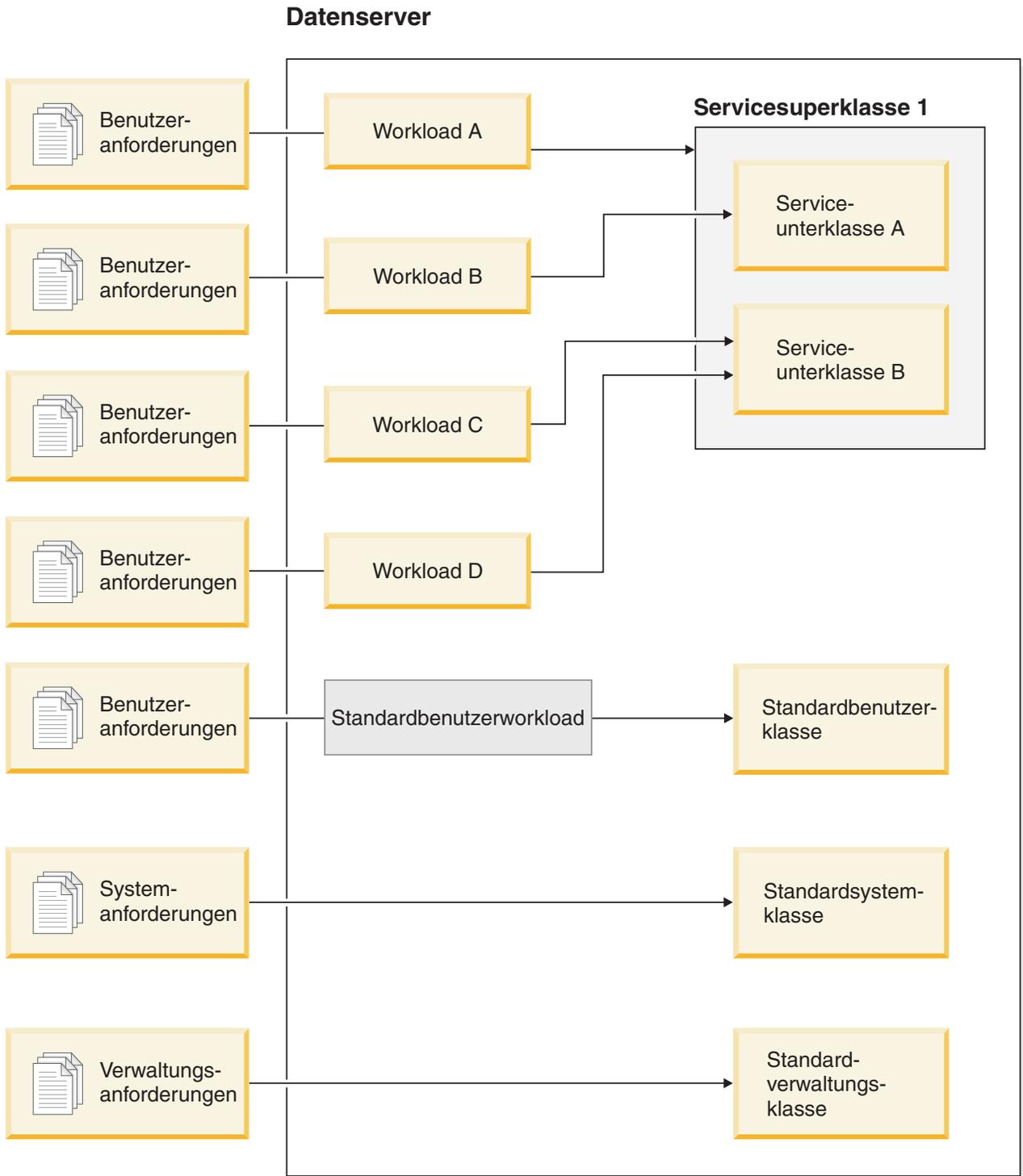


Abbildung 5. Angepasste Workload-Management-Konfiguration mit Workloads und Serviceklassen

Sobald Benutzeranforderungen in den Datenserver eingehen, werden sie als einer bestimmten Workload zugehörig identifiziert und einer Servicesuperklasse oder -unterklasse zugeordnet. Es gibt auch Systemanfragen (z. B. Vorabsezugriffe), die unter einer speziellen Standardsystems-serviceklasse ausgeführt werden, und von

DB2 gesteuerte Wartungsanforderungen (z. B. automatische RUNSTATS vom Diagnosemonitor), die unter einer Standardverwaltungsserviceklasse ausgeführt werden.

Schwellenwerte

Ressourcensteuerungen sind eine Möglichkeit, wie Sie versuchen können, einen konstanten ordnungsgemäßen Betrieb auf dem Server aufrechtzuerhalten. 'Ordnungsgemäßer Betrieb' bedeutet in diesem Zusammenhang, dass Ihre definierten Ziele erreicht werden. Manchmal trifft Arbeit ein, die Ihre normalen Erwartungen übersteigt (z. B. eine Abfrage, die Hunderttausende von Zeilen zurückgibt) und wertvolle Ressourcen zu Lasten der anderen auf dem System ausgeführten Arbeit verbraucht.

Sie können Schwellenwertobjekte erzeugen, um die Ordnung im System aufrechtzuerhalten und Arbeit mit einem anormalen Verhalten zu ermitteln. Die folgende Tabellen enthalten Informationen zu den unterschiedlichen Schwellenwerten, die Sie definieren können. Die erste Gruppe von Schwellenwerten befasst sich mit den Aktivitätsgrenzwerten, wobei die Steuerungen sich auf die Auswirkungen beziehen, die eine Aktivität auf die Ausführung des Datenservers haben kann. Zu viel Zeit, anormal hohe zurückgegebene Datenvolumina und anormal hoher Ressourcenverbrauch sind Warnzeichen dafür, dass potenziell problematische Aktivitäten möglicherweise mehr Ressourcen verbrauchen als ihnen zustehen.

Tabelle 4. Aktivitätsschwellenwerte

Schwellenwert	Beschreibung
ACTIVITYTOTALTIME	Steuert die Zeitspanne, die eine bestimmte Aktivität von der Übergabe bis zur Fertigstellung im DB2-Datenserver verbringen darf (Ausführungszeit plus Wartezeit). Mit diesem Schwellenwert können Sie Jobs erkennen, die anormal lange bis zu ihrer Fertigstellung benötigen.
CONNECTIONIDLETIME	Steuert die Zeitspanne, während der eine Verbindung im Leerlauf und nicht für Benutzeranforderungen aktiv ist. Mit diesem Schwellenwert können Sie eine ineffiziente Nutzung von Datenserverressourcen und Anwendungswartebedingungen erkennen.
ESTIMATEDSQLCOST	Steuert DML-Aktivitäten, bei denen das DB2-Optimierungsprogramm einen großen geschätzten Aufwand ermittelt. Mit diesem Schwellenwert können Sie SQL-Code mit potenziell hoher Ressourcennutzung vorhersagen, bevor dieser auf dem System ausgeführt wird, sowie schlecht geschriebenen SQL-Code identifizieren.
SQLROWSRETURNED	Steuert die Anzahl der Zeilen, die bei der SQL-Ausführung zurückgegeben werden. Mit diesem Schwellenwert können Sie ermitteln, wann das Datenvolumen eine angemessene Schwelle überschreitet.
SQLTEMPSPACE	Steuert die Größe des Tabellenbereichs für temporäre Tabellen, den eine bestimmte Aktivität in einer Partition verbrauchen darf. Mit diesem Schwellenwert können Sie Situationen verhindern, in denen einige fehlerhafte SQL-Anweisungen einen unverhältnismäßig großen Speicherbereich verbrauchen und so andere Arbeit behindern können.

Die nächste Gruppe von Schwellenwerten befasst sich mit der Steuerung des gemeinsamen Zugriffs und ist gedacht für Fälle, in denen Sie eventuell die Anzahl bestimmter gleichzeitig ausgeführter Aktivitäten einschränken wollen, um deren Auswirkung auf den Datenserver zu verringern.

Tabelle 5. Schwellenwerte für gemeinsamen Zugriff

Schwellenwert	Beschreibung
CONCURRENTWORKLOADOCCURRENCES	Steuert die Anzahl aktiver Vorkommen einer Workload, die auf einer Koordinatorpartition gleichzeitig ausgeführt werden können. Mit diesem Schwellenwert können Sie die Verteilung von Verbindungen von einer bestimmten Quelle steuern.
TOTALDBPARTITIONCONNECTIONS	Steuert die Anzahl der Datenbankverbindungen, die gleichzeitig zu einer bestimmten Partition hergestellt werden können. Mit diesem Schwellenwert können Sie verhindern, dass eine bestimmte Partition überlastet wird.
TOTALSCPARTITIONCONNECTIONS	Steuert die Anzahl der Datenbankverbindungen zu einer bestimmten Partition für Arbeit, die in einer bestimmten Serviceklasse gleichzeitig ausgeführt werden kann. Dieser Schwellenwert ist dem Schwellenwert TOTALDBPARTITIONCONNECTIONS ähnlich, aber feiner unterteilt, da die Verbindung mit einer Serviceklasse verknüpft ist.
CONCURRENTWORKLOADACTIVITIES	Steuert die Anzahl der einzelnen Aktivitäten, die in einem Workloadvorkommen ausgeführt werden können. Mit diesem Schwellenwert können Sie Arbeit in einem einzigen Workloadvorkommen beschränken.
CONCURRENTDBCOORDACTIVITIES	Steuert die Anzahl gleichzeitig ablaufender Aktivitäten in der Domäne, der der Schwellenwert zugeordnet ist (Datenbank, Arbeitsaktion, Servicesuperklasse oder Serviceunterklasse).

Welche Art von Aktionen bei einem Schwellenwertverstoß ausgeführt werden können, hängt vom Schwellenwert selbst ab.

Daten erfassen

Bei Schwellenwertverstößen werden in irgendeiner Form Daten erfasst. Standardmäßig wird ein Schwellenwertverstoß in einem aktivierten Schwellenwertverstoßereignismonitor aufgezeichnet. Eventuell wünschen Sie jedoch weitere Details. Sie können deshalb bei jeder Schwellenwertdefinition anfordern, dass mehr Daten in einem aktivierten Aktivitätsereignismonitor erfasst werden sollen, z. B. Informationen zu einzelnen Aktivitäten, der Anweisungstext, die Kompilierumgebung und sogar Eingabedatenwerte.

Ausführung stoppen

Eine übliche Aktion bei einem Schwellenwertverstoß ist das Stoppen der Aktivität. In diesem Fall wird ein Fehlercode an die übergebende Anwendung zurückgegeben, der angibt, dass gegen den Schwellenwert verstoßen wurde.

Ausführung fortsetzen

In einigen Situationen ist das Stoppen einer Aktivität eine zu harsche Maßnahme. Eine bevorzugte Maßnahme ist, die Ausführung der Aktivität weiterlaufen zu lassen und die relevanten Daten für einen Administrator zu erfassen, der dann später eine Analyse durchführen kann, um zu ermitteln, wie sich das erneute Auftreten dieses Zustands verhindern lässt. In dieser Situation wird kein Fehlercode an die übergebende Anwendung zurückgegeben. Wenn die Ausführung der Aktion fortgesetzt werden soll, erhält der Benutzer keinen Hinweis auf den Schwellenwertverstoß.

Arbeitsaktionsset

Wie in „Arbeitsklassen“ auf Seite 7 beschrieben, können Sie Arbeitsklassen definieren, die Aktivitäten eines bestimmten Typs darstellen (z. B. LOAD- oder READ-Aktivitäten). Eine Arbeitsaktion stellt eine Aktion zur Verfügung, die auf eine Arbeitsklasse angewendet werden kann. Ein Arbeitsaktionsset kann eine oder mehrere Arbeitsaktionen enthalten, die auf Aktivitäten in einer bestimmten Superklasse oder in der Datenbank als Ganzes angewendet werden können. Damit eine Arbeitsklasse aktiv ist und ihr Aktivitäten zugeordnet werden können, muss eine Arbeitsaktion für die Arbeitsklasse definiert sein.

Wenn Sie ein Arbeitsaktionsset auf eine Datenbank anwenden, gibt es mehrere Arten von Aktionen, die Sie auf Aktivitäten anwenden können, die unter eine Arbeitsklasse fallen (z. B. Schwellenwertdefinitionen, Stoppen der Ausführung, Erfassen von Aktivitätsdaten und Zählen der Aktivitäten). Das Definieren eines Schwellenwerts für eine Arbeitsaktion ist die leistungsfähigste Datenbankarbeitsaktion. Vielleicht wollen Sie verhindern, dass SQL-Code mehr als 100 000 Zeilen liest und zurückgibt. Sie können eine einzige Arbeitsklasse für ein Arbeitsaktionsset definieren, das SQL-READ-Anweisungen erkennt, und eine Arbeitsaktion mit einem Schwellenwert, der die Ausführung stoppen würde, wenn mehr als 100 000 Zeilen zurückgegeben würden. Weitere Informationen zu möglichen Aktionen finden Sie in „Arbeitsaktionen und die Domäne des Arbeitsaktionssets“ auf Seite 94.

Wenn Sie das Arbeitsaktionsset für eine Servicesuperklasse definieren, können Sie unterschiedliche Arten von Aktionen auf Aktivitäten anwenden, darunter die Zuordnung von Aktivitäten zu einer Serviceklasse, das Verhindern der Ausführung, das Erfassen von Aktivitäts- oder zusammengefassten Aktivitätsdaten und das Zählen der Aktivitäten. Typischerweise ordnet das Arbeitsaktionsset eine Aktivität einer Serviceunterklasse zu und hat für die Unterklasse definierte Schwellenwerte, um die Aktivität zu verwalten.

Benutzerdatenbank- anforderungen

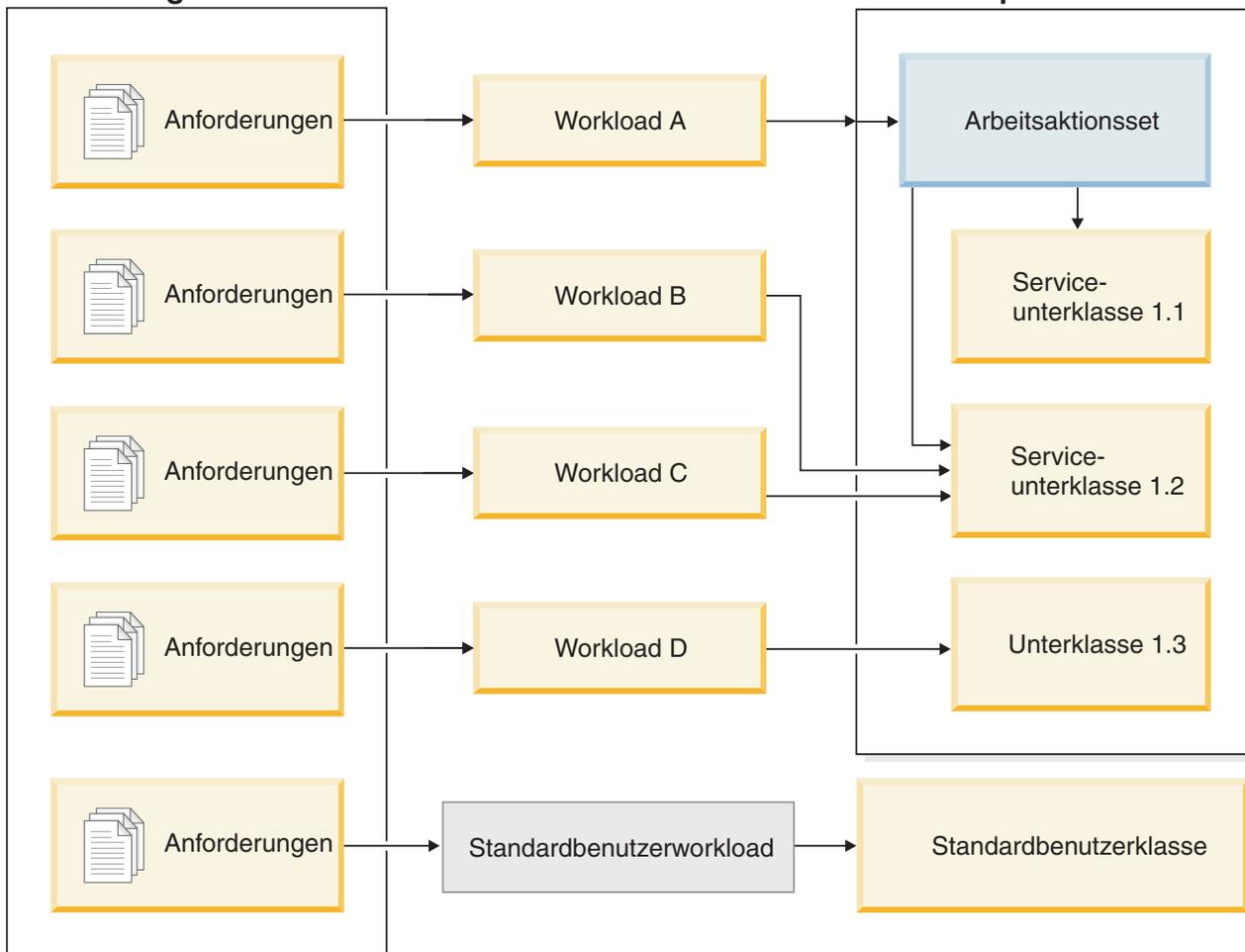


Abbildung 6. Zuordnung eines Arbeitsaktionssets für eine Servicesuperklasse

Überwachungsphase beim Workload-Management

Die dritte Phase beim Workload-Management ist die Überwachung. Diese wird zwar als letzte Workload-Management-Phase beschrieben, doch erfolgt die Überwachung ständig.

Der Hauptzweck der Überwachung ist die Überprüfung auf fehlerfreien Zustand und Effizienz Ihres Systems und der einzelnen Workloads, die darauf ausgeführt werden. Mit Tabellenfunktionen können Sie in Echtzeit auf Betriebsdaten (wie z. B. eine Liste aktiver Workloadvorkommen und der Aktivitäten in einer Serviceklasse oder der durchschnittlichen Antwortzeiten) zugreifen. Mit Ereignismonitoren können Sie detaillierte Aktivitätsinformationen und zusammengefasste Aktivitätsstatistikdaten für die Langzeitanalyse erfassen.

Beim Aufbau einer Überwachungsstrategie sollten Sie sich immer zuerst die zusammengefassten Informationen ansehen. Zusammenfassungen bieten ein gutes Bild der Gesamtaktivität des Datenservers und sind auch weniger aufwendig, da Sie nicht Informationen zu jeder einzelnen Aktivität erfassen müssen, an der Sie

eventuell interessiert sind. Sobald Sie den Umfang Ihrer Überwachungsbedürfnisse erkannt haben, können Sie genauere Informationen erfassen.

Echtzeitüberwachung

Mit Echtzeitüberwachung können Sie bei Leistungsproblemen oder Fehlerberichten ermitteln, was auf Ihrem System passiert.

Echtzeitüberwachungsdaten umfassen Statistikdaten, die die aktuelle Aktivität im System darstellen und bei der Ermittlung von Verwendungsmustern, der Ressourcenzuordnung und der Erkennung von Problembereichen helfen können.

Der Zugriff auf die Echtzeitüberwachungsdaten erfolgt über DB2-Tabellenfunktionen. Tabellenfunktionen bieten Zugriff auf eine Menge von Daten in einer DB2-Datenbank (z. B. die Workload-Management-Statistik) als virtuelle DB2-Tabelle, auf der Sie eine Anweisung `SELECT` ausführen können. Dies bietet Ihnen die Möglichkeit, Anwendungen für die Abfrage und Analyse von Daten zu schreiben, als befänden sich diese in einer physischen Tabelle auf dem Datenserver.

Einige Tabellenfunktionen geben Informationen zu der derzeit auf einem System ausgeführten Arbeit zurück. Diese Informationen sind auf verschiedenen Ebenen verfügbar:

Tabelle 6. Informationen von Tabellenfunktionen, die für die Ausführung von Arbeit verfügbar sind

Objekte, für die Informationen erfasst werden	Beschreibung
Workloadvorkommen	Die Tabellenfunktion <code>WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES</code> gibt datenbankpartitionsübergreifend eine Liste von Workloadvorkommen zurück, die einer Serviceklasse zugeordnet sind. Für jedes Vorkommen gibt es Informationen zum aktuellen Status und zu den Verbindungsattributen, die für die Zuordnung der Workload zur Serviceklasse verwendet wurden, sowie eine Aktivitätsstatistik, die den Umfang und die Erfolgsquote von Aktivitäten angibt.
Serviceklassenagenten	Die Tabellenfunktion <code>WLM_GET_SERVICE_CLASS_AGENTS</code> gibt eine Liste von Datenbankagenten zurück, die einer Serviceklasse oder einer Anwendungskennung zugeordnet sind. Die zurückgegebenen Informationen zeigen auch den aktuellen Status des Agenten, die vom Agenten durchgeführte Aktion und den Status dieser Aktion an.
Aktivitäten von Workloadvorkommen	Die Tabellenfunktion <code>WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES</code> gibt eine Liste aktueller Aktivitäten zurück, die einem Workloadvorkommen zugeordnet sind. Für jede Aktivität gibt es Informationen zum aktuellen Status der Aktivität (z. B. in Ausführung oder in der Warteschlange), zum Typ der Aktivität (z. B. <code>LOAD</code> , <code>READ</code> , <code>DDL</code>) und zum Zeitpunkt, zu dem die Aktivität gestartet wurde.
Aktivitätsdetails	Die Tabellenfunktion <code>WLM_GET_ACTIVITY_DETAILS</code> gibt Details zu einer Aktivität zurück. Eine zurückgegebene Information ist der Aktivitätstyp; je nach Typ wird ein Satz zusätzlicher Daten zurückgegeben. Für SQL-Aktivitäten z. B. sind Informationen zum Anweisungstext, den Paketdaten, den Aufwandsschätzungen und den zurückgegebenen oder geänderten Zeilen verfügbar. Details zur Isolationsstufe und zu den CPU-Zeiten sind ebenfalls verfügbar.

Allgemeine statistische Informationen sind auch auf mehreren Ebenen verfügbar, darunter die in der folgenden Tabelle beschriebenen Ebenen:

Tabelle 7. Statistische Informationen für eine Workload-Management-Lösung

Objekte, für die Statistikdaten zurückgegeben werden	Beschreibung der zurückgegebenen Statistikdaten
Servicesuperklassen	Die Tabellenfunktion WLM_GET_SERVICE_SUPERCLASS_STATS zeigt datenbankpartitionsübergreifend eine Übersichtsstatistik auf der Ebene der Servicesuperklasse: nämlich die oberen Grenzwerte für gleichzeitig bestehende Verbindungen (hilfreich bei der Ermittlung der höchsten Workloadaktivität).
Serviceunterklassen	Die Tabellenfunktion WLM_GET_SERVICE_SUBCLASS_STATS zeigt datenbankpartitionsübergreifend eine Übersichtsstatistik auf der Ebene der Serviceunterklasse (alle Aktivitäten, die in Serviceunterklassen ausgeführt werden). Die Statistikdaten umfassen die Anzahl abgeschlossener Aktivitäten und die durchschnittlichen Ausführungszeiten (hilfreich bei der Ermittlung des Gesamtsystemzustands und der Verteilung von Aktivitäten über Serviceklassen und Datenbankpartitionen hinweg).
Workloads	Die Tabellenfunktion WLM_GET_WORKLOAD_STATS zeigt datenbankpartitionsübergreifend eine Übersichtsstatistik auf der Workloadebene. Dies umfasst obere Grenzwerte für gleichzeitig ablaufende Workloadvorkommen und die Anzahl abgeschlossener Aktivitäten (hilfreich beim Überwachen des Gesamtsystemzustands oder beim Abrufen detaillierter Informationen zur Ermittlung von Problembereichen).
Arbeitsaktionssets	Die Tabellenfunktion WLM_GET_WORK_ACTION_SET_STATS zeigt datenbankpartitionsübergreifend eine Übersichtsstatistik auf der Ebene des Arbeitsaktionssets: nämlich die Anzahl der Aktivitäten jeder Arbeitsklasse, auf die die entsprechenden Arbeitsaktionen angewendet werden (hilfreich bei der Ermittlung der Effektivität eines Arbeitsaktionssets und beim Verständnis der auf dem System ausgeführten Arten von Aktivitäten).
Schwellenwertwarteschlangen	Die Tabellenfunktion WLM_GET_QUEUE_STATS zeigt datenbankpartitionsübergreifend eine Übersichtsstatistik für die Warteschlangen, die für ihre entsprechenden Schwellenwerte verwendet werden. Die Statistikdaten umfassen die Anzahl der Aktivitäten in der Warteschlange (aktuell und insgesamt) sowie die Gesamt-wartezeit in einer Warteschlange (hilfreich beim Abfragen der aktuellen Aktivität in der Warteschlange oder beim Überprüfen, ob ein Schwellenwert korrekt definiert ist: eine zu hohe Anzahl in der Warteschlange kann darauf hindeuten, dass ein Schwellenwert zu restriktiv ist, eine sehr kleine Anzahl, dass ein Schwellenwert nicht restriktiv genug ist oder nicht benötigt wird).

Statistikdaten sind nur hilfreich, wenn der Zeitraum, in dem sie erfasst werden, aussagekräftig ist. Das Erfassen von Statistikdaten über einen sehr langen Zeitraum sowie zu einem beliebigen Zeitpunkt mit der gespeicherten Prozedur WLM_COLLECT_STATS kann sich als wenig sinnvoll erweisen, wenn es schwierig wird, Änderungen an Trends oder Problembereiche zu erkennen, da zu viele alte Daten vorhanden sind. Deshalb gibt es die Möglichkeit, die Statistikdaten jederzeit zurückzusetzen.

Dank der Standardworkload und der Standardbenutzerserviceklassen sind die Überwachungsfunktionen sofort nach der Installation des DB2-Datenservers verfügbar. Diese können eine große Hilfe sein, um schnell mit der Ermittlungsphase des Workload-Management zu beginnen bzw. diese zu validieren, da sie helfen, Quellen von Aktivitäten einzugrenzen, die Sie zum Erstellen von Workloads und der Serviceklassen verwenden können, denen sie zugeordnet werden sollen.

Beispiel: Verwenden von Workload-Management-Tabellenfunktionen

Über die Echtzeitüberwachung von Workload-Management stehen große Datenmengen zur Verfügung. Das Beispiel in diesem Thema zeigt, wie Sie diese Informationen möglicherweise nutzen können.

In diesem Beispiel sind nur die Standardworkload und die Standardserviceklasse eingerichtet. Anhand dieses Beispiels können Sie verstehen, wie Sie die Tabellenfunktionen nutzen können, um zu erkennen, was auf dem Datenserver ausgeführt wird. Führen Sie die folgenden Schritte aus:

1. Verwenden Sie die Tabellenfunktion `WLM_GET_SERVICE_SUPERCLASS_STATS`, um alle Servicesuperklassen anzuzeigen. Nach der Installation oder Migration auf DB2 Version 9.5 sind drei Standardsuperklassen definiert: eine für Verwaltungsaktivitäten, eine für Systemaktivitäten und eine für Benutzeraktivitäten. `SYSDEFAULTUSERCLASS` ist die uns interessierende Serviceklasse.
2. Verwenden Sie die Tabellenfunktion `WLM_GET_SERVICE_SUBCLASS_STATS`, um statistische Daten für alle Serviceunterklassen der Superklasse `SYSDEFAULTUSERCLASS` anzuzeigen. Für jede Serviceunterklasse können Sie den aktuellen Umfang der verarbeiteten Anforderungen, die Anzahl der abgeschlossenen Aktivitäten und die Gesamtverteilung von Aktivitäten über Datenbankpartitionen anzeigen (wobei eine ungleichmäßige Verteilung möglicherweise auf ein Problem hindeutet). Sie können optional zusätzliche Statistikdaten erhalten, darunter die durchschnittliche Laufzeit für Aktivitäten, die durchschnittliche Wartezeit von Aktivitäten in der Warteschlange usw. Sie können optionale Statistikdaten für eine Serviceunterklasse ermitteln, indem Sie das Schlüsselwort `COLLECT AGGREGATE ACTIVITY DATA` in der Anweisung `ALTER SERVICE CLASS` angeben, um die Sammlung von zusammengefassten Aktivitätsstatistikdaten zu ermöglichen.
3. Verwenden Sie für eine bestimmte Serviceunterklasse die Tabellenfunktion `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES`, um die Vorkommen einer Workload aufzulisten, die der Serviceunterklasse zugeordnet sind. Die Tabellenfunktion zeigt alle Verbindungsattribute an, anhand derer Sie die Quelle der Aktivitäten ermitteln können. Diese Informationen können künftig sehr hilfreich bei der Ermittlung von angepassten Workloaddefinitionen sein. Beispielsweise hat ein bestimmtes hier aufgelistetes Workloadvorkommen ein großes Arbeitsvolumen von einer Anwendung, was durch den Zähler für abgeschlossene Aktivitäten angezeigt wird.
 - a. Verwenden Sie für diese Anwendung die Tabellenfunktion `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES`, um datenbankpartitionsübergreifend die aktuellen Aktivitäten anzuzeigen, die von der Verbindung dieser Anwendung erstellt wurden. Sie können diese Informationen für mehrere Zwecke verwenden, z. B. um zu ermitteln, welche Aktivitäten Probleme auf dem Datenserver verursachen könnten.
 - b. Rufen Sie für jede Aktivität genauere Informationen mit der Tabellenfunktion `WLM_GET_ACTIVITY_DETAILS` ab. Die Daten zeigen möglicherweise an, dass einige SQL-Anweisungen sehr viele Zeilen zurückgeben, dass einige Aktivitäten sehr lange im Leerlauf warten oder dass einige

Abfragen mit einem sehr hohen geschätzten Aufwand ausgeführt werden. In diesen und ähnlichen Situationen kann es sinnvoll sein, einige Schwellenwerte zu definieren, um möglicherweise schädigendes Verhalten zu ermitteln und in Zukunft zu verhindern.

Langzeitüberwachung

Zusätzlich zu den Tabellenfunktionen verwendet der DB2-Workload-Manager Ereignismonitore zur Erfassung von Informationen, die in Zukunft oder für die Langzeitanalyse nützlich sein könnten.

Es stehen drei Ereignismonitore zur Verfügung, die Sie in Ihrer Workload-Management-Konfiguration verwenden können. Jeder Ereignismonitor dient einem anderen Zweck:

Aktivitätseignismonitor

Dieser Monitor erfasst Informationen zu einzelnen Aktivitäten in einer Serviceklasse, Workload oder Arbeitsklasse bzw. Aktivitäten, die einen Schwellenwertverstoß verursacht haben. Wieviele Daten für jede Aktivität erfasst werden, kann konfiguriert werden. Dabei sollten Sie den Platten Speicherplatz und die Speicherdauer für die Monitoraten berücksichtigen. Eine übliche Verwendung von Aktivitätsdaten ist als Eingabe in Tools wie db2advis oder zur Verwendung von Zugriffsplänen (vom Dienstprogramm EXPLAIN), um die Tabellen-, Spalten- und Indexnutzung für eine Gruppe von Abfragen zu ermitteln.

Schwellenwertverstoß-Ereignismonitor

Dieser Monitor erfasst Informationen, wenn gegen einen Schwellenwert verstoßen wird. Er gibt an, gegen welchen Schwellenwert verstoßen wurde, welche Aktivität den Verstoß verursachte und welche Aktion in diesem Fall durchgeführt wurde.

Statistikereignismonitor

Dieser Monitor dient als wenig aufwendige Alternative zur Erfassung detaillierter Aktivitätsinformationen, indem Zusammenfassungsdaten erfasst werden (z. B. die Anzahl abgeschlossener Aktivitäten und die durchschnittliche Ausführungszeit). Zusammengefasste Daten umfassen Histogramme für eine Reihe von Aktivitätsmesswerten, einschließlich Laufzeit, Wartezeit, Ausführungszeit und geschätzter Aufwand. Mit Hilfe von Histogrammen können Sie die Verteilung von Werten verstehen, Ausreißer ermitteln und zusätzliche Statistikdaten wie Durchschnittswerte und Standardabweichungen berechnen. Mit Hilfe von Histogrammen können Sie beispielsweise nachvollziehen, welche Varianz in der Laufzeit die Benutzer erleben. Wenn eine sehr breite Streuung auftritt, spiegelt die durchschnittliche Laufzeit allein nicht die Erfahrungen der Benutzer wider.

Die folgende Abbildung zeigt die unterschiedlichen verfügbaren Überwachungsoptionen für den Zugriff auf Workloadinformationen: Tabellenfunktionen für den Zugriff auf Echtzeitstatistikdaten und Aktivitätsdetails sowie Langzeitinformationen, die als effiziente Zusammenfassungen erfasst werden, oder Details zu einzelnen Aktivitäten:

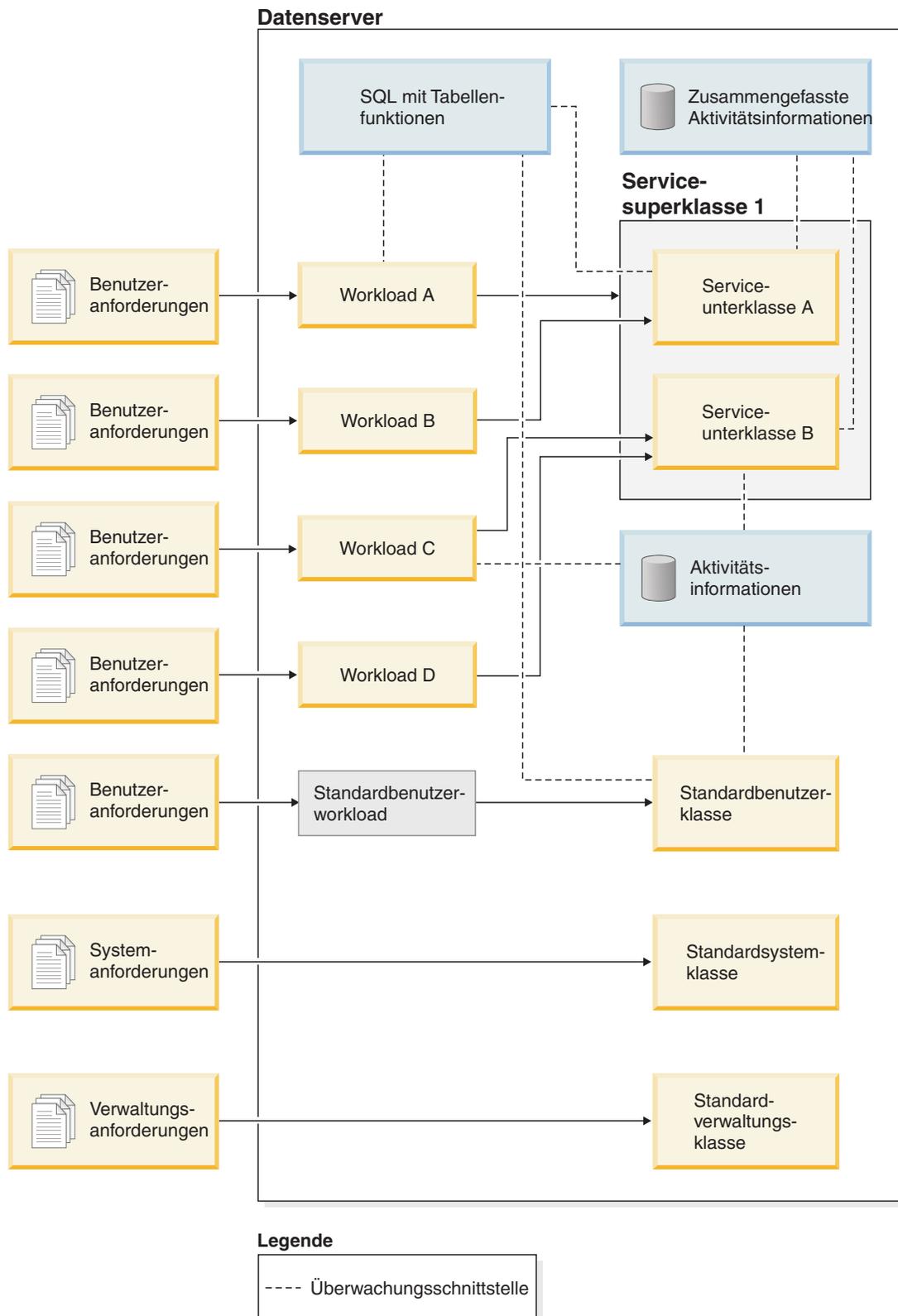


Abbildung 7. Workload-Management mit Überwachung

Aktivitäten

Eine Aktivität ist eine Arbeitseinheit, die während ihrer Laufzeit Datenbankressourcen verbraucht. Die Laufzeit kann eine oder mehrere Datenbankanforderungen umfassen. Eine Anweisung CALL oder ein Cursor sind Beispiele für Aktivitäten; eine Anweisung OPEN oder FETCH sind Beispiele für Datenbankanforderungen, die während der Laufzeit einer Cursoraktivität auftreten.

Der Datenserver erkennt die folgenden Aktivitäten für die Teilnahme in einer Workload-Management-Konfiguration:

- Alle DML-Anweisungen
- Alle DDL-Anweisungen
- Die Anweisung CALL
- Das Dienstprogramm LOAD

Arbeit, die nicht als eine der oben aufgeführten Aktivitäten klassifiziert ist, kann dennoch einer Workload (und der entsprechenden Serviceklasse, die in der Workloaddefinition angegeben ist) zugeordnet werden, basierend auf den Attributen der Verbindung, unter der die Arbeit übergeben wird. Auf diese Arbeit werden jedoch keine Schwellenwerte oder Arbeitsaktionen angewendet.

Im Kontext von Aktivitäten bezeichnet "Verschachteln" die Situation, in der eine Aktivität eine andere aufruft. Eine gespeicherte Prozedur z. B. ist eine Aktivität. Wenn die gespeicherte Prozedur eine DML-Aktivität enthält, handelt es sich hierbei um eine verschachtelte Aktivität. Wenn die gespeicherte Prozedur eine andere gespeicherte Prozedur aufruft, handelt es sich bei der zweiten gespeicherten Prozedur um eine verschachtelte Aktivität.

Workload-Management-Musteranwendung

In Version 9.5 wurde ein umfassendes Workload-Management integriert, was Ihnen eine genauere Kontrolle über Aktivitäten, Ressourcen und die Leistung sowie ein genaueres Verständnis darüber gibt, wie Ihr System arbeitet. Eine Workload-Management-Musteranwendung ist jetzt unter developerWorks verfügbar.

Die Workload-Management-Musteranwendung veranschaulicht, wie Sie Workload-Management-Features für die folgenden Ziele einsetzen können:

Schutz des Systems vor nicht mehr steuerbaren Abfragen

Nicht mehr steuerbare Abfragen sind aufwendig und verschlechtern die Leistung. Die Workload-Management-Musteranwendung ermittelt Abfragen, die potenziell nicht mehr steuerbar werden, und stoppt die Ausführung dieser Abfragen, nachdem sie gegen einen bestimmten Schwellenwert verstoßen haben.

Beschränken der gleichzeitigen Ressourcennutzung durch einzelne Anwendungen

Die Musteranwendung zeigt, wie Sie mit Workload-Management-Features verhindern können, dass Anwendungen, die große Mengen gleichzeitig ablaufender Arbeit übergeben, sich negativ auf die Leistung anderer Anwendungen auswirken.

Erreichen einer bestimmten Antwortzeit

Mit Workload-Management-Features können Sie eine bestimmte angestrebte Antwortzeit der Form "Transaktion X von Anwendung Y soll in 90% aller Fälle innerhalb 1 Sekunde abgeschlossen sein" erreichen, unabhängig davon, welche andere Aktivität gleichzeitig auf dem System ausgeführt wird. Die Musteranwendung zeigt, wie Sie eine angestrebte Antwortzeit erreichen können.

Konsistente Antwortzeit für kurze Abfragen

Abfragen, die typischerweise eine Antwortzeit von weniger als 1 Sekunde haben, sollten eine relativ konsistente Antwortzeit haben, unabhängig davon, welche anderen Workloads gerade auf dem System ausgeführt werden. Die Musteranwendung verwendet zur Überwachung der Konsistenz das Histogramm der Abfrageausführungszeit.

Schutz des Systems in Zeiträumen mit Lastspitzen

Features für Workload-Management-Richtlinien schützen das System vor Überlastung bei Lastspitzen, indem Arbeit in die Warteschlange gestellt wird, sobald das System ausreichend ausgelastet ist.

Aktivieren von gleichzeitig ablaufender Extraktions-, Transformations- und Ladeverarbeitung und von Benutzerabfragen

Mit Workload-Management-Features können Sie Extraktions-, Transformations- und Ladejobs (bei denen Daten in Tabellen geladen werden) ausführen und dabei den Leistungseinfluss für Benutzer steuern, die gleichzeitig Abfragen ausführen.

Teil 2. Identifikation und Verwaltung

Kapitel 2. Serviceklassen

Alle Datenbankankorderungen werden in einer Serviceklasse ausgeführt. Die Serviceklasse ist der wichtigste Punkt für die Ressourcenzuordnung für alle Datenbankankorderungen. Serviceklassen werden auch zur Überwachung und Steuerung von Gruppen von Datenbankaktivitäten in einer Datenbank verwendet.

Jede DB2-Serviceklasse ist entweder eine Servicesuperklasse oder eine Serviceunterklasse in einer Servicesuperklasse. Jede Datenbank kann mehrere Servicesuperklassen, und jede Servicesuperklasse kann mehrere Serviceunterklassen haben. Serviceunterklassen können nur zu einer Servicesuperklasse gehören. Die Ressourcen in der Superklasse werden von allen darin enthaltenen Unterklassen gemeinsam genutzt.

Durch die Einrichtung von Serviceklassen können Sie Aktivitäten in der Datenbank organisieren, so dass Sie die Leistungsziele Ihrer Organisation erreichen können. Ohne Serviceklassen können Anforderungen nicht in erkennbaren logischen Gruppierungen organisiert werden (siehe hierzu folgende Abbildung).

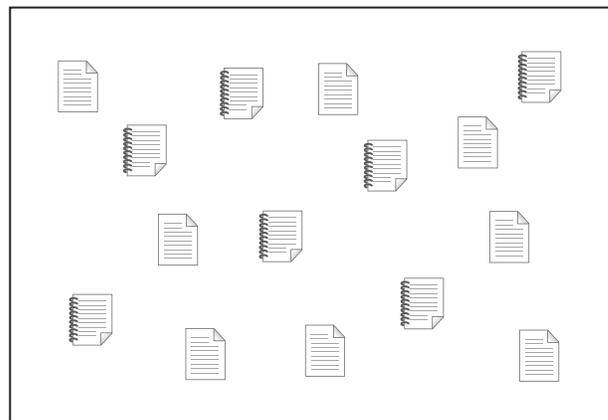


Abbildung 8. Nicht organisierte Arbeit

Sie können unterschiedliche Servicesuperklassen erstellen, um die Ausführungsumgebung für unterschiedliche Arten von Arbeit bereitzustellen, und dann den Servicesuperklassen die zutreffenden Anforderungen zuzuordnen. Nehmen wir einmal an, Sie haben Anwendungen aus zwei unterschiedlichen Geschäftsbereichen: Finance und Inventory. Dabei hat jeder Geschäftsbereich eigene Anwendungen, um die jeweiligen Aufgaben erfüllen zu können, für die der Geschäftsbereich in der Organisation zuständig ist. Sie können die Anforderungen in Kategorien aufteilen, die für Ihre Workload-Management-Ziele sinnvoll sind. In der folgenden Abbildung werden unterschiedliche Servicesuperklassen den unterschiedlichen Geschäftsbereichen zugeordnet.

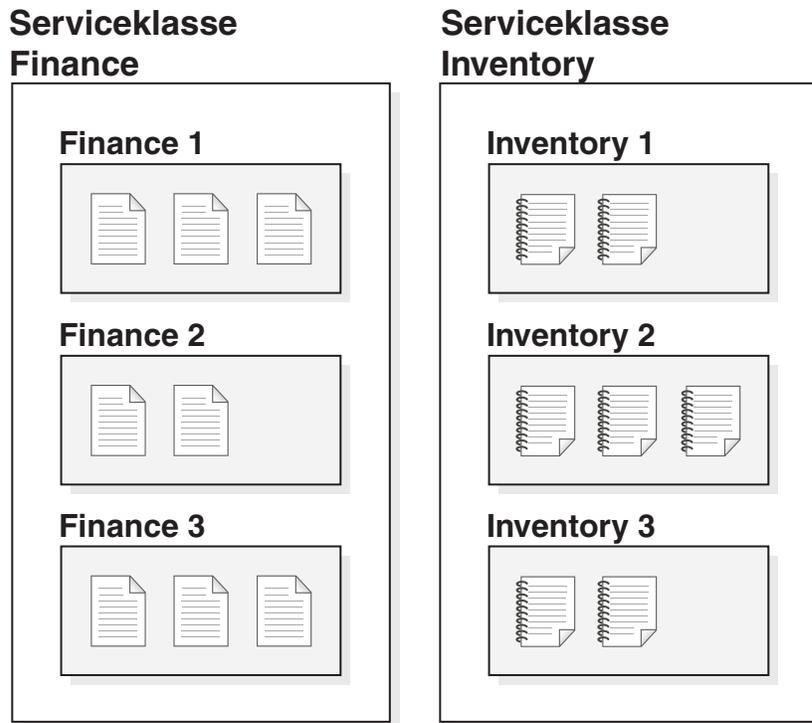


Abbildung 9. Nach Serviceklassen organisierte Arbeit

In der Abbildung oben sind die Aktivitäten in beiden Servicesuperklassen noch weiter unterteilt. Die Serviceklasse bietet eine zweistufige Hierarchie: eine Servicesuperklasse und darunter Serviceunterklassen. Diese Hierarchie erlaubt eine komplexere Unterteilung der Ausführungsumgebung und eine bessere Emulation eines realistischen Modells. Sofern nichts anderes angegeben ist, übernehmen Serviceunterklassen ihre Merkmale von der Servicesuperklasse. Mit Serviceunterklassen können Sie die Arbeit in der Servicesuperklasse weiter unterteilen.

Alle Arbeit wird in den Serviceunterklassen ausgeführt. Beim Erstellen einer Servicesuperklasse wird automatisch eine Standardserviceunterklasse erstellt. Arbeit, die der Servicesuperklasse zugeordnet wird (und darin verbleibt, d. h. keiner anderen Unterklasse zugeordnet wird), wird automatisch dieser Standardunterklasse zugeordnet. Die Servicesuperklasse fungiert als einheitlicher Hintergrund für alle zugehörigen Serviceunterklassen. Mit Ausnahme von Histogrammen und den Optionen für die Sammlung von Aktivitätsdaten übernimmt eine Serviceunterklasse die Attribute ihrer Servicesuperklasse, sofern nichts anderes angegeben ist.

Da unterschiedliche Arbeit unterschiedliche Priorität hat, können Sie bei Verwendung von Serviceklassen eine Reihe von Merkmalen dieser Serviceklasse steuern. Zum Beispiel:

- Sie können die Priorität der E/A-Seiten-Vorablesefunktion für Arbeit in der Serviceklasse festlegen.
- Sie können die Agenten-CPU-Priorität für alle Agenten in der Serviceklasse festlegen.
- Sie können die Ressourcennutzung in der Serviceklasse durch Anwendung von Integritätsbedingungen auf Arbeit in einer Serviceklasse steuern, indem Sie unterschiedliche Arten von Schwellenwerten verwenden.

Sie können Workloads verwenden, um Arbeit Servicesuperklassen zuzuordnen. Sie können Arbeit auch Serviceunterklassen in einer Servicesuperklasse zuordnen, indem Sie Workloaddefinitionen oder Arbeitsaktionen verwenden.

Mit der Anweisung `CREATE SERVICE CLASS` können Sie Serviceklassen erstellen. Mit der Anweisung `ALTER SERVICE CLASS` können Sie Serviceklassen ändern. Mit der Anweisung `DROP SERVICE CLASS` können Sie Serviceklassen löschen.

Sie können Ihre Serviceklassen durch Abfrage der Katalogsicht `SYSCAT.SERVICE-CLASSES` anzeigen.

Standardservicesuperklassen und -unterklassen

Wenn Sie DB2 Version 9.5 oder neuer installieren oder darauf migrieren, hat jede neue oder migrierte Datenbank drei vordefinierte Standardservicesuperklassen: die Standardbenutzerklasse, die Standardverwaltungs-klasse und die Standard-system-klasse.

Die Standardservicesuperklassen können nicht inaktiviert oder gelöscht werden.

Alle Standardservicesuperklassen werden mit einer Standard-serviceunterklasse erstellt. Für die Standardservicesuperklassen können keine weiteren Serviceunterklassen erstellt werden. Die Standard-serviceunterklasse hat immer den Namen `SYSDEFAULTSUBCLASS` und wird wie folgt erstellt:

SYSDEFAULTUSERCLASS

SYSDEFAULTSUBCLASS

SYSDEFAULTSYSTEMCLASS

SYSDEFAULTSUBCLASS

SYSDEFAULTMAINTENANCECLASS

SYSDEFAULTSUBCLASS

Abbildung 10. Zweistufige Serviceklassenhierarchie

Die gesamte Arbeit, die von Verbindungen an eine Standardservicesuperklasse abgesetzt wird, wird in der Standard-serviceunterklasse dieser Servicesuperklasse ausgeführt.

Standardservicesuperklassen und ihre Standard-serviceunterklassen werden nur beim Löschen der Datenbank gelöscht. Sie können nicht mit der Anweisung `DROP SERVICE CLASS` gelöscht werden.

Standardbenutzerservicesuperklasse (SYSDEFAULTUSERCLASS)

Nach der Installation oder Migration auf DB2 Version 9.5 werden standardmäßig alle Aktivitäten in `SYSDEFAULTUSERCLASS` ausgeführt.

Standardverwaltungsservicesuperklasse (SYSDEFAULTMAINTENANCECLASS)

Die Standardverwaltungsservicesuperklasse verfolgt die internen DB2-Verbindungen, die Tasks zur Datenbankpflege und -verwaltung ausführen. Verbindungen von den DB2-ABP-Agenten (ABP - Asynchronous Background Processing - asynchrone Hintergrundverarbeitung) werden dieser Servicesuperklasse zugeordnet. ABP-Agenten sind interne Agenten, die Tasks zur Datenbankpflege ausführen. Die asynchrone Indexbereinigung (AIC - Asynchronous Index Cleanup) ist ein Beispiel für eine ABP-Task. ABP-Agenten verringern automatisch ihre Ressourcennutzung und die Anzahl der Subagenten, wenn sich die Anzahl der Benutzerverbindungen auf dem Datenserver erhöht. Dienstprogramme, die von Benutzerverbindungen abgesetzt werden, werden mit Hilfe regulärer Serviceklassen zugeordnet. Auf SYSDEFAULTMAINTENANCECLASS können Sie keine Serviceklassenschwellenwerte implementieren.

Folgende interne Verbindungen werden von der Standardverwaltungsservicesuperklasse verfolgt:

- ABP-Verbindungen (einschließlich AIC)
- Vom Diagnosemonitor eingeleitetes BACKUP
- Vom Diagnosemonitor eingeleitete RUNSTATS
- Vom Diagnosemonitor eingeleitetes REORG

Standardsystemservicesuperklasse (SYSDEFAULTSYSTEMCLASS)

Die Standardsystemservicesuperklasse verfolgt interne DB2-Verbindungen und -Threads, die Tasks auf Systemebene ausführen. Sie können für diese Servicesuperklasse weder Serviceunterklassen definieren noch Workloads oder Arbeitsaktionen mit ihr verknüpfen. Außerdem können Sie bei SYSDEFAULTSYSTEMCLASS keine Serviceklassenschwellenwerte implementieren. Folgende DB2-Threads und -Verbindungen werden von der Standardsystemservicesuperklasse verfolgt:

- ABP-Dämon
- Verbindungen, die von der Query Controller-Komponente von Query Patroller abgesetzt werden
- Manager für Speicher mit automatischer Leistungsoptimierung
- EDUs (Engine Dispatchable Units - von der Steuerkomponente zuteilbare Einheiten) (db2pfchr)
- Seitenlöscher-EDUs (db2pclnr)
- Protokoll-Lese-EDUs (db2loggr)
- Protokollausgabe-EDUs (db2loggw)
- Protokolldateilese-EDUs (db2lfr)
- Deadlock-Detektor-EDUs (db2dlock)
- Ereignismonitore (db2evm)
- Verbindungen, die Tasks auf Systemebene ausführen

Zuordnung von Aktivitäten zu Serviceklassen

Alle Datenbankverbindungen werden am Anfang der ersten UOW (Unit of Work - Arbeitseinheit) einer Workload zugeordnet. Beim Start eines Workloadvorkommens werden alle Aktivitäten, die unter diesem Workloadvorkommen ausgeführt werden, basierend auf dem in der Workloaddefinition festgelegten Serviceklassennamen den Serviceklassen zugeordnet. Wenn das Workloadvorkommen einer Servicesuperklasse zugeordnet wird, kann die gesamte Arbeit, die für dieses Workloadvorkommen übergeben wird, von einer Arbeitsaktion (die auf eine Arbeitsklasse angewendet wird) einer benutzerdefinierten Serviceunterklasse in dieser Servicesuperklasse (d. h. nicht der Standardserviceunterklasse) neu zugeordnet werden, wenn ein Arbeitsaktionsset für die Servicesuperklasse definiert ist.

Der Datenserver ordnet eine Verbindung einer Workloaddefinition zu, wenn die Verbindung die für diese Workloaddefinition definierten Bedingungen erfüllt. Sie können beispielsweise eine Workload-Management-Implementierung einrichten, so dass alle Verbindungen von der Anwendung A zur Workloaddefinition Alpha gehören, während alle Verbindungen von der Anwendung B zur Workloaddefinition Beta gehören.

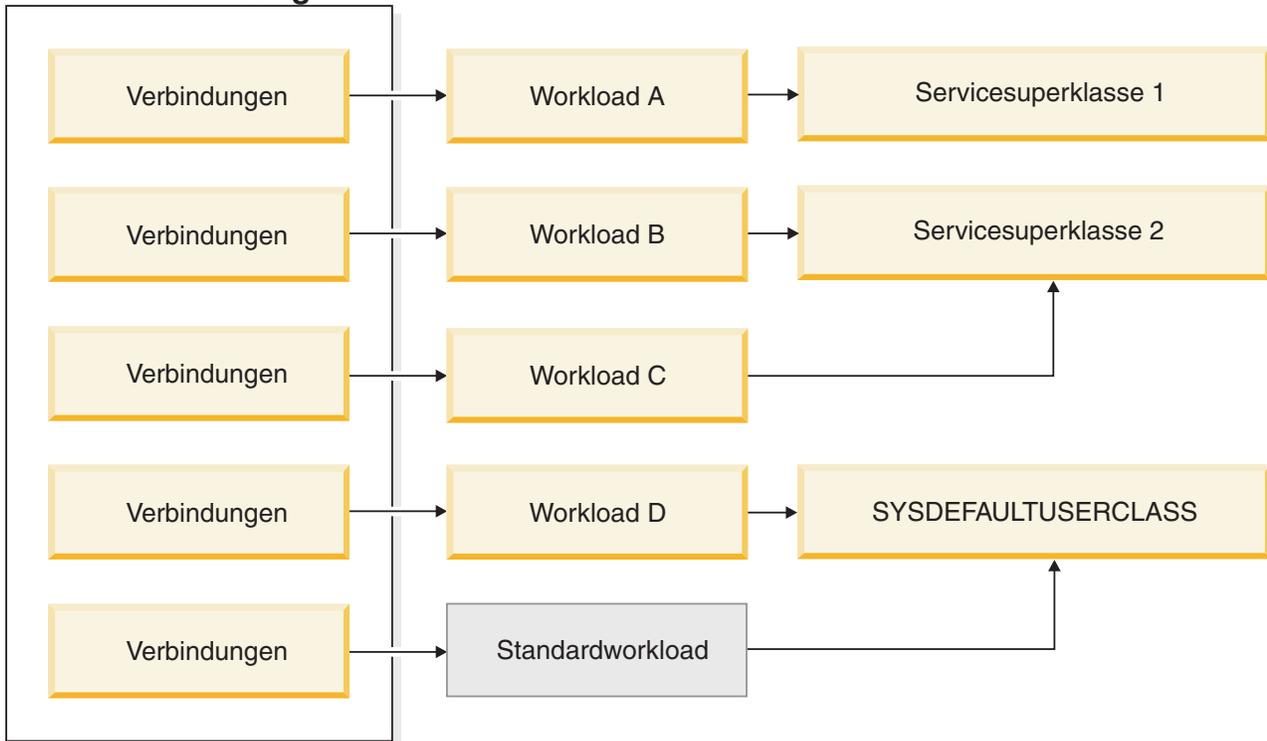
Sie können mit Hilfe der Workload Aktivitäten von einer Verbindung einer Servicesuperklasse zuordnen, indem Sie das Schlüsselwort `SERVICE CLASS` der Anweisung `CREATE WORKLOAD` angeben. Wenn keine Arbeitsklasse oder Arbeitsaktion auf die Aktivität angewendet wird, wird die Aktivität in der Standardserviceunterklasse der Servicesuperklasse ausgeführt.

Sie können mit Hilfe einer Workload auch Aktivitäten von einer Verbindung einer Serviceunterklasse in der Servicesuperklasse zuordnen, indem Sie das Schlüsselwort `UNDER` für das Schlüsselwort `SERVICE CLASS` der Anweisung `CREATE WORKLOAD` angeben. In diesem Fall gehört die Verbindung noch zur Servicesuperklasse, aber alle Aktivitäten, die von dieser Verbindung abgesetzt werden, werden automatisch der Serviceunterklasse zugeordnet, die in der Workloaddefinition angegeben ist.

Anmerkung: Nur der Koordinatoragent nimmt die Servicesuperklassenzuordnung für die Verbindung vor. Wenn der Koordinatoragent Subagenten startet, übernehmen die Subagenten die Superklassenzuordnung des Koordinatoragenten.

Die folgende Abbildung zeigt die Beziehung zwischen Verbindungen, Workloads und Servicesuperklassen. Verbindungen, die die Definition von Workload A erfüllen, werden der Servicesuperklasse 1 zugeordnet; Verbindungen, die die Definition von Workload B oder C erfüllen, werden der Servicesuperklasse 2 zugeordnet; Verbindungen, die die Definition von Workload D erfüllen, werden der Servicesuperklasse `SYSDEFAULTUSERCLASS` zugeordnet.

Benutzerverbindungen



Interne DB2-Verwaltungsverbindungen



Interne DB2-System-entitäten und -verbindungen



Abbildung 11. Zuordnung von Datenbankverbindungen zu einer Servicesuperklasse

Wenn Sie eine komplexere Workload-Management-Konfiguration haben, wollen Sie eventuell Aktivitäten basierend auf dem Aktivitätstyp oder einem anderen Aktivitätsattribut unterschiedlich behandeln. Beispielsweise wollen Sie eventuell eine der folgenden Aktionen ausführen:

- DML in eine andere Serviceunterklasse versetzen als DDL.
- Alle Leseabfragen mit einem geschätzten Aufwand von weniger als 100 Time-rons in eine andere Serviceunterklasse versetzen als alle anderen Leseabfragen.

In einer komplexeren Konfiguration können Sie die Workload einrichten, um Aktivitäten von der Verbindung der Servicesuperklasse zuzuordnen. Dann können Sie mit Arbeitsaktionen (die in einem Arbeitsaktionsset enthalten sind, das auf die Servicesuperklasse angewendet wird) Aktivitäten basierend auf ihrem Typ oder Attribut bestimmten Serviceunterklassen in einer Servicesuperklasse neu zuordnen.

Insbesondere könnten Sie ein Arbeitsaktionsset, das eine Arbeitsaktion MAP ACTIVITY enthält, auf die Servicesuperklasse anwenden. Alle Aktivitäten, die der Servicesuperklasse zugeordnet sind und einer Arbeitsklasse entsprechen, mit der eine Arbeitsaktion MAP ACTIVITY verknüpft ist, werden der in der Arbeitsaktion angegebenen Serviceunterklasse zugeordnet.

Wenn eine Workload eine Aktivität einer Serviceunterklasse zuordnet, ist diese Aktivität nicht von Arbeitsaktionen in einem Arbeitsaktionsset betroffen, das auf die Servicesuperklasse angewendet wird.

- Eine Aktivität kann von einer Workload einer Serviceunterklasse in einer Servicesuperklasse zugeordnet werden.
- Eine Arbeitsaktion, die die Aktivität einer anderen Serviceunterklasse in derselben Servicesuperklasse zuordnet, wird ebenfalls auf die Aktivität angewendet.

Wenn eine Aktivität von einer Workload oder einer Arbeitsaktion nicht einer Serviceunterklasse zugeordnet wird, wird die Aktivität der Standardunterklasse (SYSDEFAULTSUBCLASS) der Servicesuperklasse für diese Aktivität zugeordnet.

Wenn Datenbankaktivitäten ihren entsprechenden Servicesuperklassen und Serviceunterklassen zugeordnet wurden, können Sie Steuerungen für alle Aktivitäten in einer bestimmten Serviceklasse implementieren. Auf der Serviceklassenebene sind Statistikdaten verfügbar, die Sie zum Überwachen der Datenbankaktivitäten in dieser Serviceklasse verwenden können.

Die folgende Abbildung zeigt Verbindungen zur Datenbank, die über Workloads einer Servicesuperklasse oder Serviceunterklasse zugeordnet werden. Informationen dazu, wie Arbeitsaktionen für die Zuordnung von Aktivitäten zu einer Serviceunterklasse verwendet werden, finden Sie in „Arbeitsaktionen und Arbeitsaktionssets“ auf Seite 87.

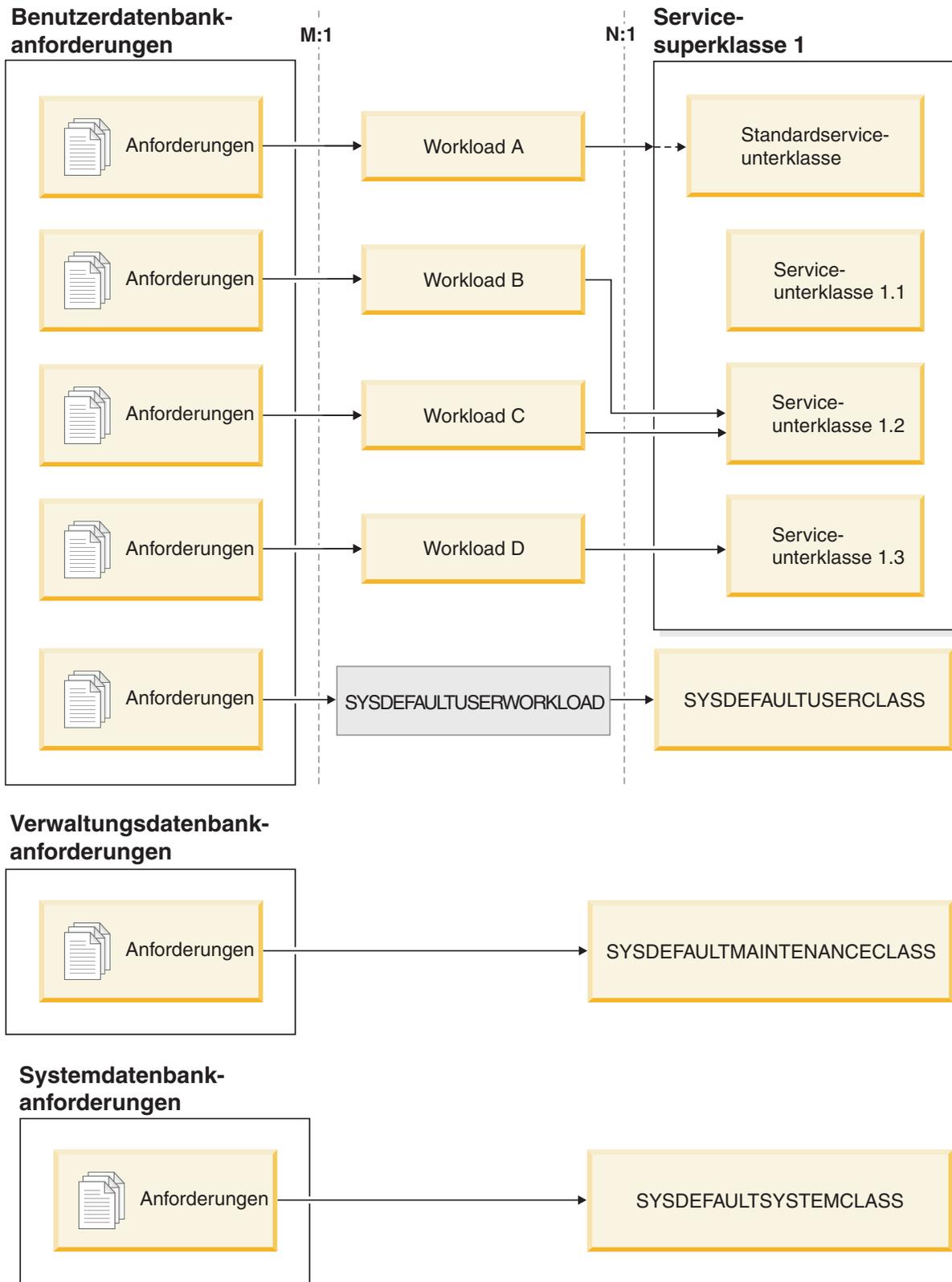


Abbildung 12. Datenbankverbindungen, die einer Servicesuperklasse zugeordnet sind

Verbindungen, die nicht einer benutzerdefinierten Workloaddefinition zugeordnet werden, werden der Standardbenutzerworkloaddefinition (`SYSDEFAULTUSERWORKLOAD`) zugeordnet. Standardmäßig werden Verbindungen von der Stan-

Standardworkloaddefinition (SYSDEFAULTUSERWORKLOAD) der Servicesuperklasse SYSDEFAULTUSERCLASS zugeordnet. Dies ist die Standardservicesuperklasse für Benutzeranforderungen. Sie können die Workload SYSDEFAULTUSERWORKLOAD so ändern, dass sie einer anderen Serviceklasse zugeordnet wird. Interne DB2-Verwaltungsverbindungen werden der Standardservicesuperklasse für Verwaltungsanforderungen (SYSDEFAULTMAINTENANCECLASS) zugeordnet. Interne Systementitäten und Verbindungen werden der Servicesuperklasse SYSDEFAULTSYSTEMCLASS zugeordnet. Dies ist die Standardservicesuperklasse für interne DB2-Verbindungen und -Threads, die Tasks auf Systemebene ausführen.

CPU-Priorität und DB2-Serviceklassen

Bei DB2-Serviceklassen kann jeder Serviceklasse eine relative Agentenpriorität zugeordnet werden. Diese Priorität wird für alle Agenten festgelegt, die in einer Serviceklasse arbeiten, und ist relativ zur Agentenpriorität aller anderen DB2-Agenten. Wenn Sie keine Agentenpriorität für eine Serviceklasse festlegen, haben alle Agenten in dieser Serviceklasse dieselbe Priorität wie alle anderen DB2-Agenten.

Durch das Festlegen der Agentenpriorität für eine DB2-Serviceklasse wird nur die Priorität der Agenten, die in dieser Serviceklasse ausgeführt werden, für neu eintreffende Arbeit angepasst. Andere Threads, die keine Agenten sind und in dieser Serviceklasse ausgeführt werden, nutzen nicht die für die Serviceklasse festgelegten Agentenpriorität.

Wenn Sie DB2-Serviceklassen mit AIX Workload Manager integrieren, können Sie mit AIX Workload Manager die CPU-Prioritäten angeben, die für die Betriebssystemklasse verwendet werden sollen. Die DB2-Serviceklasse übernimmt dann diesen Wert über den Wert von OUTBOUND CORRELATOR der DB2-Serviceklasse. In dieser Situation steuert die CPU-Priorität, die Sie über den Betriebssystem-Workload-Manager angeben, die Priorität für Agenten, die in der DB2-Serviceklasse ausgeführt werden. Jede andere Einstellung der Serviceklassenagentenpriorität wird ignoriert.

Der Konfigurationsparameter des Datenbankmanagers **agentpri** stellt die absolute CPU-Priorität aller Agenten in einer DB2-Instanz auf einen festen Wert ein. Wenn die absolute Priorität für einen Agenten festgelegt ist, kann seine relative Priorität nicht geändert werden. Aus diesem Grund ist **agentpri** nicht kompatibel mit der Serviceklassenagentenpriorität oder der Integration mit AIX Workload Manager. Wenn **agentpri** nicht auf den Standardwert eingestellt ist, haben die Serviceklassenagentenpriorität und AIX Workload Manager keine Auswirkung auf die Priorität von Agenten. Sie sollten diesen veralteten Konfigurationsparameter nicht in einer Workload-Management-Konfiguration verwenden.

Vorablesezugriffspriorität bei Serviceklassen

Vorablesefunktionen rufen Daten vom Datenträger ab und speichern diese Daten in Pufferpools, so dass Agenten schnell darauf zugreifen können. Beim DB2-Workload-Management kann jeder Servicesuperklasse und -unterklasse eine andere Vorablesezugriffspriorität zugeordnet werden.

Agenten senden Vorableseanforderungen an die Vorablesewarteschlange der Datenbank. Die Vorablesefunktionen nehmen diese Vorableseanforderungen aus der Warteschlange und legen dann die Daten in den Pufferpools ab. Wenn ein Agent bestimmte Daten benötigt, prüft er zuerst die Pufferpools, um festzustellen, ob die Daten dort verfügbar sind. Ist dies nicht der Fall, ruft der Agent die Daten vom

Datenträger ab. Vorablesefunktionen führen aufwendige Platten-E/A-Operationen aus, so dass die Agenten parallel für die Ausführung von Rechenarbeit frei werden.

Wenn Sie eine Vorablesewarteschlange verwenden, haben alle Vorableseanforderungen dieselbe Priorität und werden nach dem First-In/First-Out-Prinzip verarbeitet. Diese Methode funktioniert gut bei Datenbanken, bei denen alle Verbindungen die gleiche Priorität haben. In einigen Situationen hat eine Datenbank jedoch Verbindungen mit unterschiedlichen Prioritäten. Möglicherweise sollen die Vorablesefunktionen Anforderungen von Verbindungen mit höherer Priorität zuerst bearbeiten, bevor sie Verbindungen mit geringerer Priorität bedienen. Sie können ein Priorisierungsschema für den Vorablesezugriff implementieren, indem Sie drei unterschiedliche Vorablesewarteschlangen verwenden: HIGH (hoch), MEDIUM (mittel) und LOW (niedrig). In dieser Situation verarbeiten Vorablesefunktionen zuerst die Vorableseanforderungen in der Warteschlange mit hoher Priorität. Wenn diese Warteschlange leer ist, verarbeiten die Vorablesefunktionen die Warteschlange mit mittlerer Priorität. Wenn die beiden Warteschlangen mit hoher und mit mittlerer Priorität leer sind, verarbeiten die Vorablesefunktionen schließlich Anforderungen in der Warteschlange mit niedriger Priorität.

Für jede Verbindung, die an eine Serviceklasse weitergeleitet wird, werden die Vorableseanforderungen entsprechend der Vorablesezugriffspriorität verarbeitet, die der Serviceklasse zugeordnet ist. Jeder Serviceklasse kann eine von drei Vorablesezugriffsprioritäten zugeordnet werden: HIGH, MEDIUM und LOW. Sie legen die Vorablesezugriffspriorität einer Serviceklasse mit dem Schlüsselwort PREFETCH PRIORITY in der Anweisung CREATE SERVICE CLASS oder ALTER SERVICE CLASS fest.

Durch Angabe von DEFAULT für eine Servicesuperklasse wird für diese Servicesuperklasse als Vorablesezugriffspriorität MEDIUM (mittel) festgelegt. Sie können eine andere Vorablesezugriffspriorität für jede Serviceunterklasse in der Servicesuperklasse angeben. Wenn Sie jedoch die Standardpriorität für den Vorablesezugriff für die Serviceunterklasse verwenden, übernimmt die Serviceunterklasse die Vorablesezugriffspriorität von ihrer Servicesuperklasse.

Vorablesezugriffsanforderungen mit hoher Priorität werden vor Vorablesezugriffsanforderungen mit mittlerer Priorität verarbeitet und diese wiederum vor Vorablesezugriffsanforderungen mit niedriger Priorität.

Status von Verbindungen und Aktivitäten in einer Serviceklasse

Serviceklassen erfassen Verbindungsstatistikdaten für jede Serviceklasse. Sie können sehen, welche Verbindungen und Aktivitäten in einer Serviceklasse enthalten sind, und sich den Status der Verbindung oder Aktivität ansehen.

Status einer Verbindung

Eine Verbindung in einer Serviceklasse kann die folgenden Status haben:

Verbunden

Die Verbindung wurde erfolgreich mit der Datenbank verbunden, ist aber noch nicht mit der Workload und der Servicesuperklasse verknüpft.

Zugeordnet

Die Verbindung ist einer Workload zugeordnet und ist mit einer Servicesuperklasse verknüpft. Die Verbindung kann jetzt Aktivitäten zur Ausführung übergeben.

Temporär

Die Verbindung versucht, eine Serviceklasse zu verknüpfen, für die der Verbindungsschwellenwert erreicht ist. Die Verbindung wird für die Verknüpfung mit der Serviceklasse in die Warteschlange gestellt. Wenn die Serviceklasse nicht gegen ihren Verbindungsschwellenwert verstößt, wird die Verbindung mit der Serviceklasse verknüpft. Eine Verbindung mit dem Status 'Temporär' kann keine Aktivitäten zur Ausführung übergeben.

Beendigung

Die Verbindung hat eine Verbindungsrücksetzung vom Client empfangen oder wird zwangsweise oder auf Grund einer Fehlerbedingung beendet.

Status einer Aktivität

Eine Aktivität in einer Serviceklasse kann die folgenden Status haben:

Wird initialisiert

Die Aktivität wurde erstellt und wird zur Ausführung vorbereitet.

In Warteschlange gestellt

Die Aktivität kann auf Grund eines Schwellenwerts für gemeinsamen Zugriff auf Datenbank- oder Serviceklassenebene nicht ausgeführt werden. Die Aktivität wird in die Warteschlange gestellt, bis sie ausgeführt werden kann.

Anmerkung: Wenn unter dem Betriebssystem AIX eine Aktivität in der Warteschlange SQL4297N empfängt, stellen Sie sicher, dass auf dem DB2-Client und -Datenserver der folgende APAR installiert ist:

- Bei AIX 5.3: IY89429
- Bei AIX 5.2: IY89387

Wird ausgeführt

Die Aktivität wird ausgeführt.

Von Query Patroller in Warteschlange gestellt

Die Aktivität wird von Query Patroller in die Warteschlange gestellt.

Beendigung

Die Aktivität wird beendet.

Entitäten auf Systemebene, die nicht von Serviceklassen verfolgt werden

Serviceklassen werden zur Überwachung und Steuerung von Objekten auf Datenbankebene verwendet. Nicht alle DB2-Entitäten arbeiten jedoch direkt in einer Datenbank.

Da Serviceklassen in einer Datenbank arbeiten und in den Katalogtabellen der Datenbank gespeichert werden, können Entitäten, die nicht in einer Datenbank arbeiten, nicht von Serviceklassen verfolgt werden. Entitäten auf Instanzebene, z. B. die Systemcontroller- und Diagnoseмонитordämonen, arbeiten auf Instanzebene und sind mit keiner Datenbank direkt verknüpft. Agenten, die Instanzzuordnungen und Gatewayverbindungen durchführen, werden ebenfalls nicht von Serviceklassen verfolgt. Da Instanzzuordnungsagenten und Gatewayagenten nicht in einer Datenbank arbeiten, werden sie nicht von Serviceklassen verfolgt.

Die folgende Liste ist eine nicht vollständige Liste von Entitäten, die nicht innerhalb einer Datenbank arbeiten und daher nicht von Serviceklassen verfolgt werden:

- DB2-Systemcontroller (db2sysc)
- IPC-Listener (db2ipccm)
- TCP-Listener (db2tcpcom)
- FCM-Dämonen (db2fcms, db2fcmr)
- DB2-Resynchronisationsagenten (db2resync)
- Inaktive Agenten (Agenten ohne Datenbankzuordnung)
- Instanzzuordnungsagenten
- Gatewayagenten
- Alle anderen EDUs (Engine Dispatchable Units - von der Steuerkomponente zuteilbare Einheiten) auf Instanzebene

Arbeiten mit Serviceklassen

Erstellen einer Serviceklasse

Mit der DDL-Anweisung `CREATE SERVICE CLASS` können Sie Servicesuperklassen und darunter Serviceunterklassen erstellen.

Zum Ändern einer Serviceklasse benötigen Sie die Berechtigung `DBADM`, `SYSADM` oder `SYSCNTRL`.

Weitere Voraussetzungen werden auch in den folgenden Themen behandelt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Gehen Sie wie folgt vor, um eine Serviceklasse zu erstellen:

1. Geben Sie mindestens eines der folgenden Merkmale für die Serviceklasse in der Anweisung `CREATE SERVICE CLASS` an:
 - Geben Sie den Namen der Serviceklasse an:

Anmerkung: Sobald der Name einer Serviceklasse festgelegt ist, kann er nicht mehr geändert werden.

- Wenn Sie eine Servicesuperklasse erstellen, muss der Name unter allen Servicesuperklassen in der Datenbank eindeutig sein.

Beim Erstellen einer Servicesuperklasse wird die zugehörige Standard-serviceunterklasse automatisch erstellt. Erst nach dem Erstellen einer Servicesuperklasse können Sie darunter andere Serviceunterklassen erstellen.

- Wenn Sie eine Serviceunterklasse erstellen, muss der Name unter allen Serviceunterklassen in der Servicesuperklasse eindeutig sein. Eine Serviceunterklasse kann nicht denselben Namen haben wie die zugehörige Servicesuperklasse.
- Wenn Sie eine Serviceunterklasse erstellen, geben Sie den Namen der übergeordneten Servicesuperklasse an. Nachdem eine Serviceunterklasse unter einer Servicesuperklasse erstellt wurde, kann sie nicht mehr einer anderen Servicesuperklasse zugeordnet werden.
- Optional: Geben Sie die Agentenpriorität für die Serviceklasse an. Wenn die Agentenpriorität auf `DEFAULT` eingestellt ist, erhalten Agenten in der Serviceklasse die Priorität, die das Betriebssystem allen DB2-Threads zuordnet. Wenn Sie den Parameter `AGENT PRIORITY` auf einen anderen Wert als `DEFAULT` setzen, werden die Agententhreads auf eine Priorität entsprechend

der Standardpriorität plus dem Wert gesetzt, der am Anfang der nächsten Aktivität festgelegt wird. Wenn die Standardpriorität z. B. den Wert 20 hat und Sie die Agentenpriorität auf -10 setzen, ergibt sich eine Agentenpriorität von $20 + (-10) = 10$.

Eine Agentenpriorität von DEFAULT wird als numerischer Wert -32768 bewertet.

Unter Linux und UNIX sind die Werte -32768 und -20 bis 20 zulässig. (Ein negativer Wert bedeutet eine höhere relative Priorität.) Auf Windows-basierten Plattformen sind die Werte -32768 und -6 bis 6 zulässig. (Ein negativer Wert bedeutet eine höhere relative Priorität.)

- Geben Sie die abgehende Korrelatorzeichenfolge an, wenn Sie die DB2-Serviceklasse einer AIX-Serviceklasse zuordnen wollen. Ein Nullwert gibt an, dass es keine externe WLM-Serviceklassenzuordnung gibt.
Wenn der abgehende Korrelator festgelegt ist, werden alle Threads in der DB2-Serviceklasse mit Hilfe des abgehenden Korrelators zu Beginn der nächsten Aktivität dem Betriebssystem-Workload-Manager zugeordnet.
Wenn der abgehende Korrelator für eine Serviceunterklasse auf NONE gesetzt ist und der abgehende Korrelator für die zugehörige Servicesuperklasse festgelegt ist, übernimmt die Serviceunterklasse den für die Servicesuperklasse angegebenen abgehenden Korrelator.
- Geben Sie die Priorität für den Vorablesezugriff an. Sie können die Priorität angeben, mit der Agenten in der Serviceklasse ihre Vorablesezugriffsanforderungen übergeben. Je nach angegebenem Wert werden die Vorablesezugriffsanforderungen an die Vorablesewarteschlange mit hoher, mittlerer oder niedriger Priorität weitergeleitet. Standardmäßig ist die Vorablesezugriffspriorität auf MEDIUM eingestellt.
- Geben Sie die zu erfassenden Aktivitätsdaten an. Wenn die Erfassung von Aktivitätsdaten aktiviert ist, werden am Ende der Aktivität Aktivitätsinformationen von der Koordinatorpartition an den entsprechenden Ereignismonitor gesendet. Wenn Sie wollen, können Sie Daten von allen Datenbankpartitionen, auf denen die Aktivität ausgeführt wurde, in den Ereignismonitor schreiben, einschließlich Informationen zur ausgeführten Anweisung, ihrer Kompilierumgebung sowie alle zutreffenden Eingabedatenwerte. Sie können auch angeben, dass keine Aktivitätsdaten erfasst werden sollen. Standardmäßig werden keine Aktivitätsdaten erfasst.
- Geben Sie die zu erfassenden zusammengefassten Aktivitätsinformationen an. Die zusammengefassten Aktivitätsinformationen für die Serviceklasse ändern sich erst nach dem Durchführen eines Commits für die Operation ALTER SERVICE CLASS.
- Geben Sie an, ob die Histogramme geändert werden sollen, die von einer Serviceunterklasse verwendet werden, für die die Erfassung von zusammengefassten Aktivitätsdaten über COLLECT AGGREGATE ACTIVITY DATA oder die Erfassung von zusammengefassten Anforderungsdaten über COLLECT AGGREGATE REQUEST DATA aktiviert ist. Durch die Aktualisierung der von einer Serviceunterklasse verwendeten Histogrammschablonen werden die entsprechenden Zeilen in der Sicht SYSCAT.HISTOGRAMTEMPLATEUSE aktualisiert, die die Histogrammschablonen anzeigt, auf die eine Serviceklasse oder eine Arbeitsaktion verweist. Weitere Informationen zu Histogrammen und Histogrammschablonen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.
- Geben Sie an, ob die Serviceklasse aktiviert oder inaktiviert ist.
 - Wenn eine Serviceklasse als aktiviert erstellt wird (Standard), können der Serviceklasse Verbindungen und Aktivitäten zugeordnet werden.

Wenn eine Serviceklasse als inaktiviert erstellt wird, werden neue Verbindungen und Aktivitäten, die ihr zugeordnet werden sollen, zurückgewiesen.

- Wenn Sie eine Servicesuperklasse als inaktiviert erstellen, verhalten sich alle Serviceunterklassen, die Sie mit dieser Servicesuperklasse verknüpfen, als wären sie inaktiviert, selbst wenn sie bei der Abfrage der Sicht SYSCAT.SERVICECLASSES als aktiviert angezeigt werden.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Serviceklasse zur Sicht SYSCAT.SERVICECLASSES hinzugefügt.

Ändern einer Serviceklasse

Verwenden Sie die Anweisung ALTER SERVICE CLASS, wenn Sie die Definition einer Serviceklasse ändern wollen.

Zum Ändern einer Serviceklasse benötigen Sie die Berechtigung DBADM, SYSADM oder SYSCNTRL.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Aktivitäten, die bereits Ressourcen bezogen haben und ausgeführt werden, sind von der Anweisung ALTER SERVICE CLASS nicht betroffen. Diese Aktivitäten halten ihre Ressourcen und werden bis zu ihrer Fertigstellung ausgeführt. Wenn jedoch während einer Operation ALTER SERVICE CLASS eine Subagentenanforderung an eine ferne Datenbankpartition gesendet wird, wird die Serviceklassendefinition vom Koordinatoragenten gesehen, und der Subagent kann sich unterscheiden. Betrachten wir das folgende Beispiel, bei dem die Vorablesezugriffspriorität für die Serviceklasse anfänglich auf MEDIUM eingestellt ist:

Tabelle 8. Unterschiede zwischen den Sichten eines Koordinatoragenten und eines Subagenten für eine geänderte Serviceklasse

Ereignisreihenfolge	Verbindung 1	Verbindung 2
1	Der koordinierende Agent sendet eine DSS-Anforderung an die ferne Partition. (Die Vorablesezugriffspriorität der Serviceklasse wurde zuvor auf MEDIUM gesetzt.)	
2		ALTER SERVICE CLASS wird abgesetzt. Die Vorablesezugriffspriorität wird auf HIGH gesetzt.
3		COMMIT wird abgesetzt. (Das geänderte Serviceklassenmerkmal wird auf der Katalogpartition festgeschrieben und in allen Datenbankpartitionen in den Hauptspeicher geladen.)

Tabelle 8. Unterschiede zwischen den Sichten eines Koordinatoragenten und eines Subagenten für eine geänderte Serviceklasse (Forts.)

Ereignisreihenfolge	Verbindung 1	Verbindung 2
4	Der ferne Subagent empfängt die DSS-Anforderung. Jetzt sieht der Subagent die neue Vorablesezugriffspriorität HIGH für die Serviceklassen- definition.	

Die in der Tabelle oben beschriebene Situation ist temporär und betrifft nur Verbindungen, die während der Operation ALTER SERVICE CLASS Subagentenanforderungen absetzen. Alle neue Verbindungen sehen die aktualisierte Serviceklassendefinition mit der Vorablesezugriffspriorität HIGH.

Gehen Sie wie folgt vor, um eine Serviceklasse zu ändern:

1. Geben Sie mindestens eines der folgenden Merkmale für die Serviceklasse in der Anweisung ALTER SERVICE CLASS an:
 - Geben Sie an, ob die Serviceklasse aktiviert oder inaktiviert ist. Wenn Sie eine Serviceklasse von aktiviert in inaktiviert ändern, bleiben alle bestehenden Verbindungen oder Aktivitäten in der Serviceklasse und verwenden bis zu ihrer Fertigstellung weiterhin die zuvor zugeordneten Ressourcen. Sie können eine Serviceklasse inaktivieren, wenn die Arbeit, die in der Serviceklasse eingeht, das System überlastet oder wenn Sie alle in der Serviceklasse eingehende Arbeit zurückweisen wollen.

Beim Inaktivieren einer Servicesuperklasse geschieht Folgendes:

- a. Die Servicesuperklasse wird inaktiviert.
- b. Die zugehörigen Serviceunterklassen werden inaktiviert.

Die Serviceunterklassen sind nur inaktiviert, während ihre Servicesuperklasse inaktiviert ist. Wenn die Servicesuperklasse aktiviert wird, werden die Serviceunterklassen auf ihren vorherigen Zustand gemäß der Definition in der Katalogtabelle zurückgesetzt.

Wenn eine Serviceunterklasse inaktiviert wird, sind davon weder die zugehörige Servicesuperklasse noch andere Serviceunterklassen mit dieser Servicesuperklasse betroffen.

Eine Standard-Serviceunterklasse kann nicht explizit inaktiviert werden. Sie müssen die zugehörige Servicesuperklasse inaktivieren, um zu verhindern, dass neue Anforderungen unter einer Standard-Serviceunterklasse ausgeführt werden.

- Geben Sie die Agentenpriorität für die Serviceklasse an. Wenn die Agentenpriorität auf DEFAULT eingestellt ist, erhalten Agenten in der Serviceklasse die Priorität, die das Betriebssystem allen DB2-Threads zuordnet. Wenn Sie den Parameter AGENT PRIORITY auf einen anderen Wert als DEFAULT setzen, werden die Agententhreads auf eine Priorität entsprechend der Standardpriorität plus dem Wert gesetzt, der am Anfang der nächsten Aktivität festgelegt wird. Wenn die Standardpriorität z. B. den Wert 20 hat und Sie die Agentenpriorität auf -10 setzen, ergibt sich eine Agentenpriorität von $20 + (-10) = 10$.

Eine Agentenpriorität von DEFAULT wird als numerischer Wert -32768 bewertet.

Unter Linux und UNIX sind die Werte -32768 und -20 bis 20 zulässig. (Ein negativer Wert bedeutet eine höhere relative Priorität.) Auf Windows-basier-

ten Plattformen sind die Werte -32768 und -6 bis 6 zulässig. (Ein negativer Wert bedeutet eine höhere relative Priorität.)

- Geben Sie die Priorität für den Vorablesezugriff an. Sie können die Priorität angeben, mit der Agenten in der Serviceklasse ihre Vorablesezugriffsanforderungen übergeben. Je nach angegebenem Wert werden die Vorablesezugriffsanforderungen an die Vorablesewarteschlange mit hoher, mittlerer oder niedriger Priorität weitergeleitet. Standardmäßig ist die Vorablesezugriffspriorität auf MEDIUM eingestellt. Wenn die Vorablesezugriffspriorität nach einer Vorablesezugriffsanforderung geändert wird, ändert die Anforderung ihre Priorität nicht.
- Geben Sie die abgehende Korrelatorzeichenfolge an, wenn Sie die DB2-Serviceklasse einer AIX-Serviceklasse zuordnen wollen. Ein Nullwert gibt an, dass es keine externe WLM-Serviceklassenzuordnung gibt.

Wenn der abgehende Korrelator in einen Nullwert geändert wird, heben alle Threads in der DB2-Serviceklasse am Anfang der nächsten Aktivität ihre Zuordnung zum Betriebssystem-Workload-Manager auf.

Wenn der abgehende Korrelator für eine Serviceunterklasse auf NONE gesetzt ist und der abgehende Korrelator für die zugehörige Servicesuperklasse festgelegt ist, übernimmt die Serviceunterklasse den für die Servicesuperklasse angegebenen abgehenden Korrelator.

Wenn eine Servicesuperklasse einen abgehenden Korrelator verwendet, muss die Agentenpriorität der Servicesuperklasse auf DEFAULT gesetzt werden.

Wenn eine Serviceunterklasse einen abgehenden Korrelator verwendet (entweder explizit als Teil der Serviceunterklassendefinition oder implizit durch Übernahme von der Servicesuperklasse), muss die Agentenpriorität der Serviceunterklasse auf DEFAULT gesetzt werden.

- Geben Sie die zu sammelnden Aktivitätsdaten an. Wenn die Sammlung von Aktivitätsdaten aktiviert ist, werden am Ende der Aktivität Aktivitätsinformationen von der Koordinatorpartition an den entsprechenden Ereignismonitor gesendet. Wenn Sie wollen, können Sie Daten von allen Datenbankpartitionen, auf denen die Aktivität ausgeführt wurde, in den Ereignismonitor schreiben, einschließlich Informationen zur ausgeführten Anweisung, ihrer Kompilierumgebung sowie alle zutreffenden Eingabedatenwerte. Sie können auch angeben, dass keine Aktivitätsdaten gesammelt werden sollen. Standardmäßig werden keine Aktivitätsdaten gesammelt.
 - Geben Sie die zu sammelnden zusammengefassten Aktivitätsinformationen an. Die zusammengefassten Aktivitätsinformationen für die Serviceklasse ändern sich erst nach dem Durchführen eines Commits für die Operation ALTER SERVICE CLASS.
 - Geben Sie an, ob die Histogramme geändert werden sollen, die von einer Serviceunterklasse verwendet werden, für die die Sammlung von zusammengefassten Aktivitätsdaten über COLLECT AGGREGATE ACTIVITY DATA oder die Sammlung von zusammengefassten Anforderungsdaten über COLLECT AGGREGATE REQUEST DATA aktiviert ist. Durch die Aktualisierung der von einer Serviceunterklasse verwendeten Histogrammschablonen werden die entsprechenden Zeilen in der Sicht SYSCAT.HISTOGRAMTEMPLATEUSE aktualisiert, die die Histogrammschablonen anzeigt, auf die eine Serviceklasse oder eine Arbeitsaktion verweist. Weitere Informationen zu Histogrammen und Histogrammschablonen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Serviceklasse in der Sicht SYSCAT.SERVICECLASSES aktualisiert.

Löschen einer Serviceklasse

Mit der DDL-Anweisung `DROP SERVICE CLASS` können Sie Serviceklassen löschen.

Zum Löschen einer Serviceklasse benötigen Sie die Berechtigung `DBADM`, `SYSADM` oder `SYSCNTRL`.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Im Folgenden sind die Einschränkungen für das Löschen einer Serviceklasse aufgeführt:

- Die Standardservicesuperklassen (`SYSDEFAULTUSERCLASS`, `SYSDEFAULTMAINTENANCECLASS` und `SYSDEFAULTSYSTEMCLASS`) und ihre zugehörigen Serviceunterklassen können nicht gelöscht werden. Standardservicesuperklassen und ihre zugehörigen Serviceunterklassen können nur durch das Löschen der Datenbank gelöscht werden.
- Eine Serviceklasse kann nicht gelöscht werden, wenn eine der folgenden Bedingungen auf die Serviceklasse zutrifft:
 - Sie ist aktiviert.
 - Eine Workload, eine Arbeitsaktion oder ein Schwellenwert verweist auf sie.
 - Der Serviceklasse ist derzeit eine Verbindung oder Aktivität zugeordnet.

Gehen Sie wie folgt vor, um eine Serviceklasse zu löschen:

1. Inaktivieren Sie die Serviceklasse mit der Anweisung `ALTER SERVICE CLASS`. Wenn Sie eine Servicesuperklasse löschen, werden mit dieser Aktion alle Serviceunterklassen inaktiviert, die mit der Servicesuperklasse verknüpft sind. Durch das Inaktivieren einer Serviceklasse wird verhindert, dass weitere Aktivitäten mit der Serviceklasse verknüpft werden. Setzen Sie nach dem Inaktivieren der Serviceklasse eine Anweisung `COMMIT` ab.
Aktivitäten, die bereits unter der Serviceklasse ausgeführt werden, werden weiter ausgeführt. Mit der Tabellenfunktion `WLM_GET_SERVICE_CLASS_AGENTS` können Sie Agenten auflisten, die derzeit einer Serviceklasse zugeordnet sind. Wenn diese Aktivitäten nicht abgeschlossen werden sollen, können Sie anhand der von der Tabellenfunktion zurückgegebenen Anwendungs-ID und mit dem Befehl `FORCE APPLICATION` die Trennung dieser Anwendungen von der Datenbank erzwingen.
2. Mit der Anweisung `DROP WORKLOAD` können Sie alle Workloads löschen, die der Serviceklasse zugeordnet sind. Setzen Sie nach dem Löschen jeder Workload eine Anweisung `COMMIT` ab.
3. Löschen Sie alle zutreffenden Arbeitsaktionen, die der zu löschenden Serviceklasse zugeordnet sind:
 - Wenn Sie eine Servicesuperklasse löschen und dieser ein Arbeitsaktionsset zugeordnet ist, löschen Sie dieses Arbeitsaktionsset mit der Anweisung `DROP WORK ACTION SET`. Setzen Sie nach dem Löschen des Arbeitsaktionssets eine Anweisung `COMMIT` ab.
 - Wenn Sie eine Serviceunterklasse löschen und dieser eine Arbeitsaktion zugeordnet ist, löschen Sie die Arbeitsaktion mit dem Schlüsselwort `DROP WORK ACTION` der Anweisung `ALTER WORK ACTION SET`. Alternativ können Sie auch das Arbeitsaktionsset, das die der Serviceunterklasse zugeordnete Arbeitsaktion enthält, mit der Anweisung `DROP WORK ACTION SET` löschen. Setzen Sie nach dem Löschen jeder Arbeitsaktion oder nach dem Löschen des Arbeitsaktionssets eine Anweisung `COMMIT` ab.

4. Löschen Sie mit der Anweisung DROP THRESHOLD alle Schwellenwerte, die der zu löschenden Serviceklasse zugeordnet sind. Setzen Sie nach dem Löschen jedes Schwellenwerts eine Anweisung COMMIT ab.
5. Gehen Sie je nach dem zu löschenden Objekt wie folgt vor:
 - Wenn Sie eine Serviceunterklasse löschen wollen, verwenden Sie die Anweisung DROP SERVICE CLASS zum Löschen der Serviceunterklasse.
 - Wenn Sie eine Servicesuperklasse löschen wollen, verwenden Sie die Anweisung DROP SERVICE CLASS, um alle ihr zugeordneten Serviceunterklassen zu löschen, und setzen Sie nach dem Löschen jeder Serviceunterklasse eine Anweisung COMMIT ab. Setzen Sie dann die Anweisung DROP SERVICE CLASS ab, um die Servicesuperklasse zu löschen.

Anmerkung: Die Standardserviceunterklasse für die Servicesuperklasse kann nicht manuell gelöscht werden. Die Standardserviceunterklasse für eine Servicesuperklasse wird beim Löschen der Servicesuperklasse gelöscht.

6. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Serviceklasse aus der Sicht SYSCAT.SERVICECLASSES entfernt.

Kapitel 3. Workloads

Eine Workload ist eine von Ihnen definierte Entität, mit der übergebene Datenbankarbeit basierend auf deren Quelle und entsprechend der Datenbankverbindungsattribute angegeben wird, so dass diese später verwaltet werden kann. Bei Verwendung von Workloads können Sie Arbeit einer Servicesuperklasse oder einer Serviceunterklasse in einer Servicesuperklasse zuordnen.

Ein Workloadobjekt besteht aus den folgenden Elementen:

- Einem Workloadnamen, der in der Datenbank eindeutig ist.
- Einer eindeutigen ganzzahligen Kennung für die Workload, die vom Datenserver generiert und intern verwendet wird.
- Datenbankverbindungsattributen, die für eine Datenbankverbindung oder Anwendung, die der Workload zugeordnet werden soll, erfüllt sein müssen.
- Workloadattributen, einschließlich der folgenden Elemente:
 - Dem Namen der DB2-Serviceklasse, der die Workload zugeordnet werden soll. Wenn Sie keinen Serviceklassennamen angeben, wird die Workload der Standardbenutzerserviceklasse (SYSDEFAULTUSERCLASS) zugeordnet.
 - Einem Wert, der angibt, ob ein Vorkommen der Workload auf die Datenbank zugreifen darf. Standardmäßig können Workloadvorkommen auf die Datenbank zugreifen. Weitere Informationen finden Sie in „Erlauben des Datenbankzugriffs für Vorkommen einer Workload“ auf Seite 55 bzw. „Verhindern des Datenbankzugriffs für Vorkommen einer Workload“ auf Seite 56.
 - Einem Wert, der angibt, ob die Workload inaktiviert ist. Standardmäßig ist die Workload aktiviert. Weitere Informationen finden Sie in „Aktivieren einer Workload“ auf Seite 56 und „Inaktivieren einer Workload“ auf Seite 57.
- Der Auswertungsreihenfolge oder Position der Workload relativ zu anderen Workloads auf dem Datenserver. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.

Mit der Anweisung `CREATE WORKLOAD` können Sie Workloads erstellen. Mit der Anweisung `ALTER WORKLOAD` können Sie Workloads ändern. Mit der Anweisung `DROP WORKLOAD` können Sie Workloads löschen.

Sie können Ihre Workloads durch Abfrage der Sicht `SYSCAT.WORKLOADS` anzeigen. Die Verbindungsattribute, die Sie für die einzelnen Workloads angegeben haben, können Sie durch Abfrage der Sicht `SYSCAT.WORKLOADCONNATTR` anzeigen. Sie können durch Abfrage der Sicht `SYSCAT.WORKLOADAUTH` anzeigen, wer berechtigt ist, eine Workload zu verwenden.

Die unten aufgeführten Datenbankverbindungsattribute werden unterstützt. Sie müssen mindestens ein Datenbankverbindungsattribut in der Workload angeben. Jedes Verbindungsattribut kann einen oder mehrere Werte haben. Wenn Sie keinen Wert für ein bestimmtes Verbindungsattribut für die Workload angeben, untersucht der Datenserver dieses Attribut bei der Workloadauswertung nicht.

Tabelle 9. Verbindungsattribute in einer Workloaddefinition

Verbindungsattribut	Beschreibung
Anwendungsname	Der Name der Anwendung, die auf dem Client ausgeführt wird, so wie er dem Datenserver bekannt ist. Der Anwendungsname entspricht dem Wert im Feld Anwendungsname in der Systemmonitorausgabe. Weitere Informationen finden Sie im Monitorelement appl_name .
Systemberechtigungs-ID	Die Berechtigungs-ID des Benutzers, der die Verbindung zur Datenbank hergestellt hat, so wie sie im Sonderregister SYSTEM_USER festgelegt ist. Sie können den Wert von SYSTEM_USER ändern, indem Sie die Verbindung als Benutzer mit einer anderen Berechtigungs-ID herstellen.
Sitzungsberechtigungs-ID	Die Berechtigungs-ID, die für die aktuelle Sitzung der Anwendung verwendet wird, so wie sie im Sonderregister SESSION_USER festgelegt ist. Sie können den Wert von SESSION_USER mit der Anweisung SET SESSION AUTHORIZATION ändern.
Gruppe der Sitzungsberechtigungs-ID	Die Gruppen, zu denen der aktuelle Sitzungsbenutzer gehört.
Rolle der Sitzungsberechtigungs-ID	Die Rollen, die dem aktuellen Sitzungsbenutzer erteilt wurden. Weitere Informationen finden Sie in: <ul style="list-style-type: none"> • Rollen • Anweisung GRANT ROLE • Anweisung REVOKE ROLE
Clientbenutzer-ID	Die Clientbenutzer-ID aus den Clientinformationen, so wie sie im Sonderregister CURRENT CLIENT_USERID (oder CLIENT_USERID) festgelegt ist. Sie können den Wert der Clientbenutzer-ID mit der API sqleseti (Clientinformationen festlegen) ändern.
Clientanwendungsname	Der Anwendungsname aus den Clientinformationen, so wie er im Sonderregister CURRENT CLIENT_APPLNAME (oder CLIENT_APPLNAME) festgelegt ist. Der Clientanwendungsname entspricht dem Wert im Feld Clientanwendungsname für TP-Monitor in der Systemmonitorausgabe. Sie können den Wert des Clientanwendungsnamens mit der API sqleseti ändern.
Clientworkstationname	Der Workstationname aus den Clientinformationen, so wie er im Sonderregister CURRENT CLIENT_WRKSTNNAME (oder CLIENT_WRKSTNNAME) festgelegt ist. Sie können den Wert des Clientworkstationnamens mit der API sqleseti ändern.

Tabelle 9. Verbindungsattribute in einer Workloaddefinition (Forts.)

Verbindungsattribut	Beschreibung
Clientabrechnungszeichenfolge	Die Abrechnungszeichenfolge aus den Clientinformationen, so wie sie im Sonderregister CURRENT CLIENT_ACCTNG (oder CLIENT ACCTNG) festgelegt ist. Sie können den Wert der Clientabrechnungszeichenfolge mit der API sqleseti ändern.

In einer dreistufigen Client/Server-Umgebung wird die Datenbankverbindung vom Anwendungsserver für die Clients hergestellt. Der Anwendungsserver kann die API sqleseti verwenden, um Clientinformationen an den DB2-Datenserver zu übergeben; andernfalls werden nur die Informationen zum Anwendungsserver übergeben, und diese Informationen sind wahrscheinlich für alle über diesen Anwendungsserver geleiteten Clientanforderungen gleich. Durch Angabe von Clientinformationsattributen wie der Clientbenutzer-ID, dem Clientanwendungsnamen, dem Clientworkstationnamen und der Clientabrechnungszeichenfolge in der Workloaddefinition können Sie Benutzer, die auf unterschiedlichen Clients arbeiten, unterschiedlichen Workloads (und unterschiedlichen Serviceklassen) zuordnen.

Am Anfang der ersten UOW (Unit of Work - Arbeitseinheit) wird die Datenbankverbindung einer Workload zugeordnet, wenn ihre Verbindungsattribute den Verbindungsattributen in der Workloaddefinition entsprechen. Wenn sich ein Verbindungsattribut ändert, wird die Datenbankverbindung am Anfang der nächsten UOW einer anderen Workload neu zugeordnet. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.

SYSDEFAULTUSERWORKLOAD ist die Standardworkload. Wenn die Verbindungsattribute keiner von Ihnen definierten Workload zugeordnet werden können, wird die Datenbankverbindung der Standardworkload SYSDEFAULTUSERWORKLOAD zugeordnet, und die Arbeit wird standardmäßig in der Serviceklasse SYSDEFAULTUSERCLASS ausgeführt. Diese Situation tritt in einer neuen Datenbank oder einer neu migrierten Datenbank auf, da außer den Standardworkloads noch keine anderen Workloads vorhanden sind.

Wenn Sie die Nutzungskenndaten Ihrer Umgebung analysieren, können Sie mit der Anweisung CREATE WORKLOAD Ihre eigenen Workloads erstellen und diese bestimmten Serviceklassen zuordnen. Beim Erstellen der Workload definieren Sie sowohl die Werte, die zur Auswertung der Verbindungsattribute bei der Workloadzuordnung verwendet werden, als auch die Reihenfolge, in der die Workload relativ zu anderen Workloads ausgewertet wird. Da mehr als eine Workload mit eingehenden Verbindungsattributen übereinstimmen kann, können Sie durch die Änderung der Auswertungsreihenfolge festlegen, welche übereinstimmende Workload ausgewählt wird. Welche übereinstimmende Workload ausgewählt wird, hängt auch davon ab, ob der Sitzungsbenutzer das Zugriffsrecht USAGE für die Workload hat. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.

Ein Workloadvorkommen ist eine Datenbankverbindung mit Attributen, die mit einer Workloaddefinition übereinstimmen. Nach Abschluss der Workloadzuordnung wird ein Vorkommen einer Workload im Datenserver gestartet und in der in der Workloaddefinition angegebenen Serviceklasse ausgeführt. Dieses Workloadvorkommen bleibt bestehen, bis die Verbindung beendet wird oder bis sich ein Verbindungsattribut ändert. In diesem Fall erfolgt die Workloadneuauswertung am Anfang der nächsten UOW. Die Workloadneuauswertung und -neuzuordnung

erfolgt an den UOW-Grenzen. Ein Workloadvorkommen besteht daher aus mindestens einer UOW in einer Datenbankverbindung, die einer im Datenserver definierten Workload zugeordnet ist. Für jede Workload können auf dem Datenserver gleichzeitig mehrere Workloadvorkommen ausgeführt werden.

Die folgende Abbildung zeigt, wie mehrere Anforderungen in Bezug auf Workloads in der Reihenfolge A, B, C und D ausgewertet und dann bestimmten Workloads zugeordnet und in der zutreffenden Serviceklasse ausgeführt werden. Anforderungen, die mit keiner vorhandenen Workload übereinstimmen, werden der Workload SYSDEFAULTUSERWORKLOAD zugeordnet und standardmäßig in der Servicesuperklasse SYSDEFAULTUSERCLASS ausgeführt. Informationen zu den Aktivitätstypen, die in der Standardverwaltungsklasse und der Standard-systemklasse ausgeführt werden, finden Sie in „Standardservicesuperklassen und -unterklassen“ auf Seite 27.

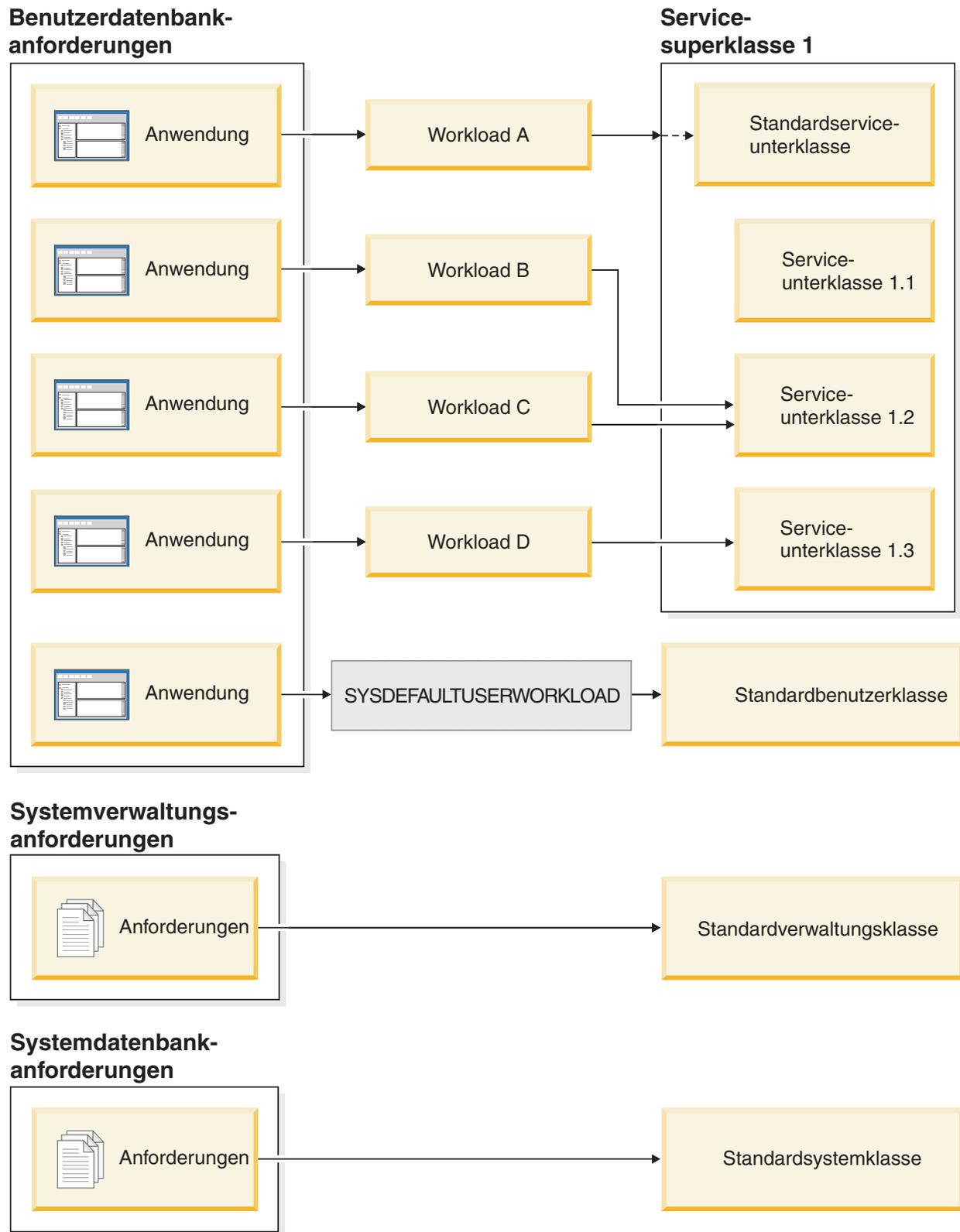


Abbildung 13. Serviceklassen und Workloads

Workloadzuordnung

Am Anfang der ersten UOW (Unit of Work - Arbeitseinheit) nach dem Herstellen der Datenbankverbindung ordnet der Datenserver die Verbindung einer Workload zu, indem er die Verbindungsattribute jeder aktivierten Workload auswertet.

Die Reihenfolge, in der die Workloads ausgewertet werden, wird vom Spaltenwert `EVALUATIONORDER` jeder Workload in der Tabelle `SYSCAT.WORKLOADS` bestimmt. Wenn eine Workload mit übereinstimmenden Verbindungsattributen gefunden wird, prüft der Datenserver, ob der aktuelle Sitzungsbenutzer das Zugriffsrecht `USAGE` für die Workload hat. Hat der Benutzer das Zugriffsrecht `USAGE` für die übereinstimmende Workload, ist die Workloadzuordnung abgeschlossen, und die Verbindung wird dieser Workload zugeordnet. Hat der Benutzer das Zugriffsrecht `USAGE` für die übereinstimmende Workload nicht, wertet der Datenserver weitere Workloads aus, bis er eine übereinstimmende Workload findet, für die der Sitzungsbenutzer das Zugriffsrecht `USAGE` hat. Wenn keine übereinstimmende Workload gefunden wird, versucht der Datenserver, die Workload `SYSDEFAULTUSERWORKLOAD` zu verwenden. Wenn der aktuelle Sitzungsbenutzer nicht das Zugriffsrecht `USAGE` für diese Workload hat, wird `SQL4707N` zurückgegeben, und die UOW wird abgewiesen. Andernfalls wird die Verbindung der Workload `SYSDEFAULTUSERWORKLOAD` zugeordnet.

Sie können die Auswertungsreihenfolge mit dem Schlüsselwort `POSITION` der Anweisung `CREATE WORKLOAD` oder `ALTER WORKLOAD` wie folgt festlegen:

- Durch Angabe der absoluten Position der Workload in der Auswertungsreihenfolge (siehe folgendes Beispiel):

```
CREATE WORKLOAD...POSITION AT 2
```

`POSITION AT 2` bedeutet, dass die Workload an zweiter Stelle in der Auswertungsreihenfolge platziert wird. Eine übereinstimmende Workload, die in der Auswertungsreihenfolge höher platziert ist, wird zuerst ausgewertet. Wenn also die Workloads an Position 2 und Position 3 übereinstimmen, wird die Workload an Position 2 vor der Workload an Position 3 ausgewertet.

Wenn die in der Anweisung `CREATE WORKLOAD` oder `ALTER WORKLOAD` festgelegte Position größer als die Gesamtzahl vorhandener Workloads ist, wird die Workload als vorletzte vor der Workload `SYSDEFAULTUSERWORKLOAD` in der Auswertungsreihenfolge platziert. Dies hat dieselbe Auswirkung wie die Angabe von `POSITION LAST` in der Anweisung `CREATE WORKLOAD` oder `ALTER WORKLOAD`.

- Durch Verwendung des Schlüsselworts `POSITION BEFORE workloadname` oder `POSITION AFTER workloadname`, wobei `workloadname` eine vorhandene Workload ist. Dieses Schlüsselwort gibt die Position einer neuen oder geänderten Workload relativ zu einer anderen Workload in der Auswertungsreihenfolge an (siehe folgendes Beispiel):

```
ALTER WORKLOAD...POSITION BEFORE workload2
```

Wenn Sie das Schlüsselwort `POSITION` nicht angeben, wird die neue Workload standardmäßig nach allen anderen definierten Workloads in der Auswertungsreihenfolge, aber noch vor der Workload `SYSDEFAULTUSERWORKLOAD` platziert, die immer als letzte berücksichtigt wird.

Die Workloadzuordnung wird am Anfang einer neuen UOW erneut ausgewertet, wenn der Datenserver erkennt, dass eines der folgenden Ereignisse aufgetreten ist:

- Ein relevantes Verbindungsattribut hat sich geändert. Eine Liste der Verbindungsattribute, die Sie in einer Workloaddefinition angeben können, fin-

den Sie in der Tabelle in Kapitel 3, „Workloads“, auf Seite 43. Die Neuauswertung der Workload erfolgt auch, wenn die aktuelle Sitzungsberechtigungs-ID sich ändert, weil die Datenbankverbindung auf Grund eines gesicherten Kontexts wechselt. Weitere Informationen finden Sie in Gesicherte Kontexte und gesicherte Verbindungen.

- Sie haben eine Workload erstellt oder geändert.
- Sie haben das Zugriffsrecht USAGE für eine Workload einem Benutzer, einer Gruppe oder einer Rolle erteilt oder das Zugriffsrecht USAGE für eine Workload einem Benutzer, einer Gruppe oder einer Rolle entzogen.

Eine Verbindung kann nicht einer anderen Workload zugeordnet werden, während noch eine Aktivität aktiv ist. Eine Aktivität ist eine Operation, die Ressourcen über mehrere UOWs verwaltet, z. B. eine Ladeoperation, eine gespeicherte Prozedur, eine Tabellenfunktion oder ein WITH HOLD-Cursor. Das aktuelle Workloadvorkommen wird ausgeführt, bis alle Aktivitäten abgeschlossen sind. Die erneute Workloadzuordnung erfolgt dann am Anfang der nächsten UOW.

Eine versuchte (erneute) Workloadzuordnung führt zu einem Fehler SQL4707N, wenn einer der folgenden Fälle vorliegt:

- Der Datenserver versucht, die Verbindung einer Workload zuzuordnen, die keine Berechtigung für den Datenbankzugriff hat. Weitere Informationen finden Sie in „Verhindern des Datenbankzugriffs für Vorkommen einer Workload“ auf Seite 56.
- Der Datenserver versucht, die Verbindung der Workload SYSDEFAULTUSER-WORKLOAD zuzuordnen, aber der aktuelle Sitzungsbenutzer hat nicht das Zugriffsrecht USAGE für diese Workload.

Wenn Sie die Berechtigung DBADM oder SYSADM haben, können Sie Ihre Datenbankverbindung der Standardverwaltungsworkload (SYSDEFAULTADM-WORKLOAD) zuordnen. Weitere Informationen finden Sie in „Zuordnen einer Verbindung zur Standardverwaltungsworkload“ auf Seite 51.

XA-Transaktionen und erneute Workloadzuordnung

XA-Aufrufe wie XA_END (Erfolg), XA-Commit und XA-Rollback setzen einen DB2-Befehl COMMIT oder ROLLBACK ab, wodurch das Ende einer UOW angegeben wird. Da die Workloadneuauswertung am Anfang einer UOW erfolgen kann, können diese XA-Aufrufe die Workloadneuauswertung einleiten, obwohl die Ursache für die Workloadneuauswertung nicht direkt mit der XA-Transaktion selbst zusammenhängt.

Standardworkloads

Die Standardbenutzerworkload (SYSDEFAULTUSERWORKLOAD) und die Standardverwaltungsworkload (SYSDEFAULTADMWORKLOAD) werden bei der Datenbankeinstellung erstellt. Sie können nicht gelöscht werden.

Nach der DB2-Installation oder -Migration auf Version 9.5 oder neuer werden alle Verbindungen der Standardworkload (SYSDEFAULTUSERWORKLOAD) zugeordnet. Verbindungen, die zu dieser Standardworkload gehören, werden der Standardbenutzerservicesuperklasse (SYSDEFAULTUSERCLASS) zugeordnet. Sie können Verbindungen von der Standardworkload benutzerdefinierten Workloads neu zuordnen, um andere benutzerdefinierte Serviceklassen zu verwenden, wenn dies

erforderlich sein sollte. Darüber hinaus können Sie SYSDEFAULTUSERWORKLOAD so ändern, dass sie Verbindungen einer anderen Serviceklasse als SYSDEFAULTUSERCLASS zuordnet.

Sie können die Workload SYSDEFAULTUSERWORKLOAD durch Abfrage der Tabelle SYSCAT.WORKLOADS anzeigen. Die folgende Tabelle zeigt den Eintrag für die Workload SYSDEFAULTUSERWORKLOAD in der Sicht SYSCAT.WORKLOADS. Informationen zum Zuordnen einer Verbindung zur Workload SYSDEFAULTUSERWORKLOAD finden Sie in „Workloadzuordnung“ auf Seite 48.

Tabelle 10. Eintrag SYSDEFAULTUSERWORKLOAD in SYSCAT.WORKLOADS

Spalte	Wert	Modifizierbar mit der Anweisung ALTER WORKLOAD, wenn Sie DBADM- oder SYSADM-Zugriff haben
WORKLOADID	1	Nein
WORKLOADNAME	SYSDEFAULTUSERWORKLOAD	Nein
EVALUATIONORDER	Zweitletzte	Nein
CREATE_TIME	Zeitmarke der Datenbankerstellung	Nein
ALTER_TIME	Zeitmarke der letzten Aktualisierung der Workloaddefinition	Nein (aber der Datenserver ändert diese Spalte bei der Aktualisierung der Workloaddefinition)
ENABLED	Y	Nein
ALLOWACCESS	Y	Ja
SERVICECLASSNAME	SYSDEFAULTSUBCLASS	Ja
PARENTSERVICECLASSNAME	SYSDEFAULTUSERCLASS	Ja
COLLECTAGGACTDATA	N	Nein (für künftige Nutzung reserviert)
COLLECTACTDATA	N	Ja
COLLECTACTPARTITION	C	Ja
EXTERNALNAME	NULL	Nein

Weitere Informationen finden Sie in SYSCAT.WORKLOADS.

Nur UOWs (Unit of Work - Arbeitseinheit), die von einem Sitzungsbenutzer mit der Berechtigung SYSADM oder DBADM übergeben wurden, können der Workload SYSDEFAULTADMWORKLOAD zugeordnet werden. Diese Workload erlaubt es SYSADM- und DBADM-Benutzern, die Datenbank abzufragen und Verwaltungs- oder Überwachungstasks auszuführen, falls die folgenden Ereignisse auftreten:

- Die Workload, der der Administrator zugeordnet ist, darf nicht auf die Datenbank zugreifen (d. h. das Schlüsselwort DISALLOW DB ACCESS der Anweisung CREATE WORKLOAD oder ALTER WORKLOAD wurde für die Workload angegeben).
- Es wurde gegen einen Schwellenwert verstoßen, wodurch verhindert wurde, dass der Administrator an der Datenbank arbeiten konnte.

Die Workload SYSDEFAULTADMWORKLOAD unterscheidet sich folgendermaßen von anderen Workloads:

- Sie kann nicht gelöscht oder inaktiviert werden.

- Sie können für sie nicht DISALLOW DB ACCESS angeben.
- Für Vorkommen dieser Workload und die darin enthaltenen Aktivitäten gilt keiner der Schwellenwerte.
- Sie können diese Workload nur in der Servicesuperklasse SYSDEFAULTUSERCLASS ausführen. Weitere Informationen finden Sie in „Standardservicesuperklassen und -unterklassen“ auf Seite 27.
- Sie können eine Verbindung dieser Workload nur mit dem Befehl SET WORKLOAD zuordnen. Dieser Befehl kann nur von der CLP-Schnittstelle abgesetzt werden. Weitere Informationen finden Sie in „Zuordnen einer Verbindung zur Standardverwaltungsworkload“.

Sie können die Workload SYSDEFAULTADMWORKLOAD durch Abfrage der Tabelle SYSCAT.WORKLOADS anzeigen. Die folgende Tabelle zeigt den Eintrag für die Workload SYSDEFAULTADMWORKLOAD in der Tabelle SYSCAT.WORKLOADS:

Tabelle 11. Eintrag SYSDEFAULTADMWORKLOAD in SYSCAT.WORKLOADS

Spalte	Wert	Modifizierbar mit der Anweisung ALTER WORKLOAD, wenn Sie DBADM- oder SYSADM-Zugriff haben
WORKLOADID	2	Nein
WORKLOADNAME	SYSDEFAULTADMWORKLOAD	Nein
EVALUATIONORDER	Letzte	Nein
CREATE_TIME	Zeitmarke der Datenbankerstellung	Nein
ALTER_TIME	Zeitmarke der letzten Aktualisierung der Workloaddefinition	Nein (aber der Datenserver ändert diese Spalte bei der Aktualisierung der Workloaddefinition)
ENABLED	Y	Nein
ALLOWACCESS	Y	Nein
SERVICECLASSNAME	SYSDEFAULTSUBCLASS	Nein
PARENTSERVICECLASSNAME	SYSDEFAULTUSERCLASS	Nein
COLLECTAGGACTDATA	N	Nein (für künftige Nutzung reserviert)
COLLECTACTDATA	N	Ja
COLLECTACTPARTITION	C	Ja
EXTERNALNAME	NULL	Nein

Weitere Informationen finden Sie in SYSCAT.WORKLOADS.

Zuordnen einer Verbindung zur Standardverwaltungsworkload

Mit dem Befehl SET WORKLOAD können Sie der Standardverwaltungsworkload (SYSDEFAULTADMWORKLOAD) eine Verbindung zuordnen.

Sie benötigen zwar keine Sonderberechtigung für die Verwendung des Befehls SET WORKLOAD, für die Zuordnung einer Verbindung zur Standardverwaltungsworkload ist jedoch die Berechtigung SYSADM oder DBADM erforderlich. Andernfalls wird bei der Workloadzuordnung SQL0552N zurückgegeben.

Die Standardverwaltungsworkload (SYSDEFAULTADMWORKLOAD) ist eine spezielle von DB2 bereitgestellte Workloaddefinition, die keinem DB2-Schwellenwert unterliegt. Diese Workload soll es dem Datenbankadministrator erlauben, seine Arbeit durchzuführen oder nach Bedarf Maßnahmen zur Fehlerbehebung zu ergreifen. Da diese Workload nicht von Schwellenwerten betroffen ist, hat sie eingeschränkte Workload-Management-Steuerung und wird nicht für die Übergabe von alltäglicher Arbeit empfohlen.

Setzen Sie den Befehl SET WORKLOAD wie folgt ab, um der Standardverwaltungsworkload eine Verbindung zuzuordnen:

```
SET WORKLOAD TO SYSDEFAULTADMWORKLOAD
```

Wann der Befehl wirksam wird, hängt vom Zeitpunkt ab, zu dem Sie ihn absetzen:

- Wenn Sie den Befehl SET WORKLOAD TO SYSDEFAULTADMWORKLOAD vor der Verbindung zur Datenbank absetzen, wird diese nach dem Herstellen der Verbindung am Anfang der ersten UOW (Unit of Work - Arbeitseinheit) der Workload SYSDEFAULTADMWORKLOAD zugeordnet.
- Wenn Sie den Befehl SET WORKLOAD TO SYSDEFAULTADMWORKLOAD am Anfang einer UOW absetzen, wird die Verbindung nach dem Herstellen der Verbindung zur Datenbank der Workload SYSDEFAULTADMWORKLOAD zugeordnet, wenn die erste Anforderung übergeben wird, die keine API sqleseti (Clientinformationen festlegen) ist.
- Wenn Sie den Befehl SET WORKLOAD TO SYSDEFAULTADMWORKLOAD in der Mitte einer UOW absetzen, nachdem eine Verbindung hergestellt wurde, wird die Verbindung am Anfang der nächsten UOW der Workload SYSDEFAULTADMWORKLOAD zugeordnet.

Wenn eine Verbindung der Workload SYSDEFAULTADMWORKLOAD zugeordnet ist, erfolgt die Neuordnung der Workload am Anfang der nächsten UOW, wenn eine der folgenden Situationen auftritt:

- Sie entziehen dem Sitzungsbenutzer die Berechtigung SYSADM oder DBADM. In dieser Situation wird SQL0552N zurückgegeben.
- Sie setzen einen Befehl SET WORKLOAD TO AUTOMATIC ab. Dieser Befehl gibt an, dass die nächste UOW nicht der Workload SYSDEFAULTADMWORKLOAD zugeordnet werden soll und dass eine normale Workloadauswertung am Anfang der nächsten UOW ausgeführt werden soll. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.

Arbeiten mit Workloads

Erstellen einer Workload

Mit der Anweisung CREATE WORKLOAD können Sie eine Workload zu den Katalogen hinzufügen.

Zum Erstellen einer Workload benötigen Sie die Berechtigung DBADM oder SYSADM.

Weitere Informationen zu Voraussetzungen finden Sie in den folgenden Themen:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Gehen Sie wie folgt vor, um eine Workload zu erstellen:

1. Geben Sie mit der Anweisung CREATE WORKLOAD mindestens eines der folgenden Merkmale für die Workload an:

- Den Namen der Workload.
- Die Position der Workload relativ zu anderen Workloads bei der Zwischenspeicherung im Hauptspeicher. Die Position der neuen Workload bestimmt die Reihenfolge, in der sie bei der Workloadzuordnung ausgewertet wird. Standardmäßig wird die neue Workload an letzter Stelle positioniert, d. h. dass sie als letztes direkt vor der Standardbenutzerworkload ausgewertet wird. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.
- Den Typ der zu erfassenden Aktivitätsinformationen. Standardmäßig werden keine Informationen zu Aktivitäten, die der Workload zugeordnet sind, an den Aktivitätsereignismonitor gesendet.
- Die Verbindungsattribute. Die eingehende Verbindungsanforderung muss Verbindungsattribute angeben, die den für die Workload definierten Attributen entsprechen, damit eine Übereinstimmung erfolgen kann. Weitere Informationen finden Sie in Kapitel 3, „Workloads“, auf Seite 43. Beachten Sie bei der Angabe der Verbindungsattribute, dass Werte mit OR und Attribute mit AND verknüpft werden: z. B. UserID (bob OR sue OR frank) AND Application (SAS).
- Einen Wert, der angibt, ob Vorkommen dieser Workload auf die Datenbank zugreifen dürfen. Standardmäßig ist der Datenbankszugriff für Vorkommen dieser Workload erlaubt.
- Einen Wert, der angibt, ob die Workload aktiviert oder inaktiviert ist. Standardmäßig ist die Workload aktiviert.
- Die Serviceklasse, unter der Vorkommen dieser Workload ausgeführt werden sollen. Standardeinstellung ist die Servicesuperklasse SYSDEFAULTUSERCLASS.

Wenn Sie eine benutzerdefinierte Servicesuperklasse angeben und die Workload nicht für die Ausführung in einer benutzerdefinierten Serviceunterklasse unter der Servicesuperklasse zuordnen, werden die Workloadvorkommen in der Serviceunterklasse SYSDEFAULTSUBCLASS der Servicesuperklasse ausgeführt.

Anmerkung: Sie können die Serviceunterklasse SYSDEFAULTSUBCLASS unter keiner Servicesuperklasse, auch nicht unter der Servicesuperklasse SYSDEFAULTUSERCLASS, angeben.

Wenn keine Vorkommen der Workload in der Serviceunterklasse SYSDEFAULTSUBCLASS ausgeführt werden sollen, können Sie die Workload zur Ausführung in einer benutzerdefinierten Serviceunterklasse über die Workload zuordnen. Sie können auch eine Arbeitsaktion verwenden, um die Workload einer anderen Serviceunterklasse zuzuordnen. (Weitere Informationen finden Sie in „Arbeitsaktionen und Arbeitsaktionssets“ auf Seite 87.)

2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload zur Sicht SYSCAT.WORKLOADS hinzugefügt. Durch das Festschreiben der Änderung erfolgt die Workloadneuauswertung am Anfang der nächsten UOW (Unit of Work- Arbeitseinheit) jeder Anwendung. Je nachdem, welche Workload ausgewählt wird, wird die Anwendung eventuell einer anderen Workload zugeordnet.

Nachdem Sie eine Workload erstellt haben, müssen Sie unter Umständen mindestens einem Sitzungsbenutzer das Zugriffsrecht USAGE erteilen. (Sitzungsbenutzer mit der Berechtigung SYSADM oder DBADM haben ein implizites Zugriffsrecht

für jede Workload.) Selbst wenn eine Verbindung eine exakte Übereinstimmung mit den Verbindungsattributen der Workload aufweist, berücksichtigt der Daten-server die Workload nicht bei der Workloadauswertung, wenn der Sitzungsbenutzer nicht das Zugriffsrecht USAGE für die Workload hat. Weitere Informationen finden Sie in „Erteilen des Zugriffsrechts USAGE für eine Workload“ auf Seite 57.

Ändern einer Workload

Eine Anweisung ALTER WORKLOAD ändert eine Workload in den Katalogen.

Zum Ändern einer Workload benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Gehen Sie wie folgt vor, um eine Workload zu ändern:

1. Geben Sie mit der Anweisung ALTER WORKLOAD mindestens eines der folgenden Merkmale für die Workload an:
 - Die Verbindungsattribute. Sie können Verbindungsattribute zur Workloaddefinition hinzufügen oder daraus löschen, sofern es sich nicht um die Workload SYSDEFAULTUSERWORKLOAD oder SYSDEFAULTADMWORKLOAD handelt. Die eingehende Verbindungsanforderung muss Verbindungsattribute angeben, die den für die Workload definierten Attributen entsprechen, damit eine Übereinstimmung erfolgen kann. Weitere Informationen finden Sie in Kapitel 3, „Workloads“, auf Seite 43. Zum Anzeigen der Verbindungsattribute für eine Workload fragen Sie die Sicht SYSCAT.WORKLOADCONNATTR ab.
 - Einen Wert, der angibt, ob ein Vorkommen dieser Workload auf die Datenbank zugreifen darf. Standardmäßig ist der Datenbankzugriff für ein Vorkommen dieser Workload erlaubt. Der Datenbankzugriff für die Workload SYSDEFAULTADMWORKLOAD kann nicht verweigert werden.
 - Einen Wert, der angibt, ob die Workload aktiviert oder inaktiviert ist. Standardmäßig ist die Workload aktiviert. Sie können die Workload SYSDEFAULTUSERWORKLOAD oder SYSDEFAULTADMWORKLOAD nicht inaktivieren.
 - Die Serviceklasse, unter der Vorkommen dieser Workload ausgeführt werden sollen. Standardeinstellung ist die Servicesuperklasse SYSDEFAULTUSERCLASS. Wenn Sie eine benutzerdefinierte Servicesuperklasse angeben, können Sie unter der Servicesuperklasse eine Serviceunterklasse angeben. Die Unterklasse SYSDEFAULTSUBCLASS kann unter keiner Servicesuperklasse, auch nicht unter der Servicesuperklasse SYSDEFAULTUSERCLASS, angegeben werden. Zudem können Sie nicht die Servicesuperklasse SYSDEFAULTSYSTEMCLASS oder SYSDEFAULTMAINTENANCECLASS angeben.
 - Die Position der Workload relativ zu anderen Workloads, womit die Reihenfolge festgelegt wird, in der die Workload bei der Workloadzuordnung ausgewertet wird. Standardmäßig wird eine neue Workload an letzter Stelle positioniert, d. h. dass sie als letztes direkt vor der Standardbenutzerworkload ausgewertet wird. Die Position der Workload SYSDEFAULTUSERWORKLOAD oder SYSDEFAULTADMWORKLOAD kann nicht festgelegt werden. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.

- Den Typ der zu erfassenden Aktivitätsinformationen. Standardmäßig werden keine Informationen zu Aktivitäten, die der Workload zugeordnet sind, an den Aktivitätsereignismonitor gesendet.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht SYSCAT.WORKLOADS aktualisiert. Durch das Festschreiben der Änderung erfolgt die Workloadneuauswertung am Anfang der nächsten UOW (Unit of Work- Arbeitseinheit) jeder Anwendung. Je nachdem, welche Workload ausgewählt wird, wird die Anwendung eventuell einer anderen Workload zugeordnet.

Nachdem Sie eine Workload geändert haben, müssen Sie unter Umständen mindestens einem Sitzungsbenutzer das Zugriffsrecht USAGE erteilen. (Sitzungsbenutzer mit der Berechtigung SYSADM oder DBADM haben ein implizites Zugriffsrecht für jede Workload.) Selbst wenn eine Verbindung eine exakte Übereinstimmung mit den Verbindungsattributen der Workload bietet, ordnet der Datenserver die Verbindung der Workload nicht zu und erzeugt kein Vorkommen der Workload, wenn der Sitzungsbenutzer nicht das Zugriffsrecht USAGE für die Workload hat. Weitere Informationen finden Sie in „Erteilen des Zugriffsrechts USAGE für eine Workload“ auf Seite 57.

Erlauben des Datenbankzugriffs für Vorkommen einer Workload

Wenn Sie eine Workload haben, die nicht auf die Datenbank zugreifen darf, Sie jetzt aber die Ausführung von Vorkommen dieser Workload erlauben wollen, ändern Sie die Workload so, dass sie auf die Datenbank zugreifen darf. Nach dem Erstellen einer Workload ist diese standardmäßig berechtigt, auf die Datenbank zuzugreifen.

Sie benötigen die Berechtigung DBADM oder SYSADM, um eine Workload so zu ändern, dass sie auf eine Datenbank zugreifen darf.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Auch wenn Sie den Zugriff einer Workload auf die Datenbank verhindern, untersucht der Datenserver diese Workload bei der Workloadzuordnung. Alle Vorkommen dieser Workload werden jedoch mit einem Fehler zurückgewiesen. Gehen Sie wie folgt vor, um einer Workload den Zugriff auf die Datenbank zu erlauben:

1. Verwenden Sie die Option ALLOW DB ACCESS der Anweisung ALTER WORKLOAD, um der Workload den Zugriff auf die Datenbank zu erlauben. Wenn Sie z. B. einer Workload namens WL1 den Datenbankzugriff erlauben wollen, geben Sie die folgende Anweisung an:
ALTER WORKLOAD WL1 ALLOW DB ACCESS
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht SYSCAT.WORKLOADS aktualisiert.

Die Änderung einer Workload, mit der der Datenbankzugriff für ihre Vorkommen erlaubt wird, wird wirksam, wenn der Datenserver die nächste UOW (Unit of Work - Arbeitseinheit) für diese Workload analysiert. Wenn Sie z. B. DISALLOW DB ACCESS für Workload A angegeben haben und Sie nun die Workload mit ALLOW DB ACCESS ändern, dürfen neue Vorkommen von Workload A ausgeführt werden. Bisher wären alle Vorkommen von Workload A mit einem Fehler zurückgewiesen worden.

Verhindern des Datenbankzugriffs für Vorkommen einer Workload

Mit dieser Task können Sie steuern, welche Workloads auf die Datenbank zugreifen dürfen. Vor der Ausführung eines Workloadvorkommens prüft der Daten-server, ob die Workload berechtigt ist, auf die Datenbank zuzugreifen. Wenn Sie den Datenbankzugriff für das Workloadvorkommen verhindern, wird ein Fehler zurückgegeben, der meldet, dass das Workloadvorkommen zurückgewiesen wird.

Zum Verhindern des Datenbankzugriffs für eine Workload benötigen Sie die Berechtigung DBADM oder SYSADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Das Verhindern des Datenbankzugriffs für ein Workloadvorkommen unterscheidet sich von der Inaktivierung einer Workload. Wenn Sie eine Workload inaktivieren, wird die Workloaddefinition nicht im Hauptspeicher zwischengespeichert und wird daher für die Workloadzuordnung nicht berücksichtigt. Gehen Sie wie folgt vor, um den Datenbankzugriff für eine Workload zu verhindern:

1. Verwenden Sie die Option `DISALLOW DB ACCESS` der Anweisung `ALTER WORKLOAD` (siehe folgendes Beispiel):

```
ALTER WORKLOAD workloadname DISALLOW DB ACCESS ...
```
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht `SYSCAT.WORKLOADS` aktualisiert.

Die Änderung einer Workload, mit der der Datenbankzugriff für ihre Vorkommen verhindert wird, wird bei bereits ausgeführten Workloadvorkommen am Anfang der nächsten UOW (Unit of Work - Arbeitseinheit) wirksam. Wenn Sie z. B. `ALLOW DB ACCESS` für die Workload A angeben und die Workload durch Angabe von `DISALLOW DB ACCESS` ändern, empfangen Vorkommen der Workload A, die bereits ausgeführt werden, am Anfang der nächsten UOW einen SQL-Fehler. Neue Vorkommen von Workload A werden zurückgewiesen.

Aktivieren einer Workload

Der DB2-Datenserver vergleicht die angegebenen Verbindungsattribute für einen Workloadagenten mit den Verbindungsattributen der aktuellen Sitzung. Bei der Suche nach übereinstimmenden Workloads berücksichtigt der Datenserver keine inaktivierten Workloads.

Zum Ändern einer Workload benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Standardmäßig ist eine Workload bei der Erstellung aktiviert. Wenn Sie eine Workload als inaktiviert erstellen, müssen Sie sie aktivieren, damit der Daten-server sie bei der Workloadauswertung berücksichtigt.

Gehen Sie wie folgt vor, um eine Workload zu aktivieren:

1. Geben Sie die Workload an, die Sie aktivieren wollen. Sie können die Gruppe der inaktivierten Workloads durch Abfrage der Sicht `SYSCAT.WORKLOADS` anzeigen (siehe folgendes Beispiel):

```
SELECT * FROM SYSCAT.WORKLOADS WHERE ENABLED='N'
```

2. Aktivieren Sie die inaktivierte Workload mit der Anweisung ALTER WORKLOAD:

```
ALTER WORKLOAD...ENABLE
```

Wenn die Anweisung ALTER WORKLOAD erfolgreich ist, wird die Definition für die Workload in den Datenbankkatalog geschrieben.

3. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht SYSCAT.WORKLOADS aktualisiert.

Die Aktivierung der Workload wird am Anfang der nächsten UOW (Unit of Work - Arbeitseinheit) wirksam. An diesem Punkt erfolgt die Workloadneuauswertung, und der Datenserver berücksichtigt die neu aktivierte Workload bei der Workloadneuauswertung.

Inaktivieren einer Workload

Mit dieser Task können Sie verhindern, dass bestimmte Workloads bei der Workloadzuordnung berücksichtigt werden. Der DB2-Datenserver vergleicht die Verbindungsattribute, die Sie für eine Workload angeben, mit den Verbindungsattributen der aktuellen Sitzung. Wenn Sie eine Workload inaktivieren, berücksichtigt der Datenserver diese bei der Suche nach einer übereinstimmenden Workload nicht. Statt dessen ordnet der Datenserver die UOW (Unit of Work - Arbeitseinheit) der nächsten übereinstimmenden Workload zu. Wenn keine benutzerdefinierte Workload übereinstimmt, wird die UOW der Standardworkload zugeordnet.

Zum Erstellen oder Ändern einer Workload benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Gehen Sie wie folgt vor, um eine Workload zu inaktivieren:

1. Verwenden Sie die Option DISABLE der Anweisung ALTER WORKLOAD, um die Workload zu inaktivieren:

```
ALTER WORKLOAD...DISABLE
```
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht SYSCAT.WORKLOADS aktualisiert.

Die Inaktivierung der Workload wird am Anfang der nächsten UOW wirksam. An diesem Punkt erfolgt die Workloadneuauswertung, und die Verbindung wird der nächsten Workload zugeordnet, die mit den Verbindungsattributen übereinstimmt und für die eine Berechtigung vorhanden ist.

Erteilen des Zugriffsrechts USAGE für eine Workload

Damit einer Workload eine Verbindung zugeordnet wird, muss der Sitzungsbenuer das Zugriffsrecht USAGE für diese Workload haben. Benutzer mit den Berechtigungen SYSADM und DBADM haben implizit das Zugriffsrecht USAGE für alle Workloads.

Sie benötigen die Berechtigung SYSADM oder DBADM, um die Anweisung GRANT USAGE ON WORKLOAD verwenden zu können.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Wenn der Datenserver eine Workload findet, die den Attributen einer eingehenden Verbindung entspricht, prüft der Datenserver, ob der Sitzungsbenutzer das Zugriffsrecht USAGE für diese Workload hat. Hat der Sitzungsbenutzer das Zugriffsrecht USAGE für diese Workload nicht, sucht der Datenserver nach der nächsten übereinstimmenden Workload. (Die Workloads, für die der Sitzungsbenutzer das Zugriffsrecht USAGE nicht hat, werden behandelt, als wären sie nicht vorhanden.) Über das Zugriffsrecht USAGE für eine Workload können Sie daher weiter steuern, welcher Workload unter den übereinstimmenden Workloads ein Benutzer, eine Gruppe oder eine Rolle zugeordnet werden sollte. Sie können z. B. mehrere Workloads mit denselben Verbindungsattributen definieren und das Zugriffsrecht USAGE für jede dieser Workloads nur bestimmten Benutzern, Gruppen oder Rollen erteilen. Weitere Informationen finden Sie in „Workloadzuordnung“ auf Seite 48.

Der Client kann die Clientbenutzer-ID, den Clientanwendungsnamen, den Clientworkstationnamen und die Clientabrechnungszeichenfolge (die einige der Verbindungsattribute darstellen, die für die Zuordnung einer Verbindung zu einer Workload verwendet werden) ohne Berechtigung festlegen. Daher können Sie über das Zugriffsrecht USAGE für Workloads auch steuern, welcher Sitzungsbenutzer die Berechtigung zur Verwendung einer Workload hat.

Sie können die Informationen zum Zugriffsrecht USAGE durch Abfrage der Sicht SYSCAT.WORKLOADAUTH anzeigen.

Wenn Sie eine Datenbank ohne die Option RESTRICT erstellen, wird das Zugriffsrecht USAGE für die Workload SYSDEFAULTUSERWORKLOAD bei der Datenbankeinstellung dem Benutzer PUBLIC erteilt. Andernfalls müssen Sie das Zugriffsrecht USAGE Benutzern ohne die Berechtigung SYSADM oder DBADM explizit erteilen. Wenn der Sitzungsbenutzer für keine der Workloads, einschließlich SYSDEFAULTUSERWORKLOAD, das Zugriffsrecht USAGE hat, wird SQL4707N zurückgegeben, wenn der Datenserver versucht, eine Workload mit der Datenbankverbindung zu verknüpfen.

Gehen Sie wie folgt vor, um das Zugriffsrecht USAGE für eine Workload zu erteilen:

1. Verwenden Sie die Anweisung GRANT USAGE ON WORKLOAD. Sie können das Zugriffsrecht USAGE bestimmten Benutzern, Gruppen, Rollen oder dem Benutzer PUBLIC erteilen. Wenn Sie z. B. das Zugriffsrecht USAGE für die Workload ACCOUNTS der Gruppe CPA erteilen wollen, müssen Sie die folgende Anweisung absetzen:

```
GRANT USAGE ON WORKLOAD ACCOUNTS TO GROUP CPA
```

Für die Workload SYSDEFAULTADMWORKLOAD können Sie das Zugriffsrecht USAGE nicht erteilen. Die Workload SYSDEFAULTADMWORKLOAD kann nur von SYSADM- und DBADM-Benutzern verwendet werden, die den Befehl SET WORKLOAD TO SYSDEFAULTADMWORKLOAD absetzen.

2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Sicht SYSCAT.WORKLOADAUTH aktualisiert. Bis zur Festschreibung der Anweisung GRANT berücksichtigt der Datenserver die Workload nicht für die Zuordnung für die neu autorisierten Benutzer, Gruppen oder Rollen.

Widerrufen des Zugriffsrechts USAGE für eine Workload

Mit der Anweisung `REVOKE USAGE ON WORKLOAD` können Sie das Zugriffsrecht `USAGE` für eine Workload widerrufen.

Sie benötigen die Berechtigung `SYSADM` oder `DBADM`, um die Anweisung `REVOKE USAGE ON WORKLOAD` verwenden zu können.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Für die Workload `SYSDEFAULTADMWORKLOAD` können Sie das Zugriffsrecht `USAGE` nicht explizit widerrufen. Nur `SYSADM`- und `DBADM`-Benutzer, die den Befehl `SET WORKLOAD TO SYSDEFAULTADMWORKLOAD` absetzen, können diese Workload verwenden. Daher funktioniert die Anweisung `REVOKE USAGE ON WORKLOAD` nicht für `SYSDEFAULTADMWORKLOAD`.

Gehen Sie wie folgt vor, um das Zugriffsrecht `USAGE` für eine Workload zu widerrufen:

1. Verwenden Sie die Anweisung `REVOKE USAGE ON WORKLOAD`. Sie können das Zugriffsrecht `USAGE` für bestimmte Benutzer, Gruppen, Rollen oder den Benutzer `PUBLIC` widerrufen. Wenn Sie z. B. das Zugriffsrecht `USAGE` für die Workload `ACCOUNTS` des Benutzers `PUBLIC` widerrufen wollen, müssen Sie die folgende Anweisung absetzen:
`REVOKE USAGE ON WORKLOAD ACCOUNTS FROM PUBLIC`
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Sicht `SYSCAT.WORKLOADAUTH` aktualisiert. Bis zum Durchführen eines Commits für die Anweisung `REVOKE` berücksichtigt der Datenserver die Workload bei der Workloadzuordnung.

Löschen einer Workload

Durch das Löschen einer Workload wird diese aus dem Datenbankkatalog entfernt.

Zum Löschen einer Workload benötigen Sie die Berechtigung `DBADM` oder `SYSADM`.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Gehen Sie wie folgt vor, um eine Workload zu löschen:

1. Inaktivieren Sie die Workload mit der Anweisung `ALTER WORKLOAD`. Weitere Informationen finden Sie in „Inaktivieren einer Workload“ auf Seite 57. Durch die Inaktivierung der Workload wird verhindert, dass neue Vorkommen der Workload in der Datenbank ausgeführt werden können.
2. Stellen Sie mit Hilfe der Tabellenfunktion `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES` sicher, dass momentan keine Vorkommen dieser Workload ausgeführt werden. Weitere Informationen finden Sie in Tabellenfunktion `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES`. Die Tabellenfunktion `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES` gibt die Anwendungskennungen zurück, die den aktiven Workloadvorkommen entsprechen. Mit dem Befehl `FORCE APPLICATION` können Sie die Anwendungen anhand der Anwendungskennungen beenden.

3. Löschen Sie die Workload mit der Anweisung `DROP WORKLOAD`. Wenn Sie z. B. die Workload `ACCTNG` löschen wollen, geben Sie die folgende Anweisung an:
`DROP WORKLOAD ACCTNG`
4. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload aus der Sicht `SYSCAT.WORKLOADS` entfernt. Außerdem werden Berechtigungsinformationen für die Workload aus der Sicht `SYSCAT.WORKLOADAUTH` entfernt.

Kapitel 4. Schwellenwerte

Sie können Schwellenwerte verwenden, um unsachgemäße Ressourcennutzung oder eine beginnende Systemüberlastung zu erkennen.

Mit Schwellenwerten können Sie explizit Grenzwerte für die Nutzung einer bestimmten Ressource festlegen. Wird gegen einen Schwellenwert verstoßen, kann eine definierte Aktion ausgelöst werden. Die folgenden Aktionen werden unterstützt:

- Die Aktivität, die den Schwellenwertverstoß verursachte, wird nicht weiter verarbeitet (STOP EXECUTION).
- Die Verarbeitung wird fortgesetzt (CONTINUE).
- Es werden Informationen zur Aktivität erfasst, die den Schwellenwertverstoß verursachte.

Unabhängig davon, ob eine Aktivität, die einen Schwellenwertverstoß verursachte, gestoppt oder fortgesetzt wird, können Sie detaillierte Informationen zur Aktivität erfassen. Informationen zur Aktivität, die den Schwellenwertverstoß verursachte, werden vom aktiven ACTIVITIES-Ereignismonitor erfasst, wenn die Ausführung der Aktivität abgeschlossen ist.

Unabhängig davon, ob Sie Informationen zur Aktivität erfassen, die den Schwellenwertverstoß verursachte, wird ein Schwellenwertverstoßeintrag in den aktiven THRESHOLD VIOLATIONS-Ereignismonitor geschrieben, wenn gegen den Schwellenwert verstoßen wird.

Schwellenwerte werden in zwei Hauptkategorien klassifiziert:

- **Aktivitätsschwellenwerte.** Ein Aktivitätsschwellenwert bezieht sich auf eine einzelne Aktivität. Der Schwellenwert für die maximale Gesamtaktivitätszeit z. B. ist ein Aktivitätsschwellenwert, da er die Gesamtdauer begrenzt, während der eine einzelne Aktivität im Datenserver verbleiben kann.
- **Zusammengefasste Schwellenwerte.** Ein zusammengefasster Schwellenwert setzt einen Grenzwert für einen Messwert, der über eine Gruppe von mehreren Aktivitäten berechnet wird. Die maximale Anzahl gleichzeitig ablaufender Aktivitäten in einer Serviceklasse ist z. B. ein zusammengefasster Schwellenwert.

Jeder Schwellenwert bezieht sich auf eine *Domäne*. Die Domäne eines Schwellenwerts definiert das Datenbankobjekt, auf die der Schwellenwert angewendet wird. Nur Aktivitäten, die in der Domäne eines Schwellenwerts stattfinden, sind davon betroffen. Es gibt die folgenden Schwellenwertdomänen:

- Datenbank
- Servicesuperklasse
- Serviceunterklasse
- Arbeitsaktion
- Workload

In jeder dieser Schwellenwertdomänen kann der Schwellenwert über ein einzelnes Workloadvorkommen, eine Datenbankpartition oder alle Partitionen einer Datenbank erzwungen werden. Dies wird als *Umsetzungsbereich* des Schwellenwerts bezeichnet.

Zusammengefasste Serviceklassenschwellenwerte haben einen von zwei Umsetzungsbereichen: Datenbank und Datenbankpartition. Ein Beispiel eines zusammengefassten Schwellenwerts, der auf Datenbankpartitionsebene angewendet wird, ist die maximale Anzahl gleichzeitig bestehender Verbindungen für eine Servicesuperklasse auf einer Partition. Ein Beispiel eines zusammengefassten Schwellenwerts, der auf Datenbankebene (d. h. über alle Datenbankpartitionen) angewendet wird, ist die maximale Anzahl gleichzeitig ablaufender Aktivitäten für eine Serviceklasse in allen Partitionen.

Einige Schwellenwerte verfügen über eine integrierte Warteschlange und werden mit zwei Begrenzungen definiert: einer *Schwellenwertbegrenzung* und einer *Warteschlangenbegrenzung*. Diese Schwellenwerte werden als *Warteschlangenschwellenwerte* bezeichnet. Die Schwellenwertbegrenzung eines Warteschlangenschwellenwerts erzwingt typischerweise eine Stufe des gemeinsamen Zugriffs (z. B. die maximale Anzahl gleichzeitig ablaufender Aktivitäten), über die hinaus weitere Anforderungen in die Warteschlange gestellt werden. Die Warteschlangenbegrenzung definiert die Obergrenze für die Warteschlange. Insbesondere wenn eine Aktivität gegen die Schwellenwertbegrenzung eines Warteschlangenschwellenwerts verstößt, werden neue Arbeitsanforderungen, die von diesem Schwellenwert überwacht werden, automatisch nach dem First-in/First-out-Prinzip in die Warteschlange gestellt, bis die Warteschlange die von der Warteschlangenbegrenzung definierte Größe erreicht. Wenn die Obergrenze erreicht ist, ist die Warteschlange voll, und die für den Schwellenwert angegebene Aktion wird auf neu eintreffende Arbeit, die von diesem Schwellenwert überwacht wird, angewendet. Bei einer Aktion STOP EXECUTION wird neu eintreffende Arbeit z. B. zurückgewiesen.

Es ist auch möglich, die Obergrenze als *unbegrenzt* zu definieren. In diesem Fall gibt es keine Obergrenze für die Größe der Warteschlange. In dieser Situation wird neu eintreffende Arbeit immer der Warteschlange hinzugefügt. Wenn Sie als Obergrenze einen festen Grenzwert und als Schwellenwertaktion CONTINUE definieren, wird die gesamte neu eintreffende Arbeit, die gegen die Schwellenwertbegrenzung verstößt, der Warteschlange hinzugefügt, und der Schwellenwert verhält sich, als wäre die zugehörige Warteschlangenbegrenzung unbegrenzt.

Unterschiedliche Schwellenwerte verfolgen unterschiedliche Arten von Arbeit. Ein Schwellenwert kann beispielsweise SQL-basierte Aktivitäten, Dienstprogramme wie das Dienstprogramm LOAD, Verbindungen, Workloadvorkommen usw. überwachen. Arbeit, die für einen Schwellenwert von Interesse ist, wird als *überwachte Arbeit* für diesen Schwellenwert bezeichnet. So bezieht sich z. B. ein Schwellenwert, der auf einer Reihe von Timerons basiert, nur auf Arbeit, der ein Timeronwert zugeordnet ist, (in dieser Situation DML-basierte Aktivitäten) und umfasst keine anderen Arbeitstypen wie DDL oder Dienstprogramme.

Ein Schwellenwert kann entweder prädiktiv oder reaktiv sein:

- Die Begrenzungen eines prädiktiven Schwellenwerts werden vor der Ausführung der überwachten Arbeit geprüft. Der Datenserver bezieht Nutzungsschätzungen vom SQL-Compiler, um zu prüfen, ob ein prädiktiver Schwellenwert verletzt würde.
- Die Begrenzungen eines reaktiven Schwellenwerts werden geprüft, während eine verfolgte Arbeit ausgeführt wird. Es werden näherungsweise berechnete Nutzungsschätzungen der Laufzeit der gesteuerten Ressource verwendet, um die Begrenzungen von reaktiven Schwellenwerten auszuwerten. Die Laufzeitnutzungsschätzungen werden nicht kontinuierlich bezogen, sondern an ausgewählten vordefinierten Prüfpunkten in der Laufzeit der verfolgten Arbeit.

Schwellenwerte werden nicht auf alle Anweisungen angewendet. Sie gelten z. B. nicht für die Anweisungen COMMIT, ROLLBACK, SAVEPOINT und ROLLBACK to SAVEPOINT.

Mit der Anweisung CREATE THRESHOLD können Sie Schwellenwerte erstellen. Mit der Anweisung ALTER THRESHOLD können Sie Schwellenwerte ändern. Mit der Anweisung DROP THRESHOLD können Sie Schwellenwerte löschen.

Sie können Ihre Schwellenwerte durch Abfrage der Sicht SYSCAT.THRESHOLDS anzeigen.

Aktivitätsschwellenwerte und zusammengefasste Schwellenwerte

Es werden zwei Arten von Workload-Management-Schwellenwerten unterstützt: Aktivitätsschwellenwerte und zusammengefasste Schwellenwerte.

Ein Aktivitätsschwellenwert bezieht sich auf eine einzelne Aktivität. Wenn die Ressourcennutzung einer einzelnen Aktivität gegen die Obergrenze des Schwellenwerts verstößt, der diese Aktivität verfolgt, wird die entsprechende Aktion ausgelöst und einmal auf die Aktivität angewendet. Nachdem der Aktivitätsschwellenwert angewendet wurde, wird er für diese Aktivität inaktiviert. Nehmen wir z. B. an, dass Sie einen Zeitschwellenwert von 5 Minuten definieren und dass als Aktion für diesen Schwellenwert CONTINUE (Fortsetzen) festgelegt ist. Wenn eine Aktivität gegen diesen Schwellenwert verstößt, wird der Schwellenwert nur einmal und nicht alle 5 Minuten angewendet.

Ein zusammengefasster Schwellenwert steuert Arbeitselemente in einer Datenbank gemeinsam. Die mit einem zusammengefassten Schwellenwert definierte Begrenzung fungiert als laufende Summe, zu der die gesamte vom Schwellenwert überwachte Arbeit beiträgt. Wenn neu instanziierte Arbeit einen Verstoß gegen die obere Begrenzung verursacht, wird die entsprechende Aktion ausgelöst. Nur die Arbeit, die den Begrenzungsverstoß verursachte, ist von der ausgelösten Aktion betroffen.

Schwellenwertzusammenfassung

Die Tabelle in diesem Abschnitt bietet eine kurze Zusammenfassung der verfügbaren Schwellenwerte mit ihren entsprechenden Definitionsdomänen und Umsetzungsbereichen.

Tabelle 12. Schwellenwerte mit Definitionsdomänen und Umsetzungsbereichen

	Schwellenwerte mit einer Datenbank als Umsetzungsbereich	Schwellenwerte mit einer Datenbankpartition als Umsetzungsbereich	Schwellenwerte mit einem Workloadvorkommen als Umsetzungsbereich
Schwellenwerte mit einer Datenbank als Schwellenwertdomäne	<ul style="list-style-type: none"> „Schwellenwert CONCURRENTDBCOORDACTIVITIES“ auf Seite 73 „Schwellenwert ESTIMATEDSQLCOST“ auf Seite 64 „Schwellenwert SQLROWSRETURNED“ auf Seite 66 „Schwellenwert ACTIVITYTOTALTIME“ auf Seite 67 „Schwellenwert CONNECTIONIDLETIME“ auf Seite 64 	<ul style="list-style-type: none"> „Schwellenwert TOTALDBPARTITIONCONNECTIONS“ auf Seite 68 „Schwellenwert SQLTEMPSPACE“ auf Seite 65 	Nicht zutreffend
Schwellenwerte mit einem Arbeitsaktionsset als Schwellenwertdomäne	<ul style="list-style-type: none"> „Schwellenwert CONCURRENTDBCOORDACTIVITIES“ auf Seite 73 „Schwellenwert ESTIMATEDSQLCOST“ auf Seite 64 „Schwellenwert SQLROWSRETURNED“ auf Seite 66 „Schwellenwert ACTIVITYTOTALTIME“ auf Seite 67 	<ul style="list-style-type: none"> „Schwellenwert SQLTEMPSPACE“ auf Seite 65 	Nicht zutreffend

Tabelle 12. Schwellenwerte mit Definitionsdomänen und Umsetzungsbereichen (Forts.)

	Schwellenwerte mit einer Datenbank als Umsetzungsbereich	Schwellenwerte mit einer Datenbankpartition als Umsetzungsbereich	Schwellenwerte mit einem Workloadvorkommen als Umsetzungsbereich
Schwellenwerte mit einer Servicesuperklasse als Schwellenwertdomäne	<ul style="list-style-type: none"> „Schwellenwert CONCURRENTDBCOORDACTIVITIES“ auf Seite 73 „Schwellenwert ESTIMATEDSQLCOST“ „Schwellenwert SQLROWSRETURNED“ auf Seite 66 „Schwellenwert ACTIVITYTOTALTIME“ auf Seite 67 „Schwellenwert CONNECTIONIDLETIME“ 	<ul style="list-style-type: none"> „Schwellenwert TOTALSPARTITIONCONNECTIONS“ auf Seite 69 „Schwellenwert SQLTEMPSPACE“ auf Seite 65 	Nicht zutreffend
Schwellenwerte mit einer Serviceunterklasse als Schwellenwertdomäne	<ul style="list-style-type: none"> „Schwellenwert CONCURRENTDBCOORDACTIVITIES“ auf Seite 73 „Schwellenwert ESTIMATEDSQLCOST“ „Schwellenwert SQLROWSRETURNED“ auf Seite 66 „Schwellenwert ACTIVITYTOTALTIME“ auf Seite 67 	<ul style="list-style-type: none"> „Schwellenwert SQLTEMPSPACE“ auf Seite 65 	Nicht zutreffend
Schwellenwerte mit einer Workloaddefinition als Schwellenwertdomäne	Nicht zutreffend	<ul style="list-style-type: none"> „Schwellenwert CONCURRENTWORKLOADOCCURRENCES“ auf Seite 70 	<ul style="list-style-type: none"> „Schwellenwert CONCURRENTWORKLOADACTIVITIES“ auf Seite 71

Aktivitätsschwellenwerte

Schwellenwert CONNECTIONIDLETIME

Der Schwellenwert CONNECTIONIDLETIME gibt die Höchstdauer an, während der eine Verbindung inaktiv sein kann (d. h. nicht an einer Benutzeranforderung arbeitet).

Typ Aktivität

Definitionsdomäne

Datenbank oder Servicesuperklasse

Umsetzungsbereich

Datenbank

Überwachte Arbeit

Benutzerverbindungen

Warteschlangensteuerung

Nein

Einheit

Dauer in Minuten, Stunden oder Tagen

Prädiktiv oder reaktiv

Reaktiv

Wenn eine Verbindung länger als während der vom Schwellenwert definierten Dauer inaktiv bleibt und die Schwellenwertaktion STOP EXECUTION ist, wird die Verbindung geschlossen.

Dieser Schwellenwert hat eine Unterteilung von 5 Minuten, so dass jeder Wert, den Sie für diesen Schwellenwert angeben, auf das nächste Mehrfache von 5 Minuten ungleich Null gerundet wird.

Schwellenwert ESTIMATEDSQLCOST

Der Schwellenwert ESTIMATEDSQLCOST gibt den maximalen geschätzten Aufwand an, der für DML-Aktivitäten zulässig ist.

Typ Aktivität

Definitionsdomäne

Datenbank, Servicesuperklasse, Serviceunterklasse, Arbeitsaktion

Umsetzungsbereich

Datenbank

Überwachte Arbeit

Details finden Sie in den später in diesem Thema enthaltenen Informationen.

Warteschlangensteuerung

Nein

Einheit

Geschätzter SQL-Aufwand, ausgedrückt in Timerons

Prädiktiv oder reaktiv

Prädiktiv

Von diesem Schwellenwert werden die folgenden Aktivitäten verfolgt:

- DML-Aktivitäten, die auf der Koordinatorpartition abgesetzt werden.
- Verschachtelte DML-Aktivitäten, die über eine Benutzeranwendung aufgerufen werden. Daher sind DML-Aktivitäten, die vom Datenserver intern abgesetzt werden, (z. B. DML-Code, der aus den DB2-Dienstprogrammen, gespeicherten SYSPROC-Prozeduren, internem SQL-Code usw. abgesetzt wird) von diesem Schwellenwert nicht betroffen, sofern ihr Aufwand nicht in der Schätzung für die übergeordnete Aktivität enthalten ist. In dieser Situation werden diese Aktivitäten indirekt überwacht. Ein Trigger ist ein Beispiel für eine indirekt überwachte Aktivität.

Informationen zu den Aktivitäten, die in die Arbeitsklasse mit dem Arbeitstyp DML fallen, finden Sie in „Arbeitstypen und SQL-Anweisungen von Arbeitsklassen“ auf Seite 91.

Der Datenserver fasst IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors als Benutzerlogik auf. Für Aktivitäten, die über IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors aufgerufen werden, gelten Schwellenwerte.

Schwellenwert SQLTEMPSPACE

Der Schwellenwert SQLTEMPSPACE gibt den Maximalwert für den Tabellenbereich für temporäre Tabellen an, der von einer DML-Aktivität auf jeder beliebigen Datenbankpartition verbraucht werden kann. DML-Aktivitäten nutzen oft einen Tabellenbereich für temporäre Tabellen für Operationen wie die Sortierung und die Bearbeitung von Zwischenergebnismengen.

Typ Aktivität

Definitionsdomäne

Datenbank, Servicesuperklasse, Serviceunterklasse, Arbeitsaktion

Umsetzungsbereich

Datenbankpartition

Überwachte Arbeit

Details finden Sie in den später in diesem Thema enthaltenen Informationen.

Warteschlangensteuerung

Nein

Einheit

Größe des Tabellenbereichs für temporäre Tabellen in Kilobyte (KB), Megabyte (MB) oder Gigabyte (GB)

Prädiktiv oder reaktiv

Reaktiv

Von diesem Schwellenwert werden die folgenden Aktivitäten verfolgt:

- DML-Aktivitäten, die auf der Koordinatorpartition abgesetzt werden.
- Verschachtelte DML-Aktivitäten, die von Benutzeranwendungen abgeleitet werden. Daher sind DML-Aktivitäten, die von DB2-Logik (Dienstprogrammen, SYS-PROC-Prozeduren oder internem SQL-Code) abgesetzt werden, von diesem Schwellenwert nicht betroffen.

Der Datenserver fasst IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors als Benutzerlogik auf. Für Aktivitäten, die über IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors aufgerufen werden, gelten Schwellenwerte.

Schwellenwert SQLROWSRETURNED

Der Schwellenwert SQLROWSRETURNED gibt die maximale Anzahl von Zeilen an, die vom Datenserver an den Client zurückgegeben werden können.

Typ Aktivität

Definitionsdomäne

Datenbank, Servicesuperklasse, Serviceunterklasse, Arbeitsaktion

Umsetzungsbereich

Datenbank

Überwachte Arbeit

Details finden Sie in den später in diesem Thema enthaltenen Informationen.

Warteschlangensteuerung

Nein

Einheit

Anzahl der Zeilen

Prädiktiv oder reaktiv

Reaktiv

Wenn mehrere Ergebnismengen von einer Anweisung CALL zurückgegeben werden, gilt der Schwellenwert separat für jede Ergebnismenge und nicht als zusammengefasster Schwellenwert für die Gesamtzahl der Zeilen, die in allen Ergebnismengen zurückgegeben werden. Wenn Sie z. B. einen Schwellenwert von 20 Zeilen definieren und die Anweisung CALL zwei Ergebnismengen mit 15 und mit 19 Zeilen zurückgibt, tritt kein Schwellenwertverstoß auf.

Von diesem Schwellenwert werden die folgenden Aktivitäten verfolgt:

- DML-Aktivitäten, die auf der Koordinatorpartition abgesetzt werden.
- Verschachtelte DML-Aktivitäten, die über eine Benutzeranwendung aufgerufen werden. Daher sind DML-Aktivitäten, die vom Datenserver intern abgesetzt werden, (z. B. DML-Code, der aus den DB2-Dienstprogrammen, gespeicherten SYSPROC-Prozeduren, internem SQL-Code usw. abgesetzt wird) von diesem Schwellenwert nicht betroffen.

Der Datenserver fasst IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors als Benutzerlogik auf. Für Aktivitäten, die über IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors aufgerufen werden, gelten Schwellenwerte.

Schwellenwert ACTIVITYTOTALTIME

Der Schwellenwert ACTIVITYTOTALTIME gibt die Zeitdauer an, die der Datenserver höchstens für die Verarbeitung einer Aktivität aufwenden darf.

Typ Aktivität

Definitionsdomäne

Datenbank, Servicesuperklasse, Serviceunterklasse, Arbeitsaktion

Umsetzungsbereich

Datenbank

Überwachte Arbeit

Erkannte Koordinator- und verschachtelte Aktivitäten (siehe „Aktivitäten“ auf Seite 20)

Warteschlangensteuerung

Nein

Einheit

Dauer in Minuten, Stunden oder Tagen

Prädiktiv oder reaktiv

Reaktiv

In Situationen, in denen die Aktivität von einem Warteschlangenschwellenwert in die Warteschlange gestellt wird, umfasst die Gesamtzeit der Aktivität auch die Zeit, während der die Aktivität in der Warteschlange auf die Ausführung wartet. Wenn ein Cursor geöffnet wird, dauert die zugehörige Aktivität so lange, bis der Cursor geschlossen wird.

Dieser Schwellenwert hat eine Unterteilung von 5 Minuten, so dass jeder Wert, den Sie für diesen Schwellenwert angeben, auf das nächste Mehrfache von 5 Minuten ungleich Null gerundet wird.

Wenn ein Zeitschwellenwert auf eine gespeicherte Prozedur angewendet wird, bezieht er sich auch auf Arbeit, die innerhalb der gespeicherten Prozedur ausgeführt wird. Wenn daher der Zeitschwellenwert für eine gespeicherte Prozedur abläuft, wird auch alle Arbeit in dieser gespeicherten Prozedur gestoppt. Hierarchien von Aufrufen gespeicherter Prozeduren können zu Hierarchien von Zeitschwellenwerten führen, die auf Aktivitäten angewendet werden, die in der untersten Verschachtelungsebene ausgeführt werden. Es wird immer der restriktivste Zeitschwellenwert in der Hierarchie (d. h. der Zeitschwellenwert mit dem kürzesten Endtermin) angewendet.

Der Datenserver fasst IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors als Benutzerlogik auf. Für Aktivitäten, die über IMPORT, EXPORT und andere Befehle des Befehlszeilenprozessors aufgerufen werden, gelten Schwellenwerte.

Auflösung des Aktivitätsschwellenwertbereichs

Da Aktivitätsschwellenwerte auf einzelne Aktivitäten angewendet werden, muss eine Entscheidung getroffen werden, welcher Schwellenwert erzwungen werden

soll, wenn sich mehrere Schwellenwerte auf eine Aktivität beziehen. Dieses Problem gibt es nicht bei zusammengefassten Schwellenwerten, da dieselbe Aktivität gleichzeitig zu mehreren Zusammenfassungen beitragen kann (z. B. bei Schwellenwerten für gemeinsamen Zugriff).

Die Auflösung des Aktivitätsschwellenwerts auf eine ausgeführte Aktivität folgt der Regel, dass ein in einer lokalen Domäne definierter Wert alle Werte aus einer breiter gefassten oder globaleren Domäne außer Kraft setzt. Im Folgenden ist eine Hierarchie von Domänen von lokal bis global aufgeführt:

1. Workload
2. Serviceunterklasse
3. Servicesuperklasse
4. Arbeitsaktion
5. Datenbank

So kann z. B. eine maximale Ausführungszeit von 1 Stunde für alle in der Datenbankdomäne definierten Datenbankabfragen von einer maximalen Ausführungszeit von 5 Stunden für die Servicesuperklasse LARGE QUERIES außer Kraft gesetzt werden, die wiederum von einer maximalen Ausführungszeit von 10 Stunden für die Serviceunterklasse VERY LARGE QUERIES außer Kraft gesetzt werden kann. Analog dazu kann die in der Datenbankdomäne definierte maximale Ausführungszeit von 1 Stunde von einem Wert von 10 Minuten in einer anderen Servicesuperklasse für wichtige Abfragen, die schnell ausgeführt werden, außer Kraft gesetzt werden.

Zusammengefasste Schwellenwerte

Schwellenwert TOTALDBPARTITIONCONNECTIONS

Der Schwellenwert TOTALDBPARTITIONCONNECTIONS gibt die maximale Anzahl gleichzeitig bestehender Datenbankverbindungen auf einer Koordinatorpartition für eine Datenbank an. Dieser Schwellenwert steuert also die maximale Anzahl von Clients, die auf jeder ihrer Datenbankpartitionen eine Verbindung zur Datenbank herstellen können. Dieser Schwellenwert wird für Benutzer mit der Berechtigung DBADM nicht erzwungen.

Typ Zusammenfassung

Definitionsdomäne

Datenbank

Umsetzungsbereich

Datenbankpartition

Überwachte Arbeit

Verbindungen

Warteschlangensteuerung

Ja (erzwungen bei 0)

Einheit

Anzahl gleichzeitig bestehender Verbindungen

Prädiktiv oder reaktiv

Prädiktiv

Wenn Sie z. B. den Schwellenwert `TOTALDBPARTITIONCONNECTIONS` auf 10 setzen und die Datenbank fünf Partitionen hat, können auf jeder Partition bis zu 10 Clients gleichzeitig verbunden sein, d. h. insgesamt 50 Clientverbindungen für die gesamte Datenbank.

Der Schwellenwert `TOTALDBPARTITIONCONNECTIONS` steuert nur Koordinatorverbindungen. Verbindungen, die von Subagenten hergestellt werden, werden nicht in den Schwellenwert eingerechnet.

Dieser Schwellenwert ist hilfreich für Situationen, in denen Sie mehrere Datenbanken in derselben Instanz haben wollen. Durch das Festlegen des Schwellenwerts `TOTALDBPARTITIONCONNECTIONS` auf einer Datenbankpartition wird sichergestellt, dass Clientverbindungen von einer Datenbank nicht alle verfügbaren Verbindungen in einer Datenbankpartition verwenden können.

Stellen Sie sicher, dass Sie den Konfigurationsparameter des Datenbankmanagers `max_connections` hoch genug eingestellt haben, um die maximale Anzahl von Verbindungen zu unterstützen, die Sie für die gesamte Datenbank erwarten. Wenn Sie einen Schwellenwert `TOTALDBPARTITIONCONNECTIONS` für eine Datenbank festlegen, müssen Sie `max_connections` auf mindestens denselben Wert einstellen. Wenn Sie mehrere Datenbanken auf derselben Instanz ausführen wollen, müssen Sie sicherstellen, dass Sie `max_connections` hoch genug einstellen, so dass er die maximale Anzahl von Verbindungen für alle Datenbanken unterstützt. Der Datenserver prüft diese Bedingung nicht, da es unmöglich ist, vorab zu wissen, wieviele der Datenbanken gleichzeitig aktiv sein werden.

Schwellenwert `TOTALSCPARTITIONCONNECTIONS`

Der Schwellenwert `TOTALSCPARTITIONCONNECTIONS` gibt die maximale Anzahl gleichzeitig bestehender Datenbankverbindungen auf einer Koordinatorpartition für eine Servicesuperklasse an.

Typ Zusammenfassung

Definitionsdomäne
Servicesuperklasse

Umsetzungsbereich
Datenbankpartition

Überwachte Arbeit
Verbindungen

Warteschlangensteuerung
Ja

Einheit
Anzahl gleichzeitig bestehender Verbindungen in einer Serviceklasse

Prädiktiv oder reaktiv
Prädiktiv

Wenn der Schwellenwert `TOTALSCPARTITIONCONNECTIONS` in der Serviceklasse erreicht wird, werden nachfolgende Koordinatorverbindungen, die mit der Servicesuperklasse verknüpft sind, in die Warteschlange gestellt, bis die festgelegte Warteschlangengröße erreicht ist. Standardmäßig ist die Warteschlangengröße Null, was heißt, dass keine Verbindungen in die Warteschlange gestellt werden können. Wenn eine Verbindung in die Warteschlange eines Schwellenwerts `TOTALSCPARTITIONCONNECTIONS` gestellt wird, gilt die Verbindung als *temporär*.

Überwachte Verbindungen sind sowohl neue Clientverbindungen als auch vorhandene Clientverbindungen, die von einer anderen Serviceklasse in diese Serviceklasse wechseln. Verbindungen wechseln zwischen Serviceklassen, indem sie mit einer anderen Workloaddefinition verknüpft werden, die einer anderen Serviceklasse zugeordnet ist. Die Workloadneuauswertung erfolgt nur an Transaktionsgrenzen, so dass Verbindungen nur an Transaktionsgrenzen zwischen Serviceklassen wechseln können; da jedoch Ressourcen, die einem WITH HOLD-Cursor zugeordnet sind, über Transaktionsgrenzen beibehalten werden, können Verbindungen mit einem offenen WITH HOLD-Cursor nicht zwischen Servicesuperklassen wechseln. Wenn der Verbindungskonzentrator aktiv ist, verlässt jede gewechselte Anwendung die Serviceklasse und gibt die Zugriffsberechtigung zurück, die vom Schwellenwert TOTALSCPARTITIONCONNECTIONS gehalten wird. Wenn die Anwendung bei der nächsten Anweisung eingewechselt wird, muss sie der Serviceklasse erneut beitreten und demzufolge den Schwellenwert passieren.

Wenn der Schwellenwert für die Warteschlangengröße erreicht ist, wird die Schwellenwertaktion ausgelöst. Der Schwellenwert TOTALSCPARTITIONCONNECTIONS steuert nur Koordinatorverbindungen. Verbindungen, die von Subagenten hergestellt werden, werden nicht in den Schwellenwert eingerechnet.

Wenn Sie einen Schwellenwert für TOTALDBPARTITIONCONNECTIONS festlegen, definieren Sie ihn groß genug, so dass er den Schwellenwert für TOTALSCPARTITIONCONNECTIONS aufnehmen kann. Wenn Sie z. B. fünf Servicesuperklassen für eine Datenbank definieren und jeder von ihnen einen Schwellenwert von 10 für TOTALSCPARTITIONCONNECTIONS hat, sollte TOTALDBPARTITIONCONNECTIONS mindestens einen Schwellenwert von 50 haben.

Schwellenwert **CONCURRENTWORKLOADOCCURRENCES**

Der Schwellenwert CONCURRENTWORKLOADOCCURRENCES ist ein zusammengefasster Schwellenwert, der die maximale Anzahl von Workloadvorkommen angibt, die gleichzeitig auf der Koordinatorpartition ausgeführt werden können.

Typ Zusammenfassung

Definitionsdomäne
Workload

Umsetzungsbereich
Datenbankpartition

Überwachte Arbeit
Workloadvorkommen

Warteschlangensteuerung
Nein

Einheit
Anzahl gleichzeitig ablaufender Workloadvorkommen

Prädiktiv oder reaktiv
Prädiktiv

Wenn beim Start eines Workloadvorkommens die von ihm generierte Arbeit an Nicht-Koordinatorpartitionen gesendet wird, zählt die Arbeit auf diesen Partitionen nicht zur Gesamtsumme des Schwellenwerts auf der Koordinatorpartition. Nehmen wir z. B. an, dass ein Schwellenwert CONCURRENTWORKLOADOCCURRENCES so definiert ist, dass nur ein Vorkommen der Workload A auf einer

Datenbankpartition zugelassen ist. Nehmen wir weiter an, dass eine Anwendung eine Verbindung zur Datenbankpartition 1 herstellt, wobei ein Vorkommen der Workload A gestartet wird, und dass diese Workload Arbeit an die Datenbankpartitionen 1, 2 und 3 sendet. In dieser Situation gibt es von Workload A ein Vorkommen auf der Datenbankpartition 1 und kein Vorkommen auf den Datenbankpartitionen 2 und 3. Wenn eine andere Anwendung eine Verbindung zur Datenbankpartition 1 herstellt und versucht wird, ein anderes Vorkommen von Workload A auf der Datenbankpartition 1 zu instanzieren, wird diese Workload daher zurückgewiesen. Auf den Datenbankpartitionen 2 und 3 können jedoch noch neue Vorkommen von Workload A gestartet werden.

Schwellenwert **CONCURRENTWORKLOADACTIVITIES**

Der Schwellenwert **CONCURRENTWORKLOADACTIVITIES** gibt die maximale Anzahl von Koordinatoraktivitäten und verschachtelten Aktivitäten an, die gleichzeitig in einem Workloadvorkommen ausgeführt werden können.

Typ Zusammenfassung

Definitionsdomäne

Workload

Umsetzungsbereich

Workloadvorkommen

Überwachte Arbeit

Erkannte Koordinatoraktivitäten und verschachtelte Aktivitäten (siehe „Aktivitäten“ auf Seite 20)

Warteschlangensteuerung

Nein

Einheit

Anzahl gleichzeitig ablaufender Workloadaktivitäten

Prädiktiv oder reaktiv

Prädiktiv

Dieser Schwellenwert bezieht sich auf ein einziges Workloadvorkommen. Wenn Sie mehrere Vorkommen einer Workload haben, die gleichzeitig ausgeführt werden, gilt der Schwellenwert separat für jedes Workloadvorkommen. Zu den verfolgten Aktivitäten gehören alle erkannten Koordinatoraktivitäten und alle verschachtelten Aktivitäten, die durch die Ausführung der Koordinatoraktivität generiert werden. Wenn z. B. eine gespeicherte Prozedur aufgerufen wird und diese gespeicherte Prozedur SQL-Code ausführt, zählen sowohl die Anweisung **CALL** (die Koordinatoraktivität) als auch die von der gespeicherten Prozedur ausgeführten SQL-Anweisungen (die verschachtelten Aktivitäten) zur Gesamtsumme des Schwellenwerts.

Dieser Schwellenwert bezieht sich nicht auf die Anweisungen **COMMIT**, **ROLLBACK** und **ROLLBACK to SAVEPOINT**.

Aspekte von verschachtelten Aktivitäten

Die verschachtelten Aktivitäten, die von diesem Schwellenwert verfolgt werden, müssen die folgenden Bedingungen erfüllen:

- Sie müssen eine erkannte Koordinatoraktivität sein. Verschachtelte Koordinatoraktivitäten, die laut der Beschreibung in „Arbeitsypen und SQL-Anweisungen von Arbeitsklassen“ auf Seite 91 keine erkannten Typen sind, werden nicht

gezählt. Analog dazu werden auch keine verschachtelten Subagentenaktivitäten wie RPC-Anforderungen, DSS-Anforderungen und verschachtelte DSS-Anforderungen gezählt.

- Sie müssen direkt von Benutzerlogik aufgerufen werden, z. B. einer benutzerdefinierten gespeicherten Prozedur, die SQL-Code absetzt, oder von der gespeicherten Prozedur SYSPROC.ADMIN_CMD. Verschachtelte Koordinatoraktivitäten, die durch den Aufruf eines DB2-Dienstprogramms oder von anderem Code in den Schemata SYSIBM, SYSFUN. oder SYSPROC gestartet werden, werden nicht zu der von diesem Schwellenwert festgelegten Obergrenze gezählt.

Beispiel

In diesem Beispiel ist der Maximalwert des Schwellenwerts CONCURRENT-WORK-LOADACTIVITIES auf 5 gesetzt. Durch die Benutzerlogik ergibt sich die folgende Operationsfolge in einem Workloadvorkommen:

1. Ein Befehl load wird abgesetzt: die aktuelle Anzahl Workloadaktivitäten beträgt 1.
 - Der Befehl load setzt intern SQL-Code ab. Die aktuelle Anzahl Workloadaktivitäten beträgt 1. (Von einem Dienstprogramm generierter SQL-Code wird beim Schwellenwert CONCURRENTWORKLOADACTIVITIES nicht berücksichtigt.)
 - Der Befehl load endet. Die aktuelle Anzahl Workloadaktivitäten beträgt 0.
2. Die gespeicherte Prozedur SYSPROC.SP1 wird mit CALL aufgerufen. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
 - Die gespeicherte Prozedur SYSPROC.SP1 generiert SQL-Code. Die aktuelle Anzahl Workloadaktivitäten beträgt 1. (Von einem Dienstprogramm generierter SQL-Code wird beim Schwellenwert CONCURRENTWORKLOADACTIVITIES nicht berücksichtigt.)
 - Die gespeicherte Prozedur SYSPROC.SP1 endet. Die aktuelle Anzahl Workloadaktivitäten beträgt 0.
3. Ein Cursor C1 wird geöffnet. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
4. Ein Befehl runstats wird abgesetzt. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
 - Der Befehl runstats generiert SQL-Code. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
 - Der Befehl runstats endet. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
5. Der Cursor C1 wird geschlossen. Die aktuelle Anzahl Workloadaktivitäten beträgt 0.
6. Die gespeicherte Prozedur BOB.SP1 wird mit CALL aufgerufen. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
 - Die gespeicherte Prozedur BOB.SP1 öffnet drei Cursor. Die aktuelle Anzahl Workloadaktivitäten beträgt 4.
 - Die gespeicherte Prozedur BOB.SP1 ruft die gespeicherte Prozedur SYSPROC.SP2 auf. Die aktuelle Anzahl Workloadaktivitäten beträgt 5.
 - Die gespeicherte Prozedur SYSPROC.SP2 setzt SQL-Code ab. Die aktuelle Anzahl Workloadaktivitäten beträgt 5.
 - Die gespeicherte Prozedur SYSPROC.SP2 endet. Die aktuelle Anzahl Workloadaktivitäten beträgt 4.

- Die gespeicherte Prozedur BOB.SP1 ruft die gespeicherte Prozedur BOB.SP2 auf. Die aktuelle Anzahl Workloadaktivitäten beträgt 5.
 - Die gespeicherte Prozedur BOB.SP2 setzt SQL-Code ab. Zu diesem Zeitpunkt wird der Schwellenwert ausgelöst.
 - Die gespeicherte Prozedur BOB.SP2 endet. Die aktuelle Anzahl Workloadaktivitäten beträgt 4.
 - Die gespeicherte Prozedur BOB.SP1 endet. Die aktuelle Anzahl Workloadaktivitäten beträgt 0.
7. Ein Cursor C2 wird geöffnet. Die aktuelle Anzahl Workloadaktivitäten beträgt 1.
 8. Die gespeicherte Prozedur BOB.SP2 wird mit CALL aufgerufen. Die aktuelle Anzahl Workloadaktivitäten beträgt 2.

Schwellenwert CONCURRENTDBCOORDACTIVITIES

Der Schwellenwert CONCURRENTDBCOORDACTIVITIES gibt die maximale Anzahl erkannter Koordinatoraktivitäten an, die gleichzeitig in allen Datenbankpartitionen in der angegebenen Definitionsdomäne ausgeführt werden können. Wenn eine Anwendung mehrere gleichzeitig ablaufende Aktivitäten startet, muss sie möglicherweise diesen Schwellenwert mehrmals passieren und verbraucht dadurch den für diesen Schwellenwert verfügbaren gemeinsamen Zugriff. Dies hat ein Szenario mit einem *Selbstdeadlock* zur Folge.

Typ Zusammenfassung

Definitionsdomäne

Datenbank, Arbeitsaktion, Servicesuperklasse, Serviceunterklasse

Umsetzungsbereich

Datenbank

Überwachte Arbeit

Erkannte Koordinatoraktivitäten und verschachtelte Aktivitäten (siehe „Arbeitsstypen und SQL-Anweisungen von Arbeitsklassen“ auf Seite 91)

Warteschlangensteuerung

Ja

Einheit

Anzahl gleichzeitig ablaufender Datenbankaktivitäten

Prädiktiv oder reaktiv

Prädiktiv

Dieser Schwellenwert ist eine Verallgemeinerung des Schwellenwerts CONCURRENTWORKLOADACTIVITIES. Der Schwellenwert CONCURRENTWORKLOADACTIVITIES bezieht sich nur auf Aktivitäten in einer Workloaddomäne. Den Schwellenwert CONCURRENTDBCOORDACTIVITIES können Sie dagegen auf eine Vielzahl von Domänen, von der gesamten Datenbank bis hin zu einer einzelnen Arbeitsaktion, anwenden. Ähnlich wie der Schwellenwert CONCURRENTWORKLOADACTIVITIES verfolgt der Schwellenwert CONCURRENTDBCOORDACTIVITIES Koordinatoraktivitäten und alle davon generierten verschachtelten Aktivitäten. Im Gegensatz zum Schwellenwert CONCURRENTWORKLOADACTIVITIES ist der Schwellenwert CONCURRENTDBCOORDACTIVITIES ein Warteschlangenschwellenwert.

Beim Erstellen von Warteschlangenschwellenwerten vom Typ `CONCURRENTDBCOORDACTIVITIES` müssen Sie sich bewusst sein, dass es Konfigurationen gibt, die zu Konkurrenzsituationen auf Grund von Warteschlangen führen. Zum Beispiel:

1. Ein Schwellenwert für gemeinsamen Zugriff vom Typ `CONCURRENTDBCOORDACTIVITIES` wird mit einem maximalen Wert für gemeinsamen Zugriff von 1 und einer Warteschlangengröße > 1 erstellt.
2. Eine Anwendung öffnet einen Cursor (oder ruft eine gespeicherte Prozedur auf), den (die) der DB2-Datenserver als Aktivität A1 erkennt, die die eindeutige für den Schwellenwert verfügbare Zugriffsberechtigung verbraucht.
3. Während die Aktivität A1 noch aktiv ist, setzt die Anwendung jetzt eine zweite SQL-Anweisung ab, die der Datenserver als Aktivität A2 erkennt und die auch dem Schwellenwert für gemeinsamen Zugriff unterliegt. Da die Aktivität A1 bereits aktiv ist, wird die neue Aktivität A2 in die Warteschlange gestellt.

Die Anwendung befindet sich jetzt in einer Situation, die nicht behoben werden kann. Sie wartet darauf, dass A2 ausgeführt wird, A2 wartet jedoch darauf, dass A1 fertig gestellt wird. Diese Situation kann ohne Eingriff von außen nicht behoben werden.

Dieses Beispiel kann auf mehrere Anwendungen und Warteschlangen verallgemeinert werden. Sie können diese Situation beheben, indem Sie die Werte für den gemeinsamen Zugriff erhöhen oder bestimmte Aktivitäten abbrechen, wenn die Werte für den gemeinsamen Zugriff korrekt festgelegt sind. Sie können auch Zeitschwellenwerte nutzen, um zu verhindern, dass eine Aktivität unendlich lange in der Warteschlange bleibt, wodurch Szenarios wie dieses ohne Eingriff von außen behoben werden können.

Auswertungsreihenfolge für Schwellenwerte

Obwohl bestimmte Schwellenwerte unabhängig von anderen Schwellenwerten ausgewertet werden (da sie von bestimmten Ereignissen wie einer neuen Verbindung oder einem neuen Workloadvorkommen gesteuert werden), sind andere Schwellenwerte voneinander abhängig und müssen in einer bestimmten Reihenfolge ausgewertet werden, wenn sie in derselben Datenbank definiert sind.

Die Auswertung von Schwellenwerten erfolgt folgendermaßen:

- `CONCURRENTWORKLOADOCCURRENCES`. Dieser Schwellenwert wird unabhängig ausgewertet, wenn ein neues Workloadvorkommen für eine Workloaddefinition gestartet wird, für die dieser Schwellenwert gilt.
- `TOTALDBPARTITIONCONNECTIONS`. Dieser Schwellenwert wird unabhängig ausgewertet, wenn eine neue Verbindung zur Datenbank hergestellt wird.
- `TOTALSPARTITIONCONNECTIONS`. Dieser Schwellenwert wird ausgewertet, wenn eine Verbindung einer Serviceklasse beitrifft (entweder eine neue Verbindung oder eine Übertragung zwischen Serviceklassen als Ergebnis einer erneuten Workloadzuordnung).

Alle übrigen Schwellenwerte basieren auf erkannten Aktivitäten aus einer SQL-Anweisung oder aus der Ausführung eines Dienstprogramms wie z. B. des Dienstprogramms `LOAD`. Prädiktive Schwellenwerte werden vor reaktiven Schwellenwerten geprüft, da vor dem Ausführen einer Datenbankaktivität durch eine Prüfung sichergestellt werden muss, dass nicht gegen prädiktive Schwellenwerte verstoßen wird. Prädiktive Schwellenwerte werden in der unten aufgeführten Reihenfolge ausgewertet.

Anmerkung: Wenn Sie keinen Schwellenwert definieren, wird dieser Schritt übersprungen. Außerdem können die beschriebenen Schritte aus Leistungsgründen während der Ausführung kombiniert werden.

1. Es wird geprüft, ob ein Schwellenwert ESTIMATEDSQLCOST vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Wenn Sie diesen Schwellenwert in mehreren Domänen definieren, wird der Schwellenwert gemäß den Regeln für die Bereichsauflösung aufgelöst. (Weitere Informationen finden Sie in „Auflösung des Aktivitätsschwellenwertbereichs“ auf Seite 67.) Das Ergebnis dieser Operation ist ein Wert von ESTIMATEDSQLCOST, der auf die Aktivität anwendbar ist.
2. Es wird geprüft, ob ein Schwellenwert ESTIMATEDSQLCOST vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Bei einem Verstoß gegen den Schwellenwert wird die entsprechende Aktion ausgeführt. Fahren Sie gegebenenfalls mit dem nächsten Schritt fort.
3. Es wird geprüft, ob ein Schwellenwert CONCURRENTWORKLOADACTIVITIES vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Wenn gegen den Schwellenwert verstoßen wird, wird die entsprechende Aktion ausgeführt. Fahren Sie gegebenenfalls mit dem nächsten Schritt fort.
4. Es wird geprüft, ob ein Schwellenwert CONCURRENTDBCOORDACTIVITIES vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Bei einem Verstoß gegen den Schwellenwert wird die entsprechende Aktion ausgeführt. Fahren Sie gegebenenfalls mit dem nächsten Schritt fort.
5. Es wird geprüft, ob ein Schwellenwert CONCURRENTDBCOORDACTIVITIES vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Bei einem Verstoß gegen den Schwellenwert wird die entsprechende Aktion ausgeführt. Fahren Sie gegebenenfalls mit dem nächsten Schritt fort.
6. Es wird geprüft, ob ein Schwellenwert CONCURRENTDBCOORDACTIVITIES vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Bei einem Verstoß gegen den Schwellenwert wird die entsprechende Aktion ausgeführt. Fahren Sie gegebenenfalls mit dem nächsten Schritt fort.
7. Es wird geprüft, ob ein Schwellenwert CONCURRENTDBCOORDACTIVITIES vorhanden ist, und wenn dies der Fall ist, ob gegen diesen Schwellenwert verstoßen wurde. Bei einem Verstoß gegen den Schwellenwert wird die entsprechende Aktion ausgeführt. Fahren Sie gegebenenfalls mit dem nächsten Schritt fort.

Die Auswertungsreihenfolge für Schwellenwerte für gemeinsamen Zugriff folgt nicht der Hierarchie für die Auflösung von Aktivitätsschwellenwerten. (Weitere Informationen finden Sie in „Auflösung des Aktivitätsschwellenwertbereichs“ auf Seite 67.) Die Schwellenwerte für gemeinsamen Zugriff des Arbeitsaktionssets auf Datenbankebene werden zuerst geprüft, um die folgende Situation zu vermeiden. Nehmen wir an, dass die folgenden Schwellenwerte definiert sind:

- Ein Arbeitsaktionsschwellenwert für gemeinsamen Zugriff für LOAD-Aktivitäten ist mit dem Wert 1 definiert.
- Der Grenzwert für gemeinsamen Zugriff der Servicesuperklasse S1 ist auf 10 gesetzt.

Nehmen wir weiter an, dass eine LOAD-Aktivität bereits in der Datenbank (unter einer beliebigen Servicesuperklasse) ausgeführt wird und dass bereits neun Aktivitäten in der Servicesuperklasse S1 ausgeführt werden, wenn eine neue LOAD-Akti-

vität als 10. Aktivität startet. Wenn die Hierarchie für die Bereichsauflösung für die Schwellenwertauswertung verwendet wird, würde die eingehende LOAD-Aktivität nicht gegen den Serviceklassenschwellenwert verstoßen und den gemeinsamen Zugriff auf 10 erhöhen. Die LOAD-Aktivität wird dann mit dem Grenzwert für gemeinsamen Zugriff des Arbeitsaktionsschwellenwerts ausgewertet. Gegen diesen wird verstoßen, da bereits eine LOAD-Aktivität in der Datenbank ausgeführt wird und der Wert für gemeinsamen Zugriff für den Arbeitsaktionsschwellenwert nur 1 beträgt. Die 10. LOAD-Aktivität wird nun in die Warteschlange gestellt.

Diese Situation führt dazu, dass alle neu in der Servicesuperklasse S1 eintreffenden Aktivitäten nun in die Warteschlange gestellt werden (da der Serviceklassengrenzwert für gemeinsamen Zugriff bereits erreicht ist). Die Schwellenwertwarteschlange der Arbeitsaktion wirkt sich auf die Serviceklasse aus. Dies ist nicht erwünscht, da Aktivitäten, die versuchen, in der Serviceklasse ausgeführt zu werden, nicht notwendigerweise eine Beziehung zur Schwellenwertbedingung der Arbeitsaktion haben (beispielsweise sollte eine Einfügeoperation, die versucht, in der Servicesuperklasse S1 ausgeführt zu werden, nicht auf eine LOAD-Aktivität warten müssen, die auf Grund einer Arbeitsaktions-Schwellenwertbedingung in die Warteschlange gestellt wurde). Der Arbeitsaktionsschwellenwert für gemeinsamen Zugriff sollte daher zuerst geprüft werden, um solche Situationen zu vermeiden. Da der Schwellenwert für gemeinsamen Zugriff zuerst geprüft wird, wird die 10. Aktivität in der Serviceklasse (bei der es sich zufällig um eine LOAD-Aktivität handelt) auf der Ebene des Arbeitsaktionsschwellenwerts blockiert, bevor sie versuchen kann, einen Platz in der Servicesuperklasse S1 einzunehmen.

Aspekte von reaktiven Schwellenwerten

Reaktive Schwellenwerte werden in diskreter Weise beim Ausführen einer Aktivität ausgewertet. Für die Auswertung des reaktiven Schwellenwerts `SQLTEMPSPACE`, `SQLROWSRETURNED` oder `ACTIVITYTOTALTIME` gibt es keine spezifische Reihenfolge.

Arbeiten mit Schwellenwerten

Erstellen eines Schwellenwerts

Mit der DDL-Anweisung `CREATE THRESHOLD` können Sie Schwellenwerte erstellen. Mit einem Schwellenwert legen Sie einen Grenzwert für die Ressourcennutzung fest.

Zum Erstellen eines Schwellenwerts benötigen Sie die Berechtigung `DBADM` oder `SYSADM`.

Weitere Informationen zu Voraussetzungen finden Sie in den folgenden Abschnitten:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Verwenden Sie die Anweisung `CREATE WORK ACTION SET` oder die Anweisung `ALTER WORK ACTION SET` mit den Schlüsselwörtern `ADD WORK ACTION`, um einen Schwellenwert für ein Arbeitsaktionsset zu erstellen. Weitere Informationen finden Sie in Anweisung `CREATE WORK ACTION SET` oder Anweisung `ALTER WORK ACTION SET`.

Gehen Sie wie folgt vor, um einen Schwellenwert zu erstellen:

1. Setzen Sie die Anweisung CREATE THRESHOLD ab, und geben Sie dabei mindestens eines des folgenden Merkmale für den Schwellenwert an:
 - Den Namen des Schwellenwerts.
 - Die Schwellenwertdomäne. Die Schwellenwertdomäne ist das Datenbankobjekt, dem der Schwellenwert zugeordnet ist und auf das er angewendet wird.

Anmerkung: Welche Domäne gilt, hängt von der Art des Schwellenwerts ab. Weitere Informationen finden Sie in „Schwellenwertzusammenfassung“ auf Seite 63.

Die folgenden Domänen sind möglich:

- Datenbank
- Servicesuperklasse
- Serviceunterklasse
- Workload

Weitere Informationen finden Sie in Kapitel 4, „Schwellenwerte“, auf Seite 61.

- Den Umsetzungsbereich für den Schwellenwert. Der Schwellenwertbereich ist der Umsetzungsbereich des Schwellenwerts in seiner Domäne.

Anmerkung: Welcher Umsetzungsbereich gilt, hängt von der Art des Schwellenwerts ab. Weitere Informationen finden Sie in „Schwellenwertzusammenfassung“ auf Seite 63.

Die folgenden Umsetzungsbereiche sind möglich:

- Datenbank
- Datenbankpartition
- Workloadvorkommen

Weitere Informationen finden Sie in Kapitel 4, „Schwellenwerte“, auf Seite 61.

- Optional: Einen Wert, der angibt, ob der Schwellenwert aktiviert oder inaktiviert ist. Standardmäßig wird der Schwellenwert als aktiviert erstellt. Wenn Sie den Schwellenwert als inaktiviert erstellen und ihn später aktivieren wollen, verwenden Sie die Anweisung ALTER THRESHOLD. Weitere Informationen finden Sie in „Ändern eines Schwellenwerts“ auf Seite 82.

Anmerkung: Wenn Sie Schwellenwerte für Arbeitsaktionen haben, können Sie diese mit der Anweisung ALTER WORK ACTION SET aktivieren und inaktivieren.

- Das Schwellenwertvergleichselement, um den Maximalwert für den Schwellenwert anzugeben. Wenn der Maximalwert erreicht wird, wird die für den Schwellenwert angegebene Aktion erzwungen. Es gibt folgende Schwellenwerte:
 - TOTALDBPARTITIONCONNECTIONS. Geben Sie 0 oder eine positive ganze Zahl an, um die maximale Anzahl gleichzeitig bestehender Koordinatorverbindungen anzugeben, die auf einer Datenbankpartition ausgeführt werden können. Ein Wert von 0 verhindert, dass neue Koordinatorverbindungen eine Verbindung herstellen. Die Definitionsdomäne muss DATABASE sein. Der Umsetzungsbereich muss PARTITION lauten. Wenn Sie z. B. angeben wollen, dass fünf Koordinatorverbindungen gleichzeitig auf einer Datenbankpartition ausgeführt werden können, geben Sie Folgendes ein:

```
TOTALDBPARTITIONCONNECTIONS > 5
```

Anmerkung: Dieser Schwellenwert gilt nicht für Benutzer mit der Berechtigung DBADM.

- TOTALSPARTITIONCONNECTIONS. Geben Sie 0 oder eine positive ganze Zahl an, um die maximale Anzahl gleichzeitig bestehender Koordinatorverbindungen anzugeben, die auf einer Datenbankpartition in einer Servicesuperklasse ausgeführt werden können. Ein Wert von 0 verhindert, dass neue Verbindungen mit der Servicesuperklasse verknüpft werden. Die Definitionsdomäne muss SERVICE SUPERCLASS sein. Der Umsetzungsbereich muss PARTITION lauten. Da dieser Schwellenwert ein Warteschlangenschwellenwert ist, hat er ein optionales Schlüsselwort AND QUEUEDCONNECTIONS. Die folgenden Argumente sind gültig:
 - AND QUEUEDCONNECTIONS > 0 gibt an, dass Koordinatorverbindungen nicht in die Warteschlange gestellt werden.
 - AND QUEUEDCONNECTIONS > *ganzzahl* gibt die maximale Warteschlangengröße an, wenn die Anzahl der Koordinatorverbindungen überschritten wird.
 - AND QUEUEDCONNECTIONS UNBOUNDED gibt an, dass es keine Maximalgröße für die Warteschlange gibt. In dieser Situation wird die Bedingung für den Schwellenwertverstoß nie erfüllt.

Wenn Sie z. B. angeben wollen, dass fünf Koordinatorverbindungen gleichzeitig auf einer Datenbankpartition in einer Servicesuperklasse ausgeführt werden und dass sieben Verbindungen in die Warteschlange gestellt werden können, geben Sie Folgendes ein:

```
TOTALSPARTITIONCONNECTIONS > 5 AND QUEUEDCONNECTIONS > 7
```

Anmerkung: Sie können mit TOTALSPARTITIONCONNECTIONS Serviceklassen praktisch inaktivieren, die sich nicht manuell inaktivieren lassen (z. B. die Standardbenutzerklasse). Obwohl Schwellenwerte nicht für Benutzer mit der Berechtigung DBADM in der Serviceklasse SYSDEFAULTADMWORKLOAD gelten, ist eine inaktivierte Serviceklasse für keinen Benutzer verfügbar.

- CONNECTIONIDLETIME. Geben Sie eine positive ganze Zahl und eines der folgenden Schlüsselwörter an, um die Höchstdauer anzugeben, während der eine Verbindung inaktiv sein kann:
 - DAY
 - DAYS
 - HOUR
 - HOURS
 - MINUTE
 - MINUTES

Da die minimale Unterteilung 5 MINUTES beträgt, werden alle von Ihnen angegebenen Werte auf das nächste Mehrfache von 5 Minuten ungleich Null gerundet. Die Definitionsdomäne muss DATABASE oder SERVICE SUPERCLASS sein. Der Umsetzungsbereich muss DATABASE lauten. Wenn Sie z. B. angeben wollen, dass eine Verbindung 90 Minuten lang inaktiv sein kann, geben Sie Folgendes ein:

```
CONNECTIONIDLETIME > 90 MINUTES
```

Der Maximalwert, der für diesen Schwellenwert angegeben werden kann, beträgt 2147483400 Sekunden. Jeder (mit dem Schlüsselwort DAY, HOUR, MINUTE oder SECOND) angegebene Wert, der größer als 2147483400 Sekunden ist, wird auf den Maximalwert begrenzt.

- CONCURRENTWORKLOADOCCURRENCES. Geben Sie eine positive ganze Zahl ungleich Null an, um die maximale Anzahl gleichzeitig ablaufender Workloadvorkommen auf einer Datenbankpartition anzugeben. Die Definitionsdomäne muss WORKLOAD sein. Wenn Sie z. B. angeben wollen, dass maximal acht Vorkommen einer Workload gleichzeitig auf einer Datenbankpartition auftreten können, geben Sie Folgendes ein:

CONCURRENTWORKLOADOCCURRENCES > 8

- CONCURRENTWORKLOADACTIVITIES. Geben Sie eine positive ganze Zahl ungleich Null an, um die maximale Anzahl gleichzeitig ablaufender Koordinator- und verschachtelter Aktivitäten auf einer Datenbankpartition für ein Workloadvorkommen anzugeben. Der Umsetzungsbereich muss WORKLOAD OCCURRENCE lauten. Wenn Sie z. B. angeben wollen, dass maximal 26 Aktivitäten gleichzeitig auf einer Datenbankpartition für ein Workloadvorkommen ausgeführt werden können, geben Sie Folgendes ein:

CONCURRENTWORKLOADACTIVITIES > 26

- CONCURRENTDBCOORDACTIVITIES. Geben Sie 0 oder eine positive ganze Zahl an, um die maximale Anzahl gleichzeitig ablaufender Koordinatoraktivitäten anzugeben, die auf allen Datenbankpartitionen in der angegebenen Domäne ausgeführt werden können. Ein Wert von 0 verhindert, dass neue Koordinatoraktivitäten ausgeführt werden (einschließlich aller Aktivitäten zum Inaktivieren oder Ändern dieses Schwellenwerts). In dieser Situation müssen Sie die Workload SYSDEFAULTADMWORKLOAD verwenden, um den Schwellenwert zu inaktivieren oder zu ändern. Die Definitionsdomäne kann DATABASE, SERVICE SUPERCLASS oder SERVICE SUBCLASS sein. Der Umsetzungsbereich muss DATABASE lauten. Da dieser Schwellenwert ein Warteschlangenschwellenwert ist, hat er ein optionales Schlüsselwort AND QUEUEDACTIVITIES. Die folgenden Argumente sind gültig:
 - AND QUEUEDACTIVITIES > 0 gibt an, dass Koordinatoraktivitäten nicht in die Warteschlange gestellt werden.
 - AND QUEUEDACTIVITIES > *ganzzahl* gibt die maximale Warteschlangengröße an, wenn die Anzahl der Koordinatoraktivitäten überschritten wird.
 - AND QUEUEDACTIVITIES UNBOUNDED gibt an, dass es keine Maximalgröße für die Warteschlange gibt. In dieser Situation wird die Bedingung für den Schwellenwertverstoß nie erfüllt.

Wenn Sie z. B. angeben wollen, dass 12 Koordinatoraktivitäten gleichzeitig in der angegebenen Datenbankdomäne ausgeführt werden und dass 9 Aktivitäten in die Warteschlange gestellt werden können, geben Sie Folgendes ein:

CONCURRENTDBCOORDACTIVITIES > 12 AND QUEUEDACTIVITIES > 9

- ESTIMATEDSQLCOST. Geben Sie eine positive große ganze Zahl ungleich Null an, um den maximalen dem Optimierungsprogramm zugeordneten Aufwand einer DML-Koordinatoraktivität oder einer DML-Aktivität anzugeben, die von Benutzerlogik aufgerufen werden kann. Die Definitionsdomäne kann DATABASE, SERVICE SUPERCLASS oder SERVICE SUBCLASS sein. Der Umsetzungsbereich muss DATABASE lauten. Wenn Sie z. B. angeben wollen, dass keine DML-Aktivität mit mehr als 1 000 Timeons in der Datenbank ausgeführt werden kann, geben Sie Folgendes ein:

ESTIMATEDSQLCOST > 1000

- SQLROWSRETURNED. Geben Sie eine positive ganze Zahl ungleich Null an, um die maximale Anzahl von Zeilen festzulegen, die vom

Anwendungsserver an die Clientanwendung zurückgegeben werden kann. Die Definitionsdomäne kann DATABASE, SERVICE SUPERCLASS oder SERVICE SUBCLASS sein. Der Umsetzungsbereich muss DATABASE lauten. Wenn Sie z. B. angeben wollen, dass höchstens 50 000 Zeilen zurückgegeben werden können, geben Sie Folgendes ein:

```
SQLROWSRETURNED > 50000
```

- ACTIVITYTOTALTIME. Geben Sie eine positive ganze Zahl ungleich Null an, um die Höchstdauer anzugeben, während der eine Aktivität ausgeführt werden kann, einschließlich der Zeitdauer, während der die Aktivität in der Warteschlange sein kann. Nach diesem Wert steht eines der folgenden Schlüsselwörter für die Dauer:
 - DAY
 - DAYS
 - HOUR
 - HOURS
 - MINUTE
 - MINUTES

Da die minimale Unterteilung 5 MINUTES beträgt, werden alle angegebenen Werte auf das nächste Mehrfache von 5 Minuten ungleich Null gerundet. Die Definitionsdomäne kann DATABASE, SERVICE SUPERCLASS oder SERVICE SUBCLASS sein. Der Umsetzungsbereich muss DATABASE lauten. Wenn Sie z. B. eine maximale Aktivitätsdauer von zwei Stunden angeben wollen, geben Sie Folgendes ein:

```
ACTIVITYTOTALTIME > 2 HOURS
```

Der Maximalwert, der für diesen Schwellenwert angegeben werden kann, beträgt 2147483400 Sekunden. Jeder (mit dem Schlüsselwort DAY, HOUR, MINUTE oder SECOND) angegebene Wert, der größer als 2147483400 Sekunden ist, wird auf den Maximalwert begrenzt.

- SQLTEMPSPACE. Geben Sie eine positive ganze Zahl ungleich Null an, um den maximalen Tabellenbereich für temporäre Tabellen festzulegen, der in einer Datenbankpartition verbraucht werden kann. Nach diesem Wert steht eines der folgenden Schlüsselwörter für den Platz:
 - K (Kilobyte)
 - M (Megabyte)
 - G (Gigabyte)

Die Definitionsdomäne kann DATABASE, SERVICE SUPERCLASS oder SERVICE SUBCLASS sein. Der Umsetzungsbereich muss DATABASE PARTITION lauten. Wenn Sie z. B. angeben wollen, dass der maximale Tabellenbereich für temporäre Tabellen in einer Datenbankpartition 100 MB groß sein soll, geben Sie Folgendes ein:

```
SQLTEMPSPACE > 100 M
```

- Die Aktionen, die bei einem Verstoß gegen die Schwellenwertbegrenzung auszuführen sind. Die Aktionen bestehen aus einer obligatorischen Fortschrittsaktion und einer optionalen Aktivitätserfassungsaktion. Die Aktivitätserfassungsaktion gibt an, welche Informationen für die Aktivität erfasst werden sollen, die den Verstoß gegen die Schwellenwertbegrenzung verursachte. Geben Sie eine Aktion nur für aktivitätsbezogene Schwellenwerte an; eine Aktion wird ignoriert, wenn Sie sie für andere Schwellenwerte angeben.

- STOP EXECUTION. Bei den Schwellenwerten TOTALDBPARTITIONCONNECTIONS und TOTALSCPARTITIONCONNECTIONS wird verhindert, dass die Verbindung hergestellt wird. Beim Schwellenwert CONNECTIONIDLETIME wird die Verbindung geschlossen. Bei CONCURRENTWORKLOADOCCURRENCES wird verhindert, dass ein neues Workloadvorkommen erstellt wird. Bei allen anderen aktivitätsbezogenen Schwellenwerten wird die Aktivität gestoppt, die den Schwellenwertverstoß verursachte. Wenn ein THRESHOLD VIOLATIONS-Ereignismonitor aktiv ist, wird ein Datensatz in den Ereignismonitor geschrieben, der angibt, dass gegen den Schwellenwert verstoßen wurde.
- CONTINUE. Wenn ein THRESHOLD VIOLATIONS-Ereignismonitor aktiv ist, wird ein Datensatz in den Ereignismonitor geschrieben, der angibt, dass gegen den Schwellenwert verstoßen wurde. Andernfalls erfolgt keine weitere Aktion.

Anmerkung: Wenn als Schwellenwertaktion CONTINUE für einen Warteschlangenschwellenwert angegeben ist, hebt diese die Begrenzung der Warteschlangengröße unabhängig von einem festen Wert für die Warteschlangengröße effektiv auf.

- COLLECT ACTIVITY DATA. Die Informationen, die für den Ereignismonitor für Aktivitäten erfasst werden sollen. Die Standardeinstellung ist COLLECT ACTIVITY DATA NONE.
 - NONE. Es werden keine Aktivitätsinformationen erfasst, wenn die Aktivität abgeschlossen wird, die den Schwellenwertverstoß verursachte.
 - Die Position, an der Daten erfasst werden sollen:
 - ON COORDINATOR DATABASE PARTITION. Aktivitätsdaten werden nur auf der Koordinatorpartition für die Aktivität erfasst, wenn ein ACTIVITIES-Ereignismonitor auf dieser Datenbankpartition aktiv ist.
 - ON ALL DATABASE PARTITIONS. Für prädiktive Schwellenwerte, die als Aktion nur CONTINUE haben, werden Aktivitätsdaten auf allen Datenbankpartitionen erfasst, in denen die Aktivität verarbeitet wird. Aktivitätsdetails oder -werte werden jedoch nur auf der Koordinatorpartition erfasst. Bei reaktiven Schwellenwerten und bei allen Schwellenwerten mit der Aktion STOP EXECUTING hat diese Option dieselbe Auswirkung wie COORDINATOR DATABASE PARTITION. Aktivitätsdaten werden nur auf den Datenbankpartitionen erfasst, auf denen ein ACTIVITIES-Ereignismonitor aktiv ist.
 - WITHOUT DETAILS. Informationen zu jeder Aktivität, die den Schwellenwertverstoß verursachte, werden an den entsprechenden Ereignismonitor gesendet, wenn die Ausführung der Aktivität abgeschlossen ist. Informationen zur Anweisung und der Kompilierumgebung werden jedoch nicht an den Ereignismonitor gesendet.
 - WITH DETAILS. Informationen zur Anweisung und zur Kompilierumgebung werden an den entsprechenden Ereignismonitor für Aktivitäten gesendet, für die diese Details verfügbar sind. Wenn Sie Details anfordern, können Sie auch AND VALUES angeben, damit für Aktivitäten mit Eingabedatenwerten diese Eingabedatenwerte an den entsprechenden Ereignismonitor gesendet werden.

Wenn z. B. alle verfügbaren Informationen zur Aktivität, die gegen die Schwellenwertbedingung verstoßen hat, (einschließlich der Anweisung, die den Schwellenwertverstoß verursachte, der Kompilierumgebung und der Eingabedatenwerte) nach Abschluss der Aktivität an den Aktivitäts-

ereignismonitor gesendet werden sollen, verwenden Sie die folgenden Schlüsselwörter in der Anweisung CREATE THRESHOLD:

COLLECT ACTIVITY DATA ON ALL DATABASE PARTITIONS WITH DETAILS AND VALUES

2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird der Schwellenwert zur Sicht SYSCAT.THRESHOLDS hinzugefügt.

Ändern eines Schwellenwerts

Mit der DDL-Anweisung ALTER THRESHOLD können Sie Schwellenwerte ändern. Sie könnten beispielsweise einen Schwellenwert ändern, wenn dieser nicht die gewünschten Resultate erzielt.

Zum Ändern eines Schwellenwerts benötigen Sie die Berechtigung DBADM oder SYSADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Verwenden Sie die Anweisung ALTER WORK ACTION SET mit dem Schlüsselwort ADD WORK ACTION, um einen Schwellenwert für ein Arbeitsaktionsset zu ändern. Weitere Informationen finden Sie in Anweisung ALTER WORK ACTION SET.

Gehen Sie wie folgt vor, um einen Schwellenwert zu ändern:

1. Geben Sie mindestens eines der folgenden Merkmale für den Schwellenwert in der Anweisung ALTER THRESHOLD an. Sie können die folgenden Merkmale ändern. Eine Erläuterung der unterstützten Werte für diese Merkmale finden Sie in „Erstellen eines Schwellenwerts“ auf Seite 76.
 - Die neue Begrenzung für das Schwellenwertvergleichselement.

Anmerkung: Sie können den Schwellenwerttyp nicht ändern (d. h. Sie können einen Schwellenwert TOTALDBPARTITIONCONNECTIONS nicht in einen Schwellenwert TOTALSCPARTITIONCONNECTIONS ändern).

- Die Aktionen, die bei einem Verstoß gegen die Schwellenwertbegrenzung auszuführen sind.
 - Eine Angabe, ob der Schwellenwert aktiviert ist. Aktivitäten werden unter den entsprechenden Schwellenwerten ausgeführt, die beim Starten der jeweiligen Aktivität aktiviert wurden.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird der Schwellenwert in der Sicht SYSCAT.THRESHOLDS aktualisiert.

Löschen eines Schwellenwerts

Sie können einen Schwellenwert, den Sie nicht mehr benötigen, mit der DDL-Anweisung DROP THRESHOLD löschen.

Zum Löschen eines Schwellenwerts benötigen Sie die Berechtigung DBADM oder SYSADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Anmerkung: Wenn Sie einen Schwellenwert in einem Arbeitsaktionsset löschen wollen, verwenden Sie die Anweisung ALTER WORK ACTION SET.

Gehen Sie wie folgt vor, um einen Schwellenwert zu löschen:

1. Wenn es sich bei dem Schwellenwert um einen Warteschlangenschwellenwert handelt, inaktivieren Sie ihn mit der ALTER THRESHOLD.
2. Wenn Sie einen Warteschlangenschwellenwert mit einer Anweisung ALTER THRESHOLD inaktiviert haben, führen Sie ein COMMIT durch, um die Änderung festzuschreiben.
3. Löschen Sie den Schwellenwert mit der Anweisung DROP THRESHOLD.
4. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird der Schwellenwert aus der Sicht SYSCAT.THRESHOLDS entfernt.

Kapitel 5. Arbeitsaktionssets, Arbeitsaktionen, Arbeitsklassensets und Arbeitsklassen

Mit Arbeitsaktionssets, Arbeitsaktionen, Arbeitsklassensets und Arbeitsklassen können Sie Aktivitäten danach klassifizieren, *was* sie sind, ähnlich wie Sie mit Workloads Aktivitäten danach klassifizieren können, *wer* sie übergeben hat. Wenn Sie Aktivitäten basierend auf Attributen wie ihrem Aktivitätstyp klassifizieren, können Sie diese Aktivitäten unterschiedlich behandeln, indem Sie unterschiedliche Arten von Aktionen auf sie anwenden.

Sie können Aktivitäten in unterschiedlichen Arbeitsklassen klassifizieren und Arbeitsklassen in Arbeitsklassensets gruppieren. Wenn Sie ein Arbeitsaktionsset, das Arbeitsaktionen enthalten kann, auf ein Arbeitsklassenset anwenden, können die Arbeitsaktionen im Arbeitsaktionsset auf die Arbeitsklassen im Arbeitsklassenset angewendet werden. Da die Objekte, die die Aktivitäten klassifizieren (Arbeitsklassen und Arbeitsklassensets) sich von den Objekten unterscheiden, die die auf die Aktivitäten anzuwendenden Aktionen definieren (Arbeitsaktionen und Arbeitsaktionssets), können die Klassifizierungsobjekte von mehreren Arbeitsaktionssets und Arbeitsaktionen gemeinsam genutzt werden.

Arbeitsklassen und Arbeitsklassensets

Eine Arbeitsklasse ist eine Methode zur Kategorisierung einzelner Datenbankaktivitäten basierend auf Attributen der Datenbankaktivität. Arbeitsklassen werden in Arbeitsklassensets gruppiert, die von unterschiedlichen Arbeitsaktionssets gemeinsam genutzt werden können.

Beispiele von Datenbankaktivitätsattributen, die bestimmen können, mit welcher Arbeitsklasse eine Aktivität verknüpft ist, sind unter anderem: Aktivitätstyp (DDL, DML, LOAD), geschätzter Aufwand (sofern verfügbar), geschätzte Kardinalität (sofern verfügbar) und Schema (sofern verfügbar).

Eine Arbeitsklasse hat die folgenden Attribute:

- Den Arbeitsklassennamen, der im Arbeitsklassenset eindeutig sein muss.
- Die Datenbankaktivitätsattribute, die aus den folgenden Informationen bestehen:
 - Dem Typ von Datenbankaktivität, der unter diese Arbeitsklasse fällt.
Mit vordefinierten Schlüsselwörtern (z. B. READ, WRITE, DML, DDL und LOAD) können Sie Datenbankanforderungen in unterschiedliche Kategorien klassifizieren. Unterschiedliche Typen von Datenbankaktivitäten können je nach Arbeitstyp mit einer Arbeitsklasse verknüpft werden. Das Schlüsselwort WRITE z. B. umfasst Aktualisierungen, Löschvorgänge, Einfügungen, Zusammenführungen sowie Auswahlvorgänge, die einen Löschvorgang, eine Einfügung oder eine Aktualisierung enthalten. Weitere Informationen finden Sie in „Arbeitsstypen und SQL-Anweisungen von Arbeitsklassen“ auf Seite 91.
 - Dem Bereich von Informationen, die Datenbankaktivitäten vom Typ DML oder XQuery weiter kategorisieren:
 - Der Typ des anzugebenden Bereichs (entweder Timeron oder Kardinalität). Die Angabe eines Wertebereichs ist optional. Wenn Sie z. B. einen Bereich für eine Arbeitsklasse angeben, können Sie angeben, dass alle Abfragen mit einem geschätzten Aufwand von maximal 100 Timerons anders als die anderen Abfragen verarbeitet werden sollen.

- Die untere Bereichsgrenze.
- Die obere Bereichsgrenze.
- Dem aufzurufenden Routinenschema.

Beim Definieren einer Arbeitsklasse können Sie das Schemaattribut verwenden, um CALL-Anweisungen basierend auf dem Schema der aufzurufenden Prozedur weiter zu klassifizieren. Wenn Sie z. B. SCHEMA1 als Schema einer Arbeitsklasse angeben und der Arbeitstyp CALL ist, werden alle CALL-Anweisungen, die eine SCHEMA1-Prozedur aufrufen, in dieser Arbeitsklasse klassifiziert. Wenn Sie das Schema für einen anderen Arbeitsklassentyp als CALL oder ALL angeben, wird der Fehler SQL0628N zurückgegeben.

- Die Auswertungsreihenfolge der Arbeitsklasse (oder Position der Arbeitsklasse im Arbeitsklassenset).

Weitere Informationen finden Sie in „Auswertungsreihenfolge von Arbeitsklassen in einem Arbeitsklassenset“ auf Seite 93.

- Eine automatisch generierte Klassen-ID, die die Arbeitsklasse eindeutig bezeichnet.

Sie können Arbeitsklassen auf zwei Arten erstellen:

- Erstellen Sie mit dem Schlüsselwort `WORK CLASS` der Anweisung `CREATE WORK CLASS SET` ein neues Arbeitsklassenset, das die neue Arbeitsklasse enthalten soll.
- Fügen Sie mit dem Schlüsselwort `ADD` der Anweisung `ALTER WORK CLASS SET` die neue Arbeitsklasse einem vorhandenen Arbeitsklassenset hinzu.

Sie können Arbeitsklassen mit dem Schlüsselwort `ALTER WORK CLASS` der Anweisung `ALTER WORK CLASS SET` ändern.

Sie können Arbeitsklassen aus einem Arbeitsklassenset mit dem Schlüsselwort `DROP WORK CLASS` der Anweisung `ALTER WORK CLASS SET` löschen oder mit der Anweisung `DROP WORK CLASS` das Arbeitsklassenset selbst löschen.

Sie können Ihre Arbeitsklassen durch Abfrage der Sicht `SYSCAT.WORKCLASSES` anzeigen.

Mit Arbeitsklassensets können Sie eine oder mehrere Arbeitsklassen gruppieren. Ein Arbeitsklassenset hat die folgenden Attribute:

- Einen eindeutigen beschreibenden Namen für das Arbeitsklassenset.
- Kommentare, die Sie für das Arbeitsklassenset angeben wollen.
- Null oder mehr Arbeitsklassen (obwohl eine Arbeitsklasse nur in einem Arbeitsklassenset vorhanden sein kann, muss ein Arbeitsklassenset keine Arbeitsklassen enthalten).
- Eine automatisch generierte ID, die die Arbeitsklasse eindeutig bezeichnet.

Sie können ein neues Arbeitsklassenset mit der Anweisung `CREATE WORK CLASS SET` erstellen. Sie können ein leeres Arbeitsklassenset erstellen und Arbeitsklassen später hinzufügen, oder Sie können ein Arbeitsklassenset erstellen, das bereits mindestens eine Arbeitsklasse enthält.

Sie können ein vorhandenes Arbeitsklassenset mit der Anweisung `ALTER WORK CLASS SET` auf die folgenden Arten ändern:

- Fügen Sie Arbeitsklassen zum Arbeitsklassenset hinzu.
- Ändern Sie Arbeitsklassenattribute für Arbeitsklassen im Arbeitsklassenset.

- Löschen Sie Arbeitsklassen aus dem Arbeitsklassenset.

Sie können keine Attribute des Arbeitsklassensets ändern.

Mit der Anweisung `DROP WORK CLASS SET` können Sie ein Arbeitsklassenset löschen.

Sie können Ihre Arbeitsklassensets durch Abfrage der Katalogsicht `SYSCAT.WORK-CLASSETS` anzeigen.

Die folgende Abbildung zeigt ein Beispiel von Arbeitsklassen in einem Arbeitsklassenset.

Arbeitsklassenset: Large activities

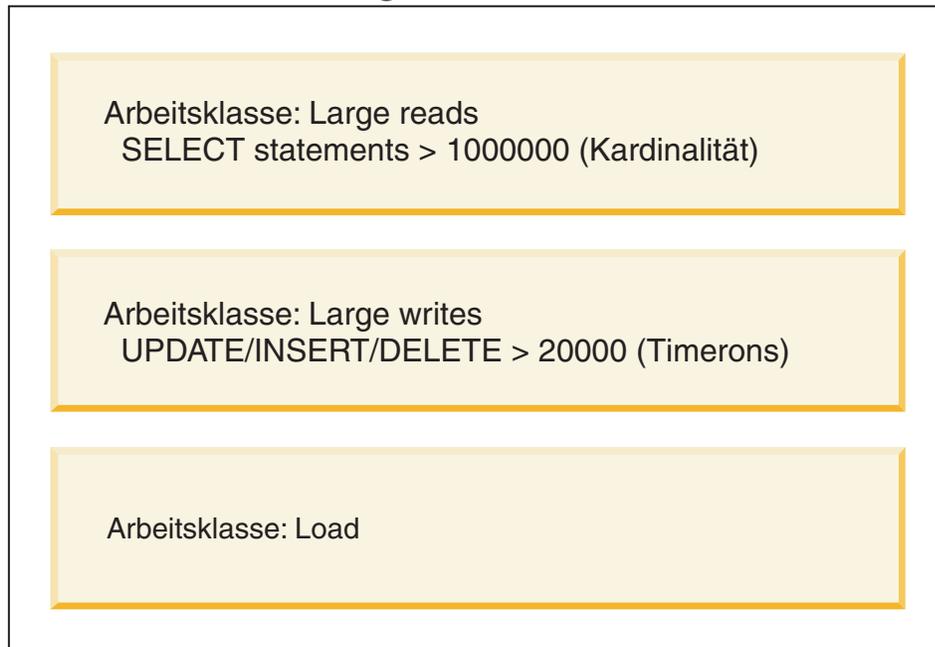


Abbildung 14. Beispiel von Arbeitsklassen in einem Arbeitsklassenset

Damit ein Arbeitsklassenset im System wirksam wird, müssen Sie ein Arbeitsaktionsset definieren und es mit dem Arbeitsklassenset verknüpfen. Durch Verwendung eines Arbeitsaktionssets können Sie ein Arbeitsklassenset entweder einer Servicesuperklasse oder der Datenbank zuordnen, um anzugeben, welche Aktion auf die Datenbankaktivitäten angewendet werden soll, die unter die Klassifizierung fallen. Wenn Sie kein Arbeitsaktionsset für das Arbeitsklassenset erstellen, ignoriert der Datenserver das Arbeitsklassenset.

Arbeitsaktionen und Arbeitsaktionssets

Eine Arbeitsaktion, die in Verbindung mit einer Arbeitsklasse verwendet wird, kann zur Steuerung bestimmter Aktivitätstypen eingesetzt werden. Sie können beispielsweise unterschiedliche Arbeitsaktionen auf `LOAD`-Aktivitäten anwenden, so dass sie anders als `DML` verarbeitet werden. Arbeitsaktionen werden in Arbeitsaktionssets gruppiert.

Eine Arbeitsaktion hat die folgenden Attribute:

- Einen benutzerdefinierten Arbeitsaktionsnamen, der im Arbeitsaktionsset eindeutig sein muss.
- Die Arbeitsklassen-ID, auf die die Arbeitsaktion angewendet wird.

Wenn auf eine Arbeitsklasse keine Arbeitsaktionen angewendet werden, verhält sich der Datenserver, als gäbe es die Arbeitsklasse nicht. Sie können für eine Arbeitsklasse mehrere Arbeitsaktionen definieren, aber jede Arbeitsaktion muss in dieser Arbeitsklasse eine andere Aktion ausführen.

- Die Aktion, die auf die Datenbankaktivität angewendet wird, die mit der Arbeitsklasse übereinstimmt. Welcher Aktionstyp für eine Arbeitsaktion gültig ist, hängt davon ab, ob das Arbeitsaktionsset, zu dem die Arbeitsaktion gehört, auf eine Datenbank oder eine Servicesuperklasse angewendet wird.

Wenn ein Arbeitsaktionsset auf eine Datenbank angewendet wird (in Abhängigkeit der Arbeitsklassen, mit denen die Arbeitsaktionen verknüpft sind), bezieht sich das Arbeitsaktionsset auf einige oder alle Aktivitäten, die in der Datenbank eingehen. Wenn ein Arbeitsaktionsset auf eine Servicesuperklasse angewendet wird (in Abhängigkeit der Arbeitsklassen, mit denen die Arbeitsaktionen verknüpft sind), bezieht sich dieses Arbeitsaktionsset auf einige oder alle Aktivitäten, die unter dieser Servicesuperklasse ausgeführt werden. Zum Beispiel:

- Ein Arbeitsaktionsset, das auf eine Datenbank angewendet wird, kann Schwellenwertarbeitsaktionen enthalten. Wenn eine Aktivität einer Arbeitsklasse zugeordnet wird, für die eine Schwellenwertarbeitsaktion definiert ist, wird der Schwellenwert auf diese Aktivität angewendet.
- Ein Arbeitsaktionsset, das auf eine Servicesuperklasse angewendet wird, kann eine Arbeitsaktion enthalten, die die Aktivität einer Serviceunterklasse in der Servicesuperklasse zuordnet. Wenn eine Aktivität einer bestimmten Arbeitsklasse im Arbeitsklassenset entspricht und das Arbeitsaktionsset eine Zuordnungsarbeitsaktion hat, die für diese Arbeitsklasse definiert ist, wird diese Aktivität der von der Arbeitsaktion definierten Serviceunterklasse zugeordnet.

Eine Liste der unterstützten Aktionen finden Sie in „Arbeitsaktionen und die Domäne des Arbeitsaktionssets“ auf Seite 94.

- Ein Objekt, das das Ziel der angegebenen Aktion ist.

Je nach Aktion kann das Objekt eine Serviceunterklasse sein, der die Aktivität zugeordnet wird, ein Schwellenwert, der angibt, welcher Grenzwert auf die Aktivität anzuwenden ist (bzw. Null, wenn die Aktion die Ausführung verhindern soll), eine der Erfassungsaktionen oder eine Zählaktivität.

- Die Schablone, die das Histogramm mit den erfassten Statistikdaten zur Dauer (in Mikrosekunden) der Aktivitäten beschreibt, die mit der dieser Arbeitsaktion zugeordneten Arbeitsklasse verknüpft sind und in einem bestimmten Intervall ausgeführt werden müssen.

Diese Informationen werden nur erfasst, wenn der Arbeitsaktionstyp COLLECT AGGREGATE ACTIVITY DATA (entweder BASE oder EXTENDED) ist. Weitere Informationen zu Histogrammen und Histogrammschablonen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.

- Eine Angabe, ob die Arbeitsaktion aktiviert ist.
- Eine automatisch generierte ID für die Arbeitsaktion.

Sie können eine Arbeitsaktion entweder mit dem Schlüsselwort WORK ACTION in der Anweisung CREATE WORK ACTION SET oder mit dem Schlüsselwort ADD in der Anweisung ALTER WORK ACTION SET erstellen. Mit dem Schlüsselwort ALTER in der Anweisung ALTER WORK ACTION SET können Sie eine Arbeitsaktion ändern. Mit dem Schlüsselwort DROP in der Anweisung ALTER WORK ACTION SET können Sie eine Arbeitsaktion aus einem Arbeitsaktionsset entfernen.

Sie können Ihre Arbeitsaktionen durch Abfrage der Sicht SYSCAT.WORKACTIONS anzeigen.

Ein Arbeitsaktionsset hat die folgenden Attribute:

- Einen Namen für das Arbeitsaktionsset, der in der Datenbank eindeutig ist.
- Den Namen des Arbeitsklassensets, das die Arbeitsklasse enthält, auf die die Gruppe von Aktionen angewendet wird.

Da die Definitionen der Arbeitsklassensets getrennt sind von den für sie definierten Arbeitsaktionssets, können Sie mehrere Arbeitsaktionssets für ein Arbeitsklassenset definieren.

- Das Objekt, dem das Arbeitsaktionsset zugeordnet ist (Datenbank oder Service-superklasse).
- Den Namen der Servicesuperklasse, auf die die Aktionen und das Arbeitsklassenset angewendet werden (für Arbeitsaktionssets, die einer Servicesuperklasse zugeordnet sind).
- Eine Angabe, ob das Arbeitsaktionsset aktiviert ist.
- Benutzerkommentare.
- Mindestens eine Arbeitsaktion (ein Arbeitsaktionsset muss keine Arbeitsaktionen enthalten).
- Eine automatisch generierte ID, die das Arbeitsaktionsset eindeutig bezeichnet.

Sie können ein Arbeitsaktionsset mit der Anweisung CREATE WORK ACTION SET erstellen, mit der Anweisung ALTER WORK ACTION SET ändern und mit der Anweisung DROP WORK ACTION SET löschen.

Sie können Ihre Arbeitsaktionssets durch Abfrage der Sicht SYSCAT.WORKACTIONSETS anzeigen.

Wenn Sie ein Arbeitsaktionsset erstellen, müssen Sie das Objekt angeben, auf das das Arbeitsaktionsset angewendet werden soll. Gültige Objekttypen sind die Datenbank oder eine Servicesuperklasse. Sie müssen auch angeben, mit welchem Arbeitsklassenset das Arbeitsaktionsset arbeiten soll. Dies erlaubt es Ihnen, die Arbeitsklassen im Arbeitsklassenset zu verwenden, um die Typen von Aktivitäten anzugeben, auf die Sie die Arbeitsaktion anwenden wollen.

Wenn Sie eine Workload so einrichten, dass ihre Datenbankaktivitäten direkt einer Serviceunterklasse zugeordnet werden, wird das Arbeitsaktionsset, das mit dieser Servicesuperklasse verknüpft ist, nie für die Aktivitäten verwendet, die von dieser Workload abgesetzt werden. Wenn eine Workload also Aktivitäten direkt einer Serviceunterklasse zuordnet, wird das Arbeitsaktionsset umgangen. Keine der Arbeitsaktionen im Arbeitsaktionsset werden auf die Aktivitäten angewendet, die direkt der Serviceunterklasse zugeordnet sind.

Beziehung zwischen Arbeitsklassen, Arbeitsklassensets, Arbeitsaktionen und Arbeitsaktionssets und deren Zuordnung zu anderen DB2-Objekten

Arbeitsklassen und Arbeitsaktionen arbeiten zusammen, um bestimmte Aktionen bestimmten Aktivitätstypen zuzuordnen. Dies wird am besten anhand eines Beispiels erläutert.

Das folgende Diagramm zeigt eine abstrakte Sicht der Beziehung zwischen Arbeitsklassen, Arbeitsklassensets, Arbeitsaktionen und Arbeitsaktionssets und deren Zuordnung zu anderen DB2-Objekten.

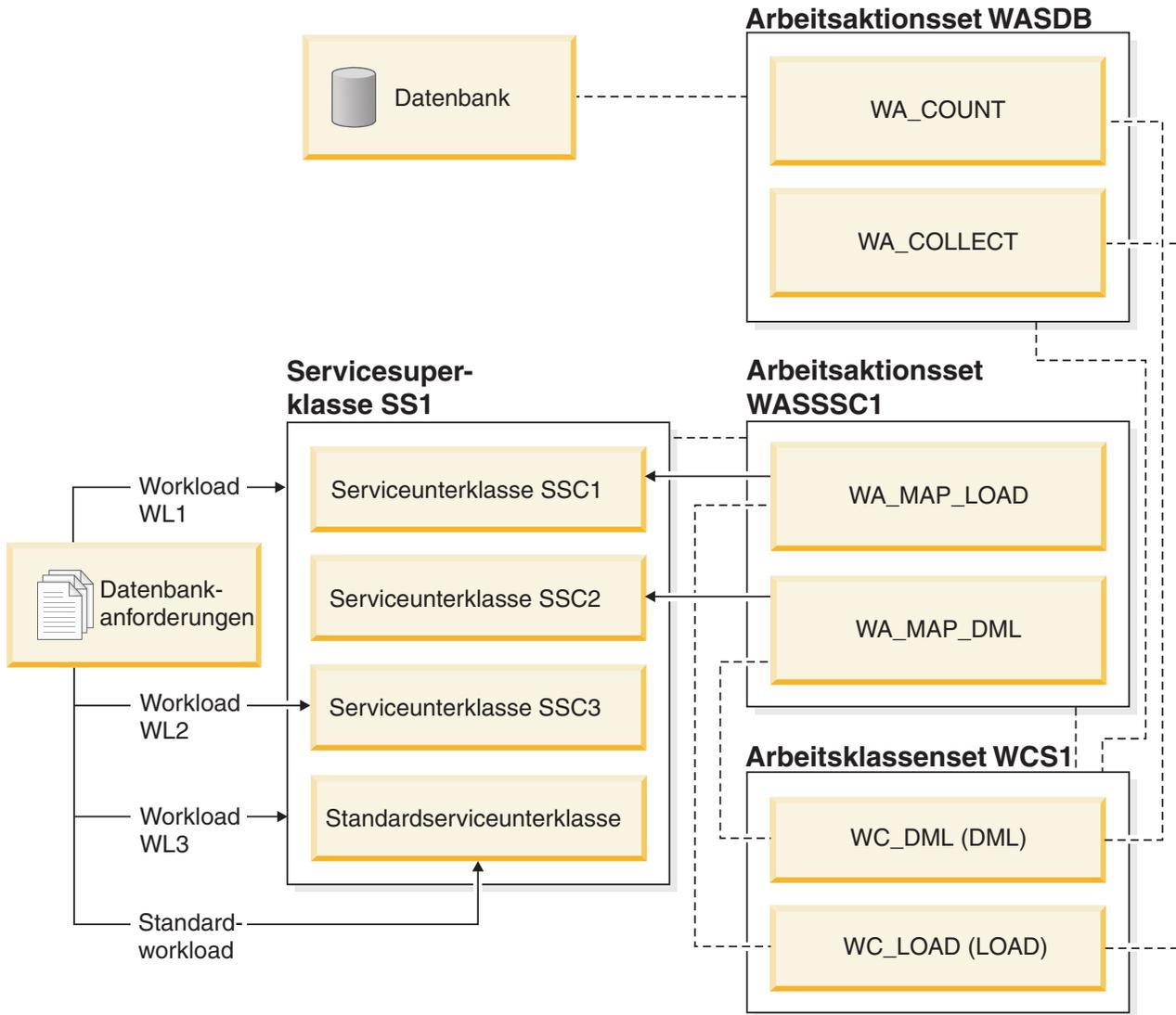


Abbildung 15. Überblick über Arbeitsaktionssets und Arbeitsklassensets

Im Diagramm werden einige Datenbankaktivitäten über die Workload WL1, die Workload WL3 und die Standardbenutzerworkload (SYSDEFAULTUSERWORKLOAD) der Servicesuperklasse SS1 zugeordnet. Da das Arbeitsaktionsset WASDB auf die Datenbank angewendet wird, werden auf alle Aktivitäten, die der Standardbenutzerworkload, der Workload WL1 oder der Workload WL3 zugeordnet sind und unter die Arbeitsklassen WC_DML oder WC_LOAD fallen, die Arbeitsaktionen im Arbeitsaktionsset WASDB angewendet.

Das heißt, Aktivitäten mit dem Arbeitstyp DML werden gezählt, und für Aktivitäten mit dem Arbeitstyp LOAD werden Aktivitätsdaten erfasst und in einen aktiven Ereignismonitor (sofern vorhanden) geschrieben.

Das Arbeitsaktionsset WASSC1 wird auf die Servicesuperklasse SS1 angewendet. Auf alle Aktivitäten, die der Standardbenutzerworkload, der Workload WL1 oder der Workload WL3 zugeordnet sind und unter die Arbeitsklasse WC_DML oder WC_LOAD fallen, werden auch die Arbeitsaktionen WA_MAP_DML und WA_MAP_LOAD angewendet. Das heißt, Aktivitäten mit dem Arbeitstyp LOAD werden von der Arbeitsaktion WA_MAP_LOAD der Serviceunterklasse SSC1 zugeordnet, und Aktivitäten mit dem Arbeitstyp DML werden von der Arbeitsaktion WA_MAP_DML der Serviceunterklasse SSC2 zugeordnet.

Aktivitäten, die entweder der Workload WL2 oder der Standardworkload zugeordnet werden (da ihre Verbindungsattribute keiner definierten Workload entsprechen), werden direkt Serviceunterklassen zugeordnet. Insbesondere ordnet die Workload WL2 ihre Aktivitäten der Serviceunterklasse SSC3 zu. Wenn eine Workload Aktivitäten direkt einer Serviceunterklasse zuordnet, werden keine Arbeitsaktionen auf diese Aktivitäten angewendet.

Arbeitstypen und SQL-Anweisungen von Arbeitsklassen

Eine Arbeitsklasse hat einen zugeordneten Arbeitstyp. Mindestens ein Aktivitätstyp kann unter einen Arbeitsklassentyp fallen. Bevor Sie eine Arbeitsklasse definieren, sollten Sie verstehen, welche Aktivitätstypen unter jeden Typ von Arbeitsklasse fallen.

Die folgende Tabelle zeigt die Typschlüsselwörter, die für Arbeitsklassen verfügbar sind, sowie die SQL-Anweisungen, die den verschiedenen Schlüsselwörtern entsprechen. Mit Ausnahme des Befehls load werden alle in der folgenden Tabelle aufgeführten Anweisungen unmittelbar vor der Ausführung bei der Verarbeitung einer Anforderung EXECUTE, EXECUTE IMMEDIATE oder OPEN abgefangen. Wenn das Dienstprogramm LOAD von einem Client abgesetzt wird, setzt es eventuell Anforderungen vor dem Start der eigenen Ladeoperation auf dem Daten-server ab.

Tabelle 13. Arbeitstypschlüsselwörter und zugehörige SQL-Anweisungen

Arbeitstypschlüsselwort	Gültige SQL-Anweisungen
READ	<ul style="list-style-type: none"> • Alle SELECT-Anweisungen (SELECT INTO, VALUES INTO, FULL SELECT) Anmerkung: SELECT-Anweisungen mit DELETE, INSERT oder UPDATE sind nicht eingeschlossen. • Alle XQuery-Anweisungen
WRITE	<ul style="list-style-type: none"> • Alle UPDATE-Anweisungen (gesucht, positioniert) • Alle DELETE-Anweisungen (gesucht, positioniert) • Alle INSERT-Anweisungen (values, Subselect) • Alle MERGE-Anweisungen • Alle SELECT-Anweisungen mit einer Anweisung DELETE, INSERT oder UPDATE

Tabelle 13. Arbeitstypschlüsselwörter und zugehörige SQL-Anweisungen (Forts.)

Arbeitstypschlüsselwort	Gültige SQL-Anweisungen
CALL	Anweisung CALL. Anmerkung: Die Anweisung CALL wird nur unter den Arbeitsklassentypen CALL und ALL klassifiziert.
DML	Alle Anweisungen, die unter den Arbeitsklassentypen READ und WRITE klassifiziert werden
DDL	<ul style="list-style-type: none"> • Alle ALTER-Anweisungen • Alle CREATE-Anweisungen • Anweisung COMMENT • Anweisung DECLARE GLOBAL TEMPORARY TABLE • Anweisung DROP • Anweisung FLUSH PACKAGE CACHE • Alle GRANT-Anweisungen • REFRESH TABLE • Alle RENAME-Anweisungen • Alle REVOKE-Anweisungen • Anweisung SET INTEGRITY
LOAD	Dienstprogramm LOAD Anmerkung: Das Dienstprogramm LOAD wird nur unter den Arbeitsklassentypen LOAD und ALL klassifiziert.
ALL	<p>Alle Datenbankaktivitäten</p> <p>Anmerkung: Wenn die Aktion ein Schwellenwert ist, hängt die Datenbankaktivität, auf die der Schwellenwert angewendet wird, vom Typ des Schwellenwerts ab. Wenn der Schwellenwerttyp z. B. ESTIMATEDSQLCOST ist, sind nur DML-Aktivitäten mit einem geschätzten Aufwand (in Timerons) vom Schwellenwert betroffen.</p> <p>Weitere Informationen finden Sie in „Beispiel: Verwenden einer Arbeitsklasse mit dem Schlüsselwort ALL“ auf Seite 172.</p>

Die folgende Abbildung zeigt eine hierarchische Ansicht der Arbeitstypschlüsselwörter:

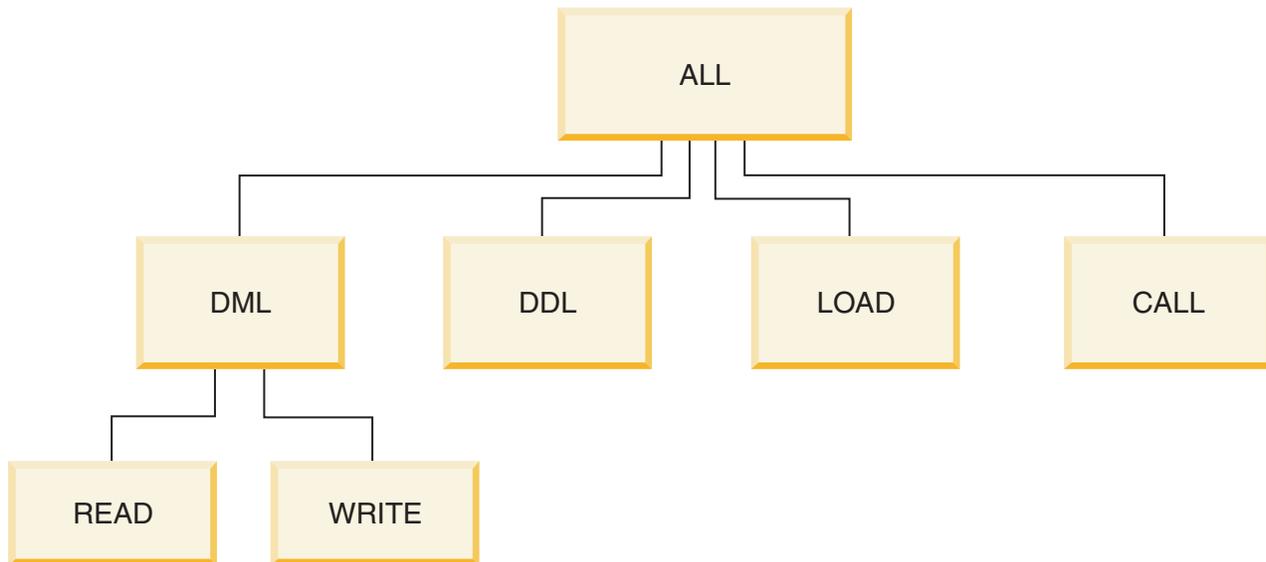


Abbildung 16. Arbeitstypschlüsselwörter

SQL-Anweisungen, die unter keines der verfügbaren Schlüsselwörter fallen, werden nicht klassifiziert und verhalten sich so, als wäre keine Arbeitsklasse und kein Arbeitsklassenset vorhanden. Wenn die Anweisung z. B. SET SCHEMA lautet und die einzige Arbeitsklasse im Arbeitsklassenset den Arbeitstyp DML hat, wird diese Anweisung nicht klassifiziert, und keine Arbeitsaktion kann auf sie angewendet werden. Wenn die Aktion MAP ist, wird die Aktivität SET SCHEMA also in der Standardserviceunterklasse (SYSDEFAULTSUBCLASS) ausgeführt. Wenn die Aktion ein Schwellenwert ist, wird kein Schwellenwert auf die Aktivität angewendet.

Auswertungsreihenfolge von Arbeitsklassen in einem Arbeitsklassenset

Ein Arbeitsklassenset kann mehrere Arbeitsklassen enthalten, die mit einer Datenbankaktivität übereinstimmen. Zur Auswahl, unter welche Arbeitsklasse aus einem Arbeitsklassenset eine Aktivität fallen soll, durchläuft der Datenserver die Arbeitsklassen entsprechend der Auswertungsreihenfolge und stoppt an der ersten Arbeitsklasse, die mit der Aktivität übereinstimmt.

Wenn keine übereinstimmende Arbeitsklasse vorhanden ist, gehört die Datenbankaktivität zu keiner Arbeitsklasse, und es wird keine Arbeitsaktion auf diese Aktivität angewendet. Nur Arbeitsklassen, auf die Arbeitsaktionen angewendet werden, werden berücksichtigt.

Sie können die Auswertungsreihenfolge von Arbeitsklassen in einem Arbeitsklassenset beim Erstellen oder Ändern eines Arbeitsklassensets beeinflussen. Wenn Sie ein Arbeitsklassenset erstellen oder ändern, können Sie mit einer der drei folgenden Methoden die Position festlegen, an der eine Arbeitsklasse im Arbeitsklassenset platziert wird:

- Geben Sie die absolute Position der Arbeitsklasse in der Liste an.
Zum Beispiel: POSITION AT 2. In dieser Situation wird die Arbeitsklasse an der zweiten Position im Arbeitsklassenset platziert. Die bisher an Position 2 befindliche Arbeitsklasse steht jetzt an dritter Stelle, die bisher 3. Arbeitsklasse steht jetzt an vierter Stelle usw. Wenn die in der Anweisung CREATE WORK CLASS SET oder ALTER WORK CLASS SET angegebene Position für die Arbeitsklasse größer ist als die Gesamtzahl von Arbeitsklassen im Arbeitsklassenset, wird die Arbeitsklasse an letzter Stelle positioniert.
- Verwenden Sie das Schlüsselwort POSITION BEFORE oder POSITION AFTER, um die Position der Arbeitsklasse relativ zu anderen bereits im Arbeitsklassenset enthaltenen Arbeitsklassen anzugeben.
- Geben Sie beim Erstellen einer Arbeitsklasse keine Position an.
In dieser Situation wird die neue Arbeitsklasse am Ende der Liste platziert. Die für die Arbeitsklasse angegebene Position in der Liste der Arbeitsklassensets ist nicht notwendigerweise der tatsächliche Wert der Spalte EVALUATIONORDER in der Sicht SYSCAT.WORKCLASSES. Der Datenserver weist automatisch den Folgewert zu, um Lücken zu vermeiden.

Arbeitsklassen werden in der Reihenfolge ihres Empfangs verarbeitet, was sich auf die Auswertungsreihenfolge auswirken kann. Nehmen wir z. B. an, dass Sie die folgende Anweisung absetzen:

```
ALTER WORK CLASS SET WCS ALTER WORK CLASS C1 POSITION AT 1 ALTER WORK CLASS C2
POSITION AT 1
```

Als Ergebnis hat die Arbeitsklasse C1 eine Auswertungsreihenfolge von 2, und die Arbeitsklasse C2 hat eine Auswertungsreihenfolge von 1, da C2 die zuletzt verarbeitete Arbeitsklasse war.

Arbeitsaktionen und die Domäne des Arbeitsaktionssets

Sie können ein Arbeitsaktionsset entweder für eine Datenbank oder für eine Servicesuperklasse definieren. Der Typ von Arbeitsaktionen, der für ein Arbeitsaktionsset definiert werden kann, hängt vom Typ des Objekts ab, für das das Arbeitsaktionsset definiert ist.

Wenn das Arbeitsaktionsset für eine Datenbank definiert ist, müssen die Arbeitsaktionen im Arbeitsaktionsset einen der folgenden Typen haben:

- Schwellenwert
Der tatsächliche Schwellenwert wird mit dem Schlüsselwort WHEN schwellenwerttyp angegeben. Mehrere Schwellenwertarbeitsaktionen können auf eine einzelne Arbeitsklasse angewendet werden, wenn alle Schwellenwerte unterschiedliche Typen haben. Wenn diese Aktion angegeben wird, wird der Schwellenwert auf alle Datenbankaktivitäten angewendet, die mit der Arbeitsklasse verknüpft sind.
- PREVENT EXECUTION
Wenn diese Aktion angegeben wird, wird die Ausführung aller Datenbankaktivitäten verhindert, die mit der zugehörigen Arbeitsklasse übereinstimmen.

- COLLECT ACTIVITY DATA

Wenn diese Aktion angegeben wird, werden Informationen zu den Datenbankaktivitäten entsprechend der Arbeitsklasse, für die diese Arbeitsaktion definiert ist, in den aktiven ACTIVITIES-Ereignismonitor geschrieben, wenn die Ausführung der Aktivitäten abgeschlossen ist. Weitere Informationen finden Sie in „Erfassen von Daten für einzelne Aktivitäten“ auf Seite 143.

- COUNT ACTIVITY

Wenn diese Aktivität angegeben wird, erhöhen alle Datenbankaktivitäten, die der zugehörigen Arbeitsklasse zugeordnet sind, den Turnstile-Zähler (Drehkreuzzähler) für diesen Arbeitsklassentyp. (Der Turnstile-Zähler für die Arbeitsklasse wird jedesmal um 1 erhöht, wenn eine Aktivität mit dieser Arbeitsklasse verknüpft wird.) Die Arbeitsaktion COUNT ACTIVITY ist eine Möglichkeit, mit der effektiv sichergestellt werden kann, dass dieser Zähler aktualisiert wird. Wenn keine Arbeitsaktion auf eine Aktivität zu einer Arbeitsklasse angewendet wird, wird der Arbeitsklassenaktivitätenzähler nicht erhöht. Manchmal geht es nur darum, die Anzahl der Aktivitäten eines bestimmten Typs zu ermitteln. Weitere Informationen finden Sie in „Erfassen von Daten für einzelne Aktivitäten“ auf Seite 143.

Wenn die Arbeitsaktionen im Arbeitsaktionsset mit keiner dieser Aktionen übereinstimmen, wird SQL4720N zurückgegeben.

Wenn Sie ein Arbeitsaktionsset für eine Servicesuperklasse definieren, muss es sich bei den Arbeitsaktionen im Arbeitsaktionsset um eine der folgenden Aktionen handeln:

- Eine Zuordnungsaktion

Sie können eine Aktivität einer beliebigen Serviceunterklasse in der Servicesuperklasse mit Ausnahme der Standardserviceunterklasse zuordnen. Sie geben die Serviceunterklasse, der die Aktivität zugeordnet werden soll, mit dem Schlüsselwort MAP ACTIVITY TO SERVICE CLASS an. Nur eine Zuordnungsarbeitsaktion im Arbeitsaktionsset kann auf dieselbe Arbeitsklasse angewendet werden.

- PREVENT EXECUTION

Das Verhalten ist mit dem für die Datenbankarbeitsaktion identisch.

- COLLECT ACTIVITY DATA

Das Verhalten ist mit dem für die Datenbankarbeitsaktion identisch.

- COLLECT AGGREGATE ACTIVITY DATA

Wenn diese Aktion angegeben wird, werden zusammengefasste Datenbankaktivitätsdaten entsprechend der Arbeitsklasse erfasst, für die diese Arbeitsaktion definiert ist.

- COUNT ACTIVITY

Das Verhalten ist mit dem für die Datenbankarbeitsaktion identisch.

Wenn die Arbeitsaktionen im Arbeitsaktionsset mit keiner dieser Aktionen übereinstimmen, wird SQL4720N zurückgegeben.

Die folgende Abbildung zeigt ein Beispiel, wie die Arbeitsklassen in einem Arbeitsklassenset mit dem Namen LARGE ACTIVITIES sowohl auf die Datenbank als auch auf eine Servicesuperklasse angewendet werden. Zur Erreichung dieses Ziels werden die beiden Arbeitsaktionssets Database large activities und Service class large activities erstellt.

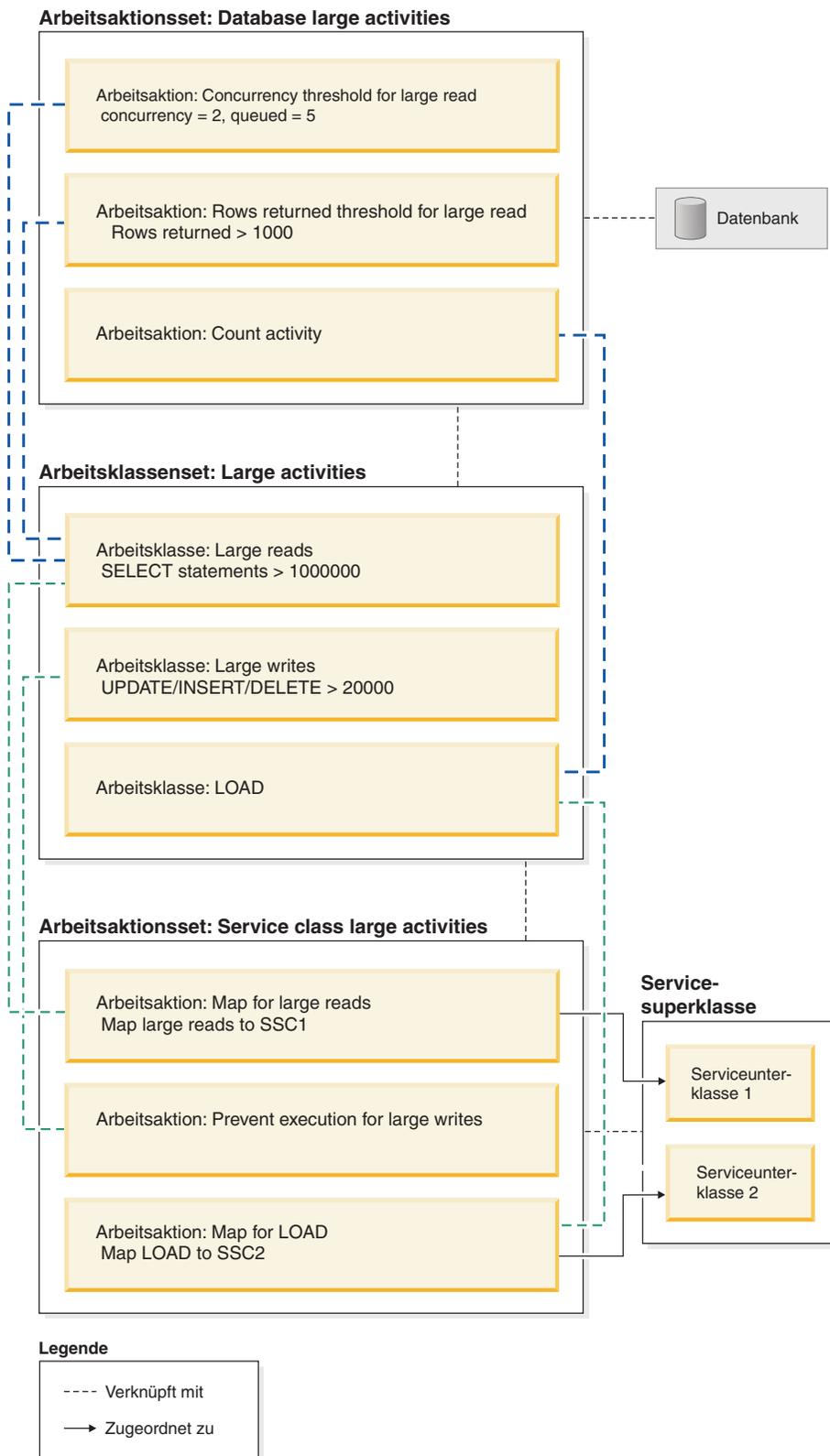


Abbildung 17. Beispiel von Arbeitsaktionen, Arbeitsaktionssets, Arbeitsklassen und Arbeitsklassensets

- Die Arbeitsaktionssets sind wie folgt aufgebaut:
- Database large activities enthält:

- Concurrency threshold for large reads. Diese Aktion erlaubt, dass zwei große Lesevorgänge gleichzeitig ausgeführt und fünf große Lesevorgänge in die Warteschlange gestellt werden können.
- Rows returned threshold for large reads. Diese Aktion verhindert, dass große Lesevorgänge mehr als 1000 Zeilen zurückgeben.
- Count activity for load. Diese Aktion zählt, wie oft das Dienstprogramm LOAD in der Datenbank ausgeführt wird.
- Service class large activities enthält:
 - Map for large reads. Diese Aktion ordnet große Lesevorgänge der Serviceunterklasse 1 zu.
 - Map for large writes. Diese Aktion verhindert, dass große Schreibvorgänge ausgeführt werden.
 - Map for LOAD. Diese Aktion ordnet Ladevorgänge der Serviceunterklasse 2 zu.

Ein Arbeitsaktionsset muss nicht für jede Arbeitsklasse im Arbeitsklassenset, auf das das Arbeitsaktionsset angewendet wird, eine Aktion enthalten. Zudem können auf eine Arbeitsklasse mehrere Arbeitsaktionen angewendet werden, sofern die Aktionstypen unterschiedlich sind. Auf eine Arbeitsklasse können mehrere Arbeitsaktionen angewendet werden, sofern die Schwellenwerttypen unterschiedlich sind.

In Arbeitsaktionen verwendbare Schwellenwerte

Für Datenbanken definierte Arbeitsaktionssets können Arbeitsaktionen enthalten, die Schwellenwerte angeben.

Die folgenden Schwellenwerte werden unterstützt:

- Zusammengefasster Schwellenwert:
 - CONCURRENTDBCOORDACTIVITIES
- Aktivitätsschwellenwerte:
 - SQLTEMPSPACE
 - SQLROWSRETURNED
 - ACTIVITYTOTALTIME
 - ESTIMATEDSQLCOST

Von Schwellenwerten unterstützte Arbeitsklassifikationen

Obwohl jeder der Schwellenwerttypen, die in Arbeitsaktionen verwendet werden können, mit einer beliebigen Arbeitsklasse verknüpft werden kann, werden nicht alle Typen von Datenbankaktivitäten für alle diese Schwellenwerttypen unterstützt.

Wenn Sie z. B. eine Arbeitsklasse für DDL erstellen und dann diese Arbeitsklasse mit einer Schwellenwertarbeitsaktion ESTIMATEDSQLCOST verknüpfen, wird dieser Schwellenwert auf keine der Anfragen angewendet, die unter DDL klassifiziert sind, da DDL-Anweisungen keinen geschätzten Aufwand haben. Wenn Sie eine Arbeitsklasse für ALL erstellen und dann diese Arbeitsklasse mit einer Schwellenwertarbeitsaktion ESTIMATEDSQLCOST verknüpfen, wird der Schwellenwert nur auf die Datenbankaktivitäten mit geschätztem Aufwand angewendet, obwohl alle Datenbankaktivitäten zur Arbeitsklasse ALL gehören.

Die folgende Tabelle zeigt, welche Arbeitsklassenkategorien von welchen Schwellenwerttypen unterstützt werden:

Tabelle 14. Von Schwellenwerten unterstützte Arbeitsklassifikationen

	„Schwellenwert CONCURRENTDBCOORDACTIVITIES“ auf Seite 73	„Schwellenwert SQLTEMPSPACE“ auf Seite 65	„Schwellenwert SQLROWSRETURNED“ auf Seite 66	„Schwellenwert ESTIMATEDSQLCOST“ auf Seite 64	„Schwellenwert ACTIVITYTOTALTIME“ auf Seite 67
READ	Ja	Ja	Ja	Ja	Ja
WRITE	Ja	Ja	Ja	Ja	Ja
CALL	Ja	Nein	Nein (siehe Anmerkung)	Nein	Ja
DML	Ja	Ja	Ja	Ja	Ja
DDL	Ja	Nein	Nein	Nein	Ja
LOAD	Ja	Nein	Nein	Nein	Ja
ALL	Ja	Einige	Einige	Einige	Ja

Anmerkung: Obwohl die Anweisungen in der aufgerufenen Prozedur eventuell Zeilen zurückgeben, werden diese nicht vom Schwellenwert SQLROWSRETURNED kontrolliert, da die Zeilen nicht als Ergebnis einer Anweisung CALL zurückgegeben werden.

Zuordnen von Aktivitäten zu Arbeitsklassen

Wenn ein Arbeitsklassenset über ein Arbeitsaktionsset mit einer Datenbank oder mit einer Servicesuperklasse verknüpft wird und dies direkt vor der Ausführung in der Verarbeitung einer Ausführungs-, Sofortausführungs- oder Öffnungsanforderung oder direkt vor der Ausführung des Dienstprogramms LOAD geschieht, wird die Datenbankaktivität geprüft, um festzustellen, ob sie einer der Bedingungen entspricht, die in den Arbeitsklassen im Arbeitsklassenset angegeben sind.

Die Arbeitsklassen werden im Arbeitsklassenset nach ihrer Auswertungsreihenfolge sortiert. Basierend auf dieser Auswertungsreihenfolge wird die Datenbankaktivität gegen jede Arbeitsklasse auf der Grundlage der Attribute der Datenbankaktivität (wie Aktivitätstyp und Kardinalität) geprüft, bis eine Übereinstimmung gefunden wird oder bis die Liste der Arbeitsklassen im Arbeitsklassenset abgearbeitet wurde.

Nehmen wir an, dass die folgenden Arbeitsklassen in einem Arbeitsklassenset enthalten sind und dass auf alle Arbeitsklassen eine Arbeitsaktion angewendet wird:

- Auswertungsreihenfolge: 1; Arbeitsklassenname: MyLoad; Arbeitsklassentyp: LOAD
- Auswertungsreihenfolge: 2; Arbeitsklassenname: SmallRead; Arbeitsklassentyp: READ; weitere Attribute: geschätzter Aufwand < 300 Timerons
- Auswertungsreihenfolge: 3; Arbeitsklassenname: AllDML; Arbeitsklassentyp: DML
- Auswertungsreihenfolge: 4; Arbeitsklassenname: LargeRead; Arbeitsklassentyp: READ; weitere Attribute: geschätzter Aufwand > 301 Timerons
- Auswertungsreihenfolge: 5; Arbeitsklassenname: MyDDL; Arbeitsklassentyp: DDL

Wenn eine Anweisung SELECT mit einem geschätzten Aufwand von 200 Timerons empfangen wird, wird sie der Arbeitsklasse SmallRead zugeordnet. Wenn eine DDL-Aktivität (wie z. B. CREATE TABLE) empfangen wird, wird sie der Arbeitsklasse MyDDL zugeordnet. Wenn eine Anweisung SELECT mit einem geschätzten Aufwand von 500 Timerons empfangen wird, wird sie der Arbeitsklasse AllDML zugeordnet, da AllDML vor der Arbeitsklasse LargeRead positioniert ist. Weitere Informationen finden Sie in „Beispiel: Verwenden einer Arbeitsklasse mit dem Schlüsselwort ALL“ auf Seite 172.

Wenn auf eine Arbeitsklasse keine Arbeitsaktion angewendet ist, wird diese Arbeitsklasse ignoriert, und ihr werden keine Aktivitäten zugeordnet. Weitere Informationen finden Sie in „Auswertungsreihenfolge von Arbeitsklassen in einem Arbeitsklassenset“ auf Seite 93.

Anwenden von Arbeitsaktionen auf Datenbankaktivitäten

Nur ein einziges Arbeitsaktionsset kann auf eine Datenbank oder eine Service-superklasse angewendet werden.

Wenn Arbeit an den Datenserver übergeben wird, wird sie entweder mit einer benutzerdefinierten Workload oder mit der Standardworkload verknüpft und dann einer Serviceklasse zugeordnet.

Die folgende Abbildung zeigt, wie eine Arbeitsaktion auf eine Aktivität angewendet wird.

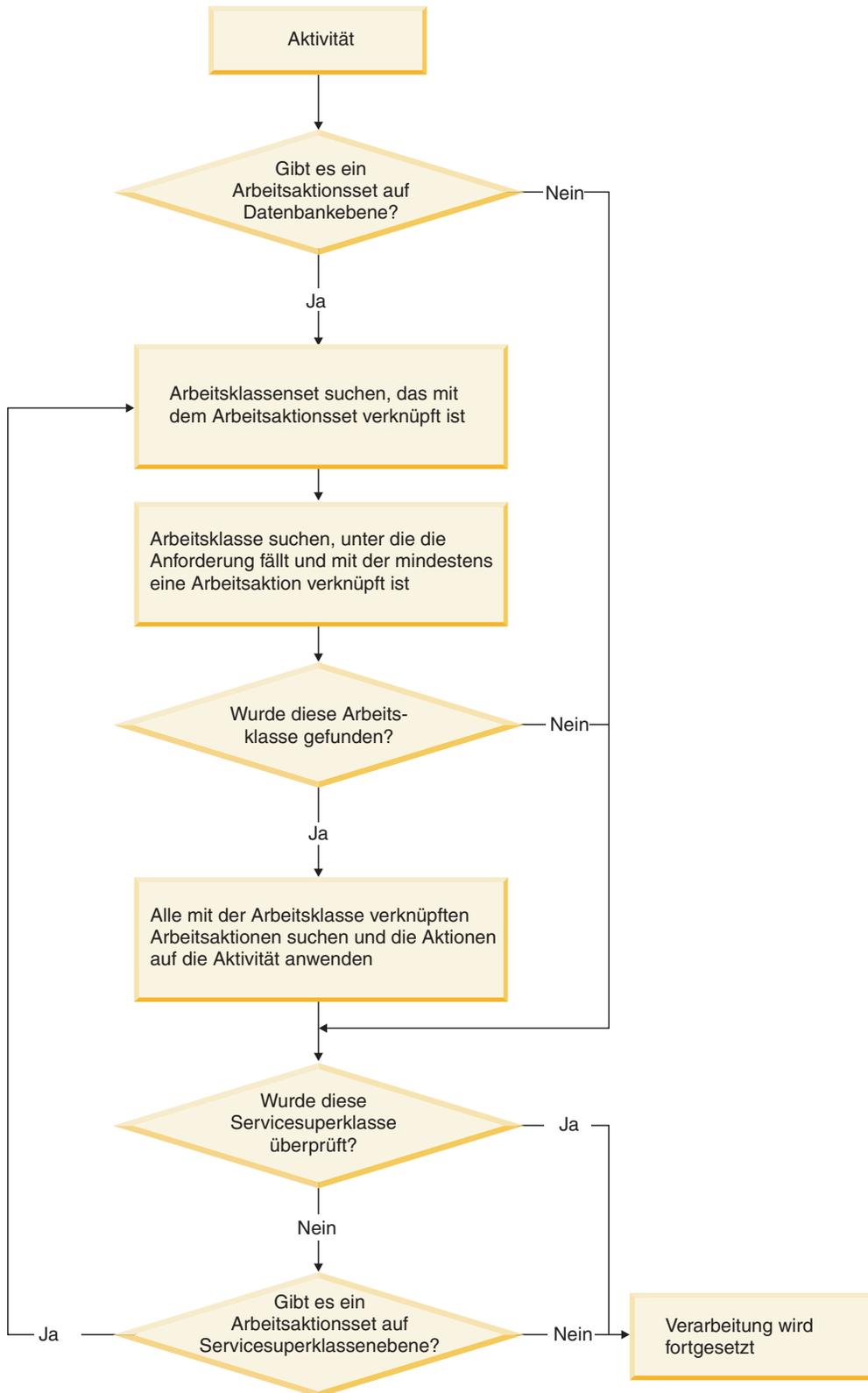


Abbildung 18. Anwenden einer Arbeitsaktion auf eine Aktivität

Eine Arbeitsaktion wird wie folgt einer Aktivität zugeordnet:

1. Wenn eine Aktivität einer Servicesuperklasse oder einer Serviceunterklasse zugeordnet wird, prüft der Datenserver, ob ein aktiviertes Arbeitsaktionsset auf Datenbankebene vorhanden ist.
2. Wenn ein aktiviertes Arbeitsaktionsset auf Datenbankebene vorhanden ist, prüft der Datenserver, ob die Aktivität unter eine der Arbeitsklassen im Arbeitsklassenset fällt, mit dem das Arbeitsaktionsset auf Datenbankebene verknüpft ist.
3. Wenn die Aktivität unter eine Arbeitsklasse fällt, auf die mindestens eine Arbeitsaktion angewendet ist, werden diese Arbeitsaktionen auf die Aktivität angewendet.
4. Wenn die Aktivität von der Workload einer Servicesuperklasse zugeordnet wird, prüft der Datenserver als nächstes, ob ein Arbeitsaktionsset auf die Servicesuperklasse angewendet ist.
5. Wenn ein Arbeitsaktionsset auf die Servicesuperklasse angewendet ist, prüft der Datenserver, ob die Aktivität unter eine der Arbeitsklassen im Arbeitsklassenset fällt, mit dem das Arbeitsaktionsset auf der Ebene der Servicesuperklasse verknüpft ist.
6. Wenn die Aktivität unter eine Arbeitsklasse fällt, auf die mindestens eine Arbeitsaktion angewendet ist, werden diese Arbeitsaktionen auf die Aktivität angewendet.

In den folgenden Situationen ist eine Aktivität nicht von einem Arbeitsaktionsset betroffen:

- Aktivitäten fallen in die Standardsystemserviceklasse (SYSDEFAULTSYSTEMCLASS) und in die Standardverwaltungsserviceklasse (SYSDEFAULTMAINTENANCECLASS).
- Aktivitäten werden der Standardverwaltungsworkload (SYSDEFAULTADMWORKLOAD) zugeordnet.
- Aktivitäten befinden sich in einer Ladeoperation. Die Ladeoperation selbst durchläuft nicht die Auswertung des Arbeitsaktionssets.
- Es handelt sich um untergeordnete Aktivitäten von gespeicherten Systemprozeduren. Die einzige Ausnahme ist die gespeicherte Prozedur SYSPROC.ADMIN_CMD. Untergeordnete Aktivitäten von SYSPROC.ADMIN_CMD durchlaufen die Auswertung des Arbeitsaktionssets.
- Das Arbeitsaktionsset ist inaktiviert.
- Die Workload ordnet die Aktivität direkt einer Serviceunterklasse zu.

Vergleich von Workload und Arbeitsaktionsset

Je nach Art der Kontrolle, die Sie über Ihre Datenbankaktivitäten haben wollen, können Sie ausschließlich Workloads oder Workloads und Arbeitsaktionen verwenden, um Aktivitäten zu Serviceklassen zuzuordnen.

Mit Workloads werden Anforderungen basierend auf Verbindungsattributen angegeben und einer Serviceklasse zugeordnet. Workloads sind die primäre Methode, um Arbeit zur Ausführung an eine bestimmte DB2-Serviceklasse weiterzuleiten. Wenn Sie weiter eingrenzen wollen, wie Anforderungen angegeben werden, können Sie Arbeitsklassen verwenden, um die Aktivitäten basierend auf ihrem Typ und anderen Aktivitätsattributen zu klassifizieren. Beispielsweise können Sie READ-, WRITE- und LOAD-Aktivitäten in unterschiedlichen Arbeitsklassen klassifizieren.

Wenn Sie Arbeitsklassen verwenden (die in Arbeitsklassensets gruppiert sind), können Sie über Arbeitsaktionen die unterschiedlichen Aktivitätstypen steuern. Sie können z. B. eine Arbeitsaktion verwenden, um einen bestimmten Typ von Aktivität einer Serviceunterklasse zuzuordnen, und eine andere Arbeitsaktion verwenden, um eine als Schwellenwert bezeichnete Steuerung anzuwenden, die sicherstellt, dass derselbe Typ von Aktivität nicht gegen bestimmte Bedingungen verstößt.

Arbeitsaktionen werden in Arbeitsaktionssets gruppiert. Ein einzelnes Arbeitsaktionsset kann sich auf Aktivitäten in der Datenbank oder auf Aktivitäten in einer Servicesuperklasse (aber nicht auf beide gleichzeitig) beziehen. Arbeitsklassensets und Arbeitsaktionssets arbeiten zusammen. Eine Arbeitsklasse muss zur Kategorisierung einer Aktivität als bestimmter Typ von Arbeit vorhanden sein, bevor eine Arbeitsaktion darauf angewendet werden kann. Ein Arbeitsklassenset kann mit mehreren Arbeitsaktionssets verknüpft sein, aber ein Arbeitsaktionsset kann nur mit einem Arbeitsklassenset verknüpft sein.

Die folgende Abbildung zeigt ein Beispiel einer Workload-Management-Implementierung mit Workloads und Arbeitsaktionssets. In dieser Abbildung wird davon ausgegangen, dass eine Anforderung der Workload WL_A zugeordnet wird, basierend auf der Benutzer-ID, mit der die Anforderung übergeben wurde. Die Workload WL_A definiert, dass die Anforderung in der Servicesuperklasse SC_A ausgeführt werden soll. Nehmen wir an, dass eine Arbeitsklasse im Arbeitsklassenset WCS_1 dem Arbeitstyp entspricht, den die Anforderung ausführt, die mit der Workload WL_A verknüpft ist.

Nehmen wir z. B. an, dass eine Aktivität, die die Kataloge nicht aktualisiert (eine READ-Aktivität), in das System eintritt. Das Arbeitsaktionsset WAS_1 auf Datenbankebene (das mit dem Arbeitsklassenset WCS_1 verknüpft ist) enthält eine Arbeitsaktion, die auf die Arbeitsklasse READ angewendet wird und einen Schwellenwert erzwingt, der angibt, dass höchstens 500 Aktivitäten gleichzeitig in der gesamten Datenbank ausgeführt werden können. Wenn die Anforderung die von diesem Schwellenwert definierte Begrenzung nicht überschreitet, wird sie anschließend (von der Workload WL_A) der Servicesuperklasse SC_A zugeordnet. Hier trifft die Anforderung auf das Arbeitsaktionsset WAS_2 auf der Ebene der Servicesuperklasse, das auch mit dem Arbeitsklassenset WCS_1 verknüpft ist und sich auf Aktivitäten in der Servicesuperklasse SC_A bezieht. Dieses Arbeitsaktionsset enthält eine Zuordnungsarbeitsaktion, die auch auf die Arbeitsklasse READ angewendet wird, so dass alle READ-Aktivitäten der Serviceunterklasse SSC_1a in der Servicesuperklasse SC_A zugeordnet werden.

Eine ähnliche Situation tritt mit der Anforderung auf, die (ebenfalls basierend auf ihren Verbindungsattributen) mit der Workload WL_B verknüpft ist. Die Workload WL_B ordnet Aktivitäten der Servicesuperklasse SC_B zu. Nehmen wir an, dass die Anforderung sich auf eine LOAD-Aktivität bezieht und dass das Arbeitsklassenset WCS_2 eine Arbeitsklasse für LOAD-Aktivitäten enthält. Das Arbeitsklassenset WCS_2 ist mit dem Arbeitsaktionsset WAS_3 auf der Ebene der Servicesuperklasse verknüpft, das auf Aktivitäten in der Servicesuperklasse SC_B angewendet wird. Nehmen wir an, dass das Arbeitsaktionsset WAS_3 eine Zuordnungsarbeitsaktion enthält, die auf die Arbeitsklasse LOAD angewendet wird, so dass sie beim Zuordnen der LOAD-Aktivität zur Servicesuperklasse SC_B durch die Workload WL_B von der Arbeitsaktion zur Ausführung der Serviceunterklasse SSC_1b zugeordnet wird.

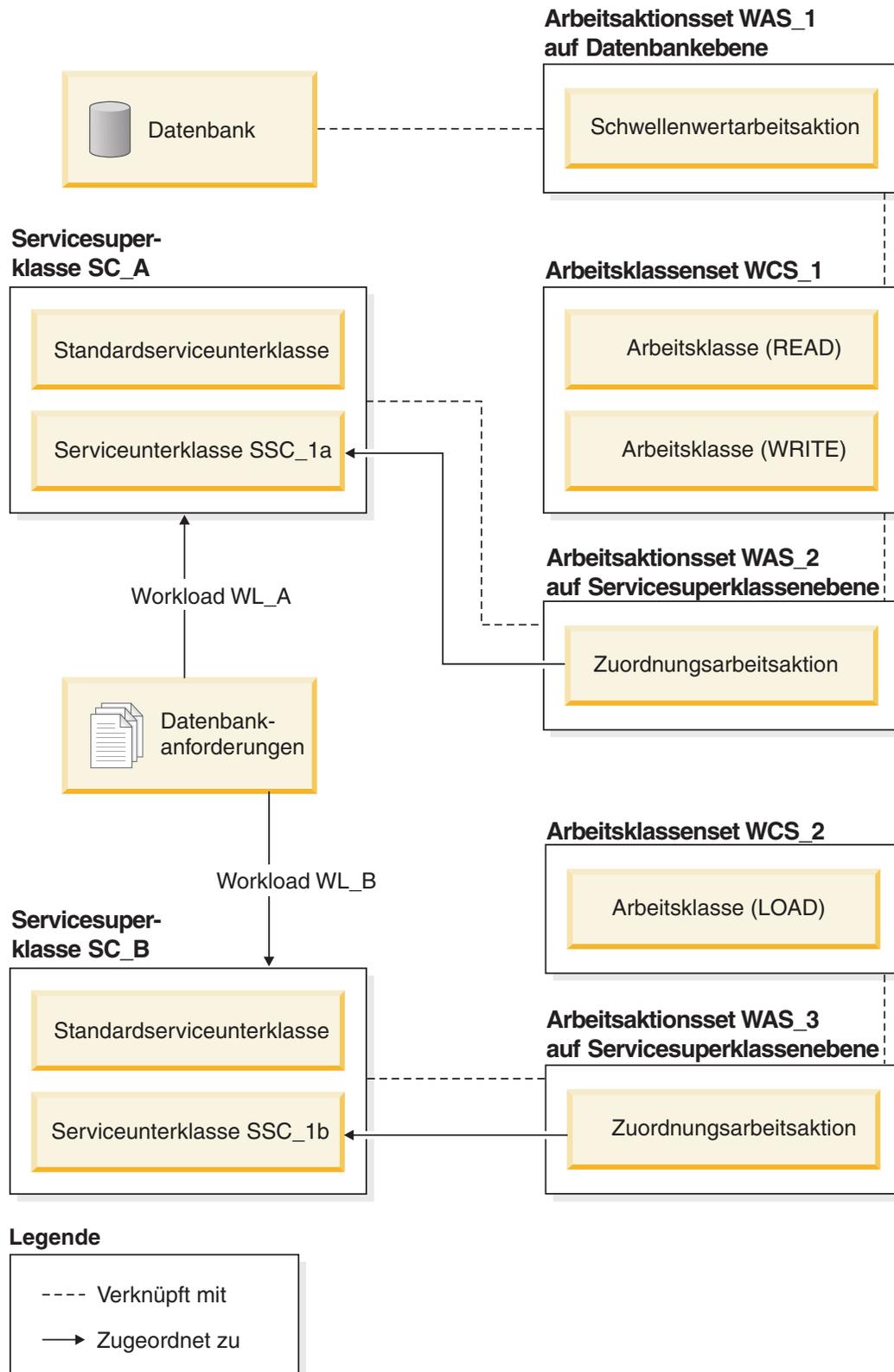


Abbildung 19. Workloads und Arbeitsaktionssets

Arbeiten mit Arbeitsaktionssets und Arbeitsaktionen

Erstellen eines Arbeitsaktionssets

Mit der Anweisung `CREATE WORK ACTION SET` können Sie eine Arbeitsaktion und ein Arbeitsaktionsset erstellen.

Zum Erstellen eines Arbeitsaktionssets benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Weitere Voraussetzungen sind in den folgenden Themen aufgeführt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Wenn Sie ein Arbeitsaktionsset erstellen:

- Sie ordnen es einem Arbeitsklassenset zu. Das Arbeitsklassenset muss bereits vorhanden sein.
- Sie ordnen es auch der Datenbank oder einer Servicesuperklasse zu. Wenn Sie das Arbeitsaktionsset einer Servicesuperklasse zuordnen, muss die Serviceklasse bereits vorhanden sein. Das Arbeitsaktionsset kann nicht für die Systemserviceklasse (`SYSDEFAULTSYSTEMCLASS`) oder die Verwaltungsserviceklasse (`SYSDEFAULTMAINTENANCECLASS`) definiert werden.

Gehen Sie wie folgt vor, um ein Arbeitsaktionsset zu erstellen:

1. Verwenden Sie die Anweisung `CREATE WORK ACTION SET` mit den folgenden Optionen:
 - Geben Sie einen Namen für das Arbeitsaktionsset an. Der Name des Arbeitsaktionssets muss in der Datenbank eindeutig sein.
 - Geben Sie das Objekt an, dem das Arbeitsaktionsset zugeordnet werden soll. Sie können eine Datenbank oder eine Servicesuperklasse angeben. Wenn Sie angeben, dass das Arbeitsaktionsset einer Datenbank zugeordnet werden soll, kann keine der Arbeitsaktionen im Arbeitsaktionsset eine Zuordnungsarbeitsaktion oder eine Arbeitsaktion für das Erfassen von zusammengefassten Aktivitätsdaten sein. Wenn Sie angeben, dass das Arbeitsaktionsset mit einer Servicesuperklasse verknüpft werden soll, darf keine der Arbeitsaktionen im Arbeitsaktionsset ein Schwellenwert sein. Wenn Sie z. B. das Arbeitsaktionsset auf die Servicesuperklasse `REPORTS` anwenden wollen, geben Sie Folgendes an:
`FOR SERVICE CLASS REPORTS`
Wenn Sie das Arbeitsaktionsset auf die Datenbank anwenden wollen, geben Sie Folgendes an:
`FOR DATABASE`
 - Geben Sie das Arbeitsklassenset an, mit dem das Arbeitsaktionsset verknüpft werden soll. Die Arbeitsklassen im Arbeitsklassenset klassifizieren die Datenbankaktivitäten, auf die die Arbeitsaktionen im Arbeitsaktionsset angewendet werden. Wenn Sie z. B. das Arbeitsaktionsset mit dem Arbeitsklassenset `LARGEREDS` verknüpfen wollen, geben Sie Folgendes an:
`USING WORK CLASS SET LARGEREDS`
 - Optional: Erstellen Sie eine oder mehrere Arbeitsaktionen für das Arbeitsaktionsset. Anweisungen hierzu finden Sie in „Erstellen einer Arbeitsaktion“ auf Seite 107.

- Geben Sie an, ob das Arbeitsaktionsset aktiviert oder inaktiviert ist. Standardmäßig ist das Arbeitsaktionsset aktiviert. Wenn das Arbeitsaktionsset inaktiviert ist, berücksichtigt der Datenserver dieses Arbeitsaktionsset (und alle darin enthaltenen Arbeitsaktionen) nicht bei der Ausführung von Aktivitäten.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird das Arbeitsaktionsset zur Sicht SYSCAT.WORKACTIONSETS hinzugefügt.
Ein neues Arbeitsaktionsset wird in der Datenbank erst nach dem Durchführen des Commits wirksam und beeinflusst keine der momentan ausgeführten Datenbankaktivitäten.

Ändern eines Arbeitsaktionssets

Mit der Anweisung ALTER WORK ACTION SET können Sie eine Arbeitsaktion in einem Arbeitsaktionsset hinzufügen, ändern oder löschen oder das Arbeitsaktionsset aktivieren oder inaktivieren.

Zum Ändern eines Arbeitsaktionssets benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Voraussetzungen sind in den folgenden Themen aufgeführt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Wenn Sie ein Arbeitsaktionsset erstellen, das mit einem bestimmten Arbeitsklassenset zusammenarbeiten soll, können Sie es nicht so ändern, dass es mit einem anderen Arbeitsklassenset zusammenarbeitet, da die Arbeitsaktionen im Arbeitsaktionsset von den Arbeitsklassen im Arbeitsklassenset abhängig sind. Wenn Sie das Arbeitsklassenset ändern wollen, auf das dieses Arbeitsaktionsset angewendet werden soll, müssen Sie das Arbeitsaktionsset löschen und neu erstellen.

Sie können nicht ändern, auf welches Objekt das Arbeitsaktionsset angewendet wird, da der Typ der Arbeitsaktionen im Arbeitsaktionsset davon abhängt, für welches Objekt (Datenbank oder Servicesuperklasse) das Arbeitsaktionsset definiert ist. Wenn Sie das Objekt ändern wollen, dem das Arbeitsaktionsset zugeordnet ist, müssen Sie das Arbeitsaktionsset löschen und neu erstellen.

Gehen Sie wie folgt vor, um ein Arbeitsaktionsset zu ändern:

1. Wenn Sie eine neue Arbeitsaktion zum Arbeitsaktionsset hinzufügen wollen, verwenden Sie das Schlüsselwort ADD. Informationen zu den Parametern, die Sie beim Hinzufügen einer Arbeitsaktion zu einem Arbeitsaktionsset angeben können, finden Sie in „Erstellen einer Arbeitsaktion“ auf Seite 107.
2. Wenn Sie eine vorhandene Arbeitsaktion ändern wollen, verwenden Sie das Schlüsselwort ALTER. Informationen zum Ändern einer Arbeitsaktion finden Sie in „Ändern einer Arbeitsaktion“ auf Seite 110.
3. Wenn Sie eine Arbeitsaktion löschen wollen, verwenden Sie das Schlüsselwort DROP. Informationen zum Löschen einer Arbeitsaktion aus einem Arbeitsaktionsset finden Sie in „Löschen einer Arbeitsaktion“ auf Seite 112.
4. Sie können ein derzeit inaktiviertes Arbeitsaktionsset aktivieren und umgekehrt. Wenn Sie ein aktiviertes Arbeitsaktionsset inaktivieren, ignoriert der Datenserver dieses Set nach dem Durchführen eines Commits für Ihre Änderungen. Weitere Informationen finden Sie in „Inaktivieren eines Arbeitsaktionssets“ auf Seite 106. Wenn Sie das Arbeitsaktionsset aktivieren,

wird das Arbeitsaktionsset nach dem Durchführen eines Commits für Ihre Änderungen auf die nächste zutreffende Aktivität angewendet, die in die Datenbank eingeht.

5. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht SYSCAT.WORKACTIONSETS aktualisiert. Alle hinzugefügten, geänderten oder gelöschten Arbeitsaktionen werden in der Sicht SYSCAT.WORKACTIONS aktualisiert.

Inaktivieren eines Arbeitsaktionssets

Mit dem Schlüsselwort `DISABLE` der Anweisung `CREATE WORK ACTION SET` oder der Anweisung `ALTER WORK ACTION SET` können Sie ein Arbeitsaktionsset inaktivieren.

Zum Inaktivieren eines Arbeitsaktionssets benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Während der Laufzeit wird ein inaktiviertes Arbeitsaktionsset so behandelt, als wäre es nicht vorhanden. Nehmen wir beispielsweise an, dass Sie ein Arbeitsaktionsset namens `READACTIVITIES` haben, das mit einem Arbeitsklassenset namens `READCLASSES` verknüpft ist, und dass dieses Arbeitsaktionsset für eine Servicesuperklasse namens `READSERVICECLASS` definiert ist. Das Arbeitsaktionsset `SMALLREAD` enthält eine Arbeitsaktion, die alle `SELECT`-Anweisungen der Serviceunterklasse `SMALLREADSERVICECLASS` neu zuordnet. Wenn das Arbeitsaktionsset `READACTIVITIES` inaktiviert ist, werden alle `SELECT`-Anweisungen so behandelt, als wäre das Arbeitsaktionsset `READACTIVITIES` nicht vorhanden. Sie werden daher der Standardserviceunterklasse zugeordnet.

Gehen Sie wie folgt vor, um ein Arbeitsaktionsset zu inaktivieren:

1. Verwenden Sie abhängig davon, ob Sie ein Arbeitsaktionsset erstellen oder ändern, eine der folgenden Anweisungen:
 - Verwenden Sie die Anweisung `CREATE WORK ACTION SET`, um das Arbeitsaktionsset zu inaktivieren. Zum Beispiel:

```
CREATE WORK ACTION SET name_des_arbeitsaktionssets ... DISABLE
```
 - Verwenden Sie die Anweisung `ALTER WORK ACTION SET`. Zum Beispiel:

```
ALTER WORK ACTION SET name_des_arbeitsaktionssets ... DISABLE
```
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Workload in der Sicht `SYSCAT.WORKACTIONSETS` aktualisiert.

Löschen eines Arbeitsaktionssets

Mit der Anweisung `DROP WORK ACTION SET` können Sie ein Arbeitsaktionsset löschen.

Zum Löschen eines Arbeitsaktionssets benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Beim Löschen eines Arbeitsaktionssets wird auch das Arbeitsaktionsset mit allen darin enthaltenen Arbeitsaktionen gelöscht.

Wenn das Arbeitsaktionsset eine Schwellenwertarbeitsaktion `CONCURRENTDBCOORDACTIVITIES` enthält, muss diese Arbeitsaktion zuerst inaktiviert werden, bevor das Arbeitsaktionsset gelöscht werden kann.

Gehen Sie wie folgt vor, um ein Arbeitsaktionsset zu löschen:

1. Verwenden Sie die Anweisung `DROP WORK ACTION SET`.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird das Arbeitsaktionsset aus der Sicht `SYSCAT.WORKACTIONSETS` entfernt. Darüber hinaus werden alle Arbeitsaktionen, die Teil des Arbeitsaktionssets waren, aus der Sicht `SYSCAT.WORKACTIONS` entfernt. Wenn das Arbeitsaktionsset Schwellenwertarbeitsaktionen enthält, werden die Schwellenwerte aus der Sicht `SYSCAT.THRESHOLDS` entfernt.

Erstellen einer Arbeitsaktion

Mit der Anweisung `CREATE WORK ACTION SET` oder `ALTER WORK ACTION SET` können Sie eine Arbeitsaktion erstellen.

Zum Erstellen einer Arbeitsaktion benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Weitere Voraussetzungen sind in den folgenden Themen aufgeführt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Wenn Sie eine Arbeitsaktion erstellen:

- Sie ordnen eine Arbeitsaktion einer Arbeitsklasse zu. Die Arbeitsklasse muss bereits im Arbeitsklassenset vorhanden sein, auf das das Arbeitsaktionsset angewendet wird.
- Wenn die Arbeitsaktion ein Schwellenwert ist, muss das Arbeitsaktionsset für die Datenbank definiert sein. Eine Liste der unterstützten Schwellenwerte für Arbeitsaktionen finden Sie in „In Arbeitsaktionen verwendbare Schwellenwerte“ auf Seite 97.
- Wenn Sie eine Zuordnungsarbeitsaktion erstellen, muss das Arbeitsaktionsset für eine Servicesuperklasse definiert sein. Die zuzuordnende Serviceunterklasse muss bereits in der Servicesuperklasse vorhanden sein, für die dieses Arbeitsaktionsset definiert wird. Zudem können Sie nicht die Standardserviceunterklasse angeben.
- Auf dieselbe Arbeitsklasse aus demselben Arbeitsaktionsset kann jeweils nur eine Arbeitsaktion eines Typs angewendet werden. Schwellenwerte bilden hierzu eine Ausnahme. Sie können mehrere Schwellenwerte auf eine Arbeitsklasse anwenden, aber jeder Schwellenwert muss einen anderen Typ haben.
- Wenn Sie eine Arbeitsaktion für das Erfassen von zusammengefassten Aktivitätsdaten erstellen, muss das Arbeitsaktionsset für eine Servicesuperklasse definiert sein.

Gehen Sie wie folgt vor, um eine Arbeitsaktion zu erstellen:

1. Verwenden Sie das Schlüsselwort *arbeitsaktionsdefinition* der Anweisung `CREATE WORK ACTION SET` oder das Schlüsselwort `ADD arbeitsaktionsdefinition` der Anweisung `ALTER WORK ACTION SET`. Geben Sie mindestens eine der folgenden Optionen für die Arbeitsaktion an:
 - Einen Namen für die Arbeitsaktion. Der Name der Arbeitsaktion muss im Arbeitsaktionsset eindeutig sein.
 - Den Namen der Arbeitsklasse, auf die diese Arbeitsaktion angewendet wird. Die Arbeitsklasse muss eine der Arbeitsklassen im Arbeitsklassenset sein, mit

dem das Arbeitsaktionsset verknüpft ist. Wenn Sie z. B. diese Arbeitsaktion auf die Arbeitsklasse LARGEDML anwenden wollen, geben Sie Folgendes an:
ON WORK CLASS LARGEDML

- Die Aktion, die die Aktivitäten anwenden soll, die der Arbeitsklasse für diese Arbeitsaktion entsprechen:

- Wenn das Arbeitsaktionsset mit einer Servicesuperklasse verknüpft ist, können Sie das Schlüsselwort MAP ACTIVITY angeben, so dass die Arbeitsaktion Aktivitäten einer Serviceunterklasse in der Servicesuperklasse zuordnet. Standardmäßig werden bei Zuordnungsarbeitsaktionen verschachtelte Aktivitäten derselben Serviceunterklasse wie das übergeordnete Element zugeordnet. Ein Cursor, der in einer Routine geöffnet wurde, ist ein Beispiel für eine verschachtelte Aktivität.

Wenn Sie z. B. wollen, dass die Arbeitsaktion der Serviceunterklasse SMALLREAD zugeordnet wird, und alle verschachtelten Aktivitäten derselben Serviceunterklasse zugeordnet werden sollen, geben Sie Folgendes an:

MAP ACTIVITY TO SMALLREAD

Sie könnten auch Folgendes angeben:

MAP ACTIVITY WITH NESTED TO SMALLREAD

Wenn die Arbeitsaktion der Serviceunterklasse zugeordnet werden soll, verschachtelte Aktivitäten jedoch nicht dieser Serviceunterklasse zugeordnet werden sollen, geben Sie Folgendes an:

MAP ACTIVITY WITHOUT NESTED TO SMALLREAD

Wenn Sie die Arbeitsaktion als WITHOUT NESTED definieren, werden verschachtelte Aktivitäten gemäß ihrem Aktivitätstyp behandelt, statt automatisch derselben Serviceunterklasse wie die übergeordnete Aktivität zugeordnet zu werden. Wenn z. B. eine CALL-Aktivität der Serviceunterklasse subsc1 zugeordnet wird und die Routine einen offenen Cursor enthält, könnte der offene Cursor einer anderen Serviceunterklasse zugeordnet werden, wenn er unter eine andere Arbeitsklasse fällt, auf die eine andere Zuordnungsarbeitsaktion angewendet wird.

- Wenn das Arbeitsaktionsset einer Datenbank zugeordnet ist, können Sie ein Schlüsselwort WHEN angeben, um festzulegen, dass ein Schwellenwert auf die Aktivität angewendet werden soll, und um die Aktion zu definieren, die durchzuführen ist, wenn die Aktivität einen Schwellenwertverstoß verursacht. Sie können die folgenden Schwellenwerte für eine Arbeitsaktion angeben:
 - CONCURRENTDBCOORDACTIVITIES und das zugehörige Schlüsselwort QUEUEDACTIVITIES
 - SQLTEMPSPACE
 - SQLROWSRETURNED
 - ESTIMATEDSQLCOST
 - ACTIVITYTOTALTIME

Anmerkung: Der Maximalwert, der für den Schwellenwert ACTIVITYTOTALTIME angegeben werden kann, beträgt 2147483400 Sekunden. Jeder (mit dem Schlüsselwort DAY, HOUR, MINUTE oder SECOND) angegebene Wert, der größer als 2147483400 Sekunden ist, wird auf den Maximalwert begrenzt.

Bei einem Schwellenwertverstoß können Sie die folgenden zu ergreifenden Maßnahmen angeben:

- Ob Aktivitätsdaten zu der Aktivität erfasst werden sollen, die den Schwellenwertverstoß verursachte. Wenn die Aktivitätsdaten beim Abschluss der Aktivitätsausführung erfasst werden, werden sie in einen aktiven Ereignismonitor für Aktivitäten geschrieben. Standardmäßig werden Daten zu Aktivitäten nicht erfasst. Wenn Sie Daten zu dieser Aktivität erfassen wollen, können Sie diese von der Koordinatorpartition, einer bestimmten Datenbankpartition oder von allen Datenbankpartitionen erfassen. Sie haben die Möglichkeit, diese Daten mit oder ohne Details zur Anweisung und ihrer Kompilierungsumgebung zu erfassen. Wenn Sie Details zur Anweisung und Kompilierungsumgebung erfassen wollen, können Sie auch angeben, dass die in der Aktivität verwendeten Eingabedatenwerte erfasst werden sollen.
- Ob die Aktivität, die den Schwellenwertverstoß verursachte, weiter ausgeführt werden darf. Standardmäßig wird die Aktivität gestoppt.

Wenn die Arbeitsaktion z. B. prüfen soll, ob DML-Anweisungen mit einem Aufwand von über 2 000 Timerons vorhanden sind, bei einem Schwellenwertverstoß die Basisdaten zu dieser Aktivität erfassen soll und die Aktion dann weiter ausgeführt werden soll, geben Sie Folgendes an:

```
WHEN ESTIMATEDSQLCOST > 2000 COLLECT ACTIVITY DATA CONTINUE
```

- Mit dem Schlüsselwort `PREVENT EXECUTION` können Sie verhindern, dass Aktivitäten ausgeführt werden, die der für diese Arbeitsaktion definierten Arbeitsklasse entsprechen.
- Wenn Sie die Anzahl der Datenbankaktivitäten, die der Arbeitsklasse zugeordnet sind, ohne den zusätzlichen Aufwand einer weiteren Aktion (wie z. B. dem Erfassen von Daten oder dem Zuordnen einer Aktivität) zählen wollen, können Sie das Schlüsselwort `COUNT ACTIVITY` angeben.
- Geben Sie das Schlüsselwort `COLLECT ACTIVITY DATA` an, um Aktivitätsdaten für Aktivitäten zu erfassen, die unter die Arbeitsklasse fallen. Wenn die Aktivitätsdaten beim Abschluss der Aktivitätsausführung erfasst werden, werden sie in einen aktiven Ereignismonitor für Aktivitäten geschrieben. Standardmäßig werden Daten zu Aktivitäten nicht erfasst. Wenn Sie Daten zu dieser Aktivität erfassen wollen, können Sie diese von der Koordinatorpartition oder von allen Datenbankpartitionen erfassen. Wenn Sie Aktivitätsdetails wie die Anweisung und die Kompilierungsumgebung erfassen wollen, können Sie dies mit dem Schlüsselwort `WITH DETAILS` tun. Sie können auch das Schlüsselwort `AND VALUES` verwenden, um Eingabedatenwerte (für Aktivitäten mit Eingabedatenwerten) an den Aktivitätsereignismonitor zu senden.

Nehmen wir beispielsweise an, Sie haben ein Arbeitsaktionsset, das auf eine Servicesuperklasse angewendet wird. Sie wollen, dass Aktivitätsdaten für alle Aktivitäten, die dieser Arbeitsaktion zugeordnet sind, in den entsprechenden Ereignismonitor geschrieben werden, einschließlich aller zusammengefasster Aktivitätsinformationen, aller Informationen zur Kompilierungsumgebung sowie aller Eingabedatenwerte. Geben Sie in diesem Fall Folgendes an:

```
COLLECT ACTIVITY DATA ON ALL WITH DETAILS AND VALUES
```

- Geben Sie das Schlüsselwort `COLLECT AGGREGATE ACTIVITY DATA` an, um zusammengefasste Aktivitätsdaten für Aktivitäten zu erfassen, die in die Arbeitsklasse fallen. In diesem Fall werden zusammengefasste Aktivitätsdaten erfasst und an den entsprechenden Ereignismonitor gesendet. Diese Informationen werden regelmäßig in einem vom Datenbankkonfigurationsparameter `wlm_collect_int` festgelegten Intervall erfasst.

Nehmen wir beispielsweise an, Sie haben ein Arbeitsaktionsset, das auf eine Servicesuperklasse angewendet wird. Sie wollen, dass zusammengefasste Aktivitätsdaten für alle Aktivitäten, die dieser Arbeitsaktion zugeordnet sind,

in den entsprechenden Ereignismonitor geschrieben werden, einschließlich aller Basisdaten, dem Histogramm des per DLL geschätzten Aufwands für Aktivitäten sowie dem Histogramm der per DML ermittelten Zwischenankunftszeit (Inter-Arrival Time) von Aktivitäten. Geben Sie in diesem Fall Folgendes an:

```
COLLECT AGGREGATE ACTIVITY DATA EXTENDED
```

- Die von einer Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA verwendeten Histogrammschablonen zum Beschreiben der für die entsprechende Arbeitsklasse erstellten Histogramme. Durch die Angabe der von einer Arbeitsaktion verwendeten Histogrammschablonen werden die entsprechenden Zeilen in der Sicht SYSCAT.HISTOGRAMTEMPLATEUSE hinzugefügt, die die Histogrammschablonen anzeigt, auf die die Serviceklasse oder Arbeitsaktion verweist. Wenn Sie z. B. Zwischenankunftsstatistikdaten für die Standardhistogrammschablone für die Zwischenankunftszeit erfassen wollen, geben Sie Folgendes an:

```
INTERARRIVALTIME HISTOGRAM TEMPLATE SYSDEFAULTHISTOGRAM
```

Weitere Informationen zu Histogrammen und Histogrammschablonen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.

- Angabe, ob die Arbeitsaktion aktiviert oder inaktiviert ist. Standardmäßig wird eine Arbeitsaktion als aktiviert erstellt. Sie können jedoch mit dem Schlüsselwort ENABLE oder DISABLE angeben, ob sie aktiviert oder inaktiviert sein soll. Wenn die Arbeitsaktion inaktiviert ist, berücksichtigt der Datenserver diese Arbeitsaktion nicht, wenn Aktivitäten in die Datenbank oder die Servicesuperklasse eingehen (je nach Objekt, für das Sie das Arbeitsaktionsset erstellt haben).
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen werden die Arbeitsaktionen zur Sicht SYSCAT.WORKACTIONS hinzugefügt. Wenn es sich bei der Arbeitsaktion um einen Schwellenwert handelt, wird der Schwellenwert zur Sicht SYSCAT.THRESHOLDS hinzugefügt.

Eine neue Arbeitsaktion wird in der Datenbank erst nach dem Durchführen des Commits wirksam und beeinflusst keine der momentan ausgeführten Datenbankaktivitäten.

Ändern einer Arbeitsaktion

Wenn Sie eine Arbeitsaktion ändern müssen, verwenden Sie die Anweisung ALTER WORK ACTION SET.

Zum Ändern einer Arbeitsaktion benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Voraussetzungen sind in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303 aufgeführt.

Gehen Sie wie folgt vor, um eine Arbeitsaktion zu ändern:

1. Mit dem Schlüsselwort ALTER der Anweisung ALTER WORK ACTION SET können Sie eines oder mehrere der folgenden Merkmale der Arbeitsaktion ändern.
 - Sie können die Arbeitsklasse ändern, auf die diese Arbeitsaktion angewendet wird. Die Arbeitsklasse muss bereits im Arbeitsklassenset vorhanden sein, auf das das Arbeitsaktionsset angewendet wird.

- Wenn die Arbeitsaktion einer Serviceunterklasse zugeordnet ist, können Sie ändern, welcher Serviceunterklasse die Datenbankaktivität zugeordnet werden soll. Sie können die Zuordnung nur zu einer Serviceunterklasse in derselben Servicesuperklasse ändern. Die Zuordnung zur Standardserviceunterklasse ist nicht möglich. Sie können auch ändern, ob verschachtelte Aktivitäten in der Aktivität derselben Serviceunterklasse zugeordnet werden sollen. Wenn z. B. die Arbeitsaktion derzeit als WITH NESTED definiert ist, können Sie dies in WITHOUT NESTED ändern. Diese Änderung würde bewirken, dass die verschachtelten Aktivitäten gemäß ihrem Aktivitätstyp behandelt werden, statt automatisch derselben Serviceunterklasse wie die übergeordnete Aktivität zugeordnet zu werden. Wenn z. B. eine CALL-Aktivität der Serviceunterklasse SUBSC1 zugeordnet wird und die Routine einen offenen Cursor enthält, könnte der offene Cursor einer anderen Serviceunterklasse zugeordnet werden, wenn er unter eine andere Arbeitsklasse fällt, auf die eine andere Zuordnungsarbeitsaktion angewendet wird.
- Sie können den Aktionstyp für die Arbeitsaktion (d. h. Zuordnung, Schwellenwert, Verhindern der Ausführung, Aktivitätenzählung, Erfassung) ändern. Sie müssen ihn jedoch in einen gültigen Arbeitstyp ändern. Wenn z. B. die Arbeitsaktion die Aktivität einer Serviceunterklasse zuordnen soll, können Sie die Arbeitsaktion nicht in einen Schwellenwert ändern oder umgekehrt. Der Grund hierfür ist, dass in diesem Beispiel das Arbeitsaktionsset auf eine Servicesuperklasse angewendet werden musste, um eine Zuordnungsaktion zu haben, und Schwellenwertaktionen für Arbeitsaktionssets, die Servicesuperklassen zugeordnet sind, nicht gültig sind. Wenn Sie den Typ einer Arbeitsaktion ändern, die eine Schwellenwertarbeitsaktion ist, oder wenn Sie den Typ einer Arbeitsaktion in einen Schwellenwert ändern, geschieht Folgendes:
 - Wenn die Arbeitsaktion ein Schwellenwert war und in eine andere Aktion geändert wurde, wird der Schwellenwert aus der Sicht SYSCAT.THRESHOLDS entfernt.
 - Wenn die Arbeitsaktion kein Schwellenwert war und in einen Schwellenwert geändert wurde, wird ein neuer Schwellenwert in der Sicht SYSCAT.THRESHOLDS erstellt.

Anmerkung: Wenn die Aktion ein Schwellenwert ist, können Sie den Schwellenwerttyp nicht in einen anderen Schwellenwert ändern. Wenn also die Arbeitsaktion ein Schwellenwert SQLROWSRETURNED war, können Sie ihn nicht in einen Schwellenwert SQLTEMPSPACE ändern. Außerdem können Sie den Arbeitsaktionstyp eines aktivierten Arbeitsaktionsschwellenwerts CONCURRENTDBCOORDACTIVITIES nicht ändern.

- Sie können die von einer Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA verwendeten Histogrammschablonen zum Beschreiben der für die entsprechende Arbeitsklasse erstellten Histogramme ändern. Durch das Aktualisieren der von einer Arbeitsaktion verwendeten Histogrammschablonen werden die entsprechenden Zeilen in der Sicht SYSCAT.HISTOGRAMTEMPLATEUSE aktualisiert, die die Histogrammschablonen anzeigt, auf die die Serviceklasse oder Arbeitsaktion verweist. Weitere Informationen zu Histogrammen und Histogrammschablonen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.
- Sie können angeben, ob Sie die Arbeitsaktion aktivieren oder inaktivieren wollen. Standardmäßig sind Arbeitsaktionen aktiviert. Wenn die Arbeitsaktion aktiviert ist, berücksichtigt sie der Datenservert für die Anwendung auf die Aktivität, die unter die Arbeitsklasse für die Arbeitsaktion fällt. Wenn die Arbeitsaktion inaktiviert ist, ignoriert der Datenservert sie.

2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Arbeitsaktion in der Sicht SYSCAT.WORKACTIONS aktualisiert.

Inaktivieren einer Arbeitsaktion

Sie können eine Arbeitsaktion inaktivieren, die nicht mehr auf eine Arbeitsklasse angewendet werden soll. Während der Laufzeit wird die inaktivierte Arbeitsaktion so behandelt, als wäre sie nicht vorhanden.

Zum Inaktivieren einer Arbeitsaktion benötigen Sie die Berechtigung SYSADM oder DBADM.

Gehen Sie wie folgt vor, um eine Arbeitsaktion zu inaktivieren:

1. Verwenden Sie abhängig davon, ob Sie ein Arbeitsaktionsset erstellen oder ändern, eine der folgenden Anweisungen:
 - Verwenden Sie das Schlüsselwort DISABLE und das Schlüsselwort ADD der Anweisung CREATE WORK ACTION SET. Zum Beispiel:

```
ADD WORK ACTION arbeitsaktionsname ON WORK CLASS arbeitsklassenname ... DISABLE
```
 - Verwenden Sie das Schlüsselwort DISABLE und das Schlüsselwort ALTER der Anweisung ALTER WORK ACTION SET. Zum Beispiel:

```
ALTER WORK ACTION arbeitsaktionsname ... DISABLE
```
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Arbeitsaktion in der Sicht SYSCAT.WORKACTIONS aktualisiert.

Löschen einer Arbeitsaktion

Wenn Sie eine Arbeitsaktion nicht mehr benötigen, können Sie sie aus dem Arbeitsaktionsset löschen.

- Zum Löschen einer Arbeitsaktion benötigen Sie die Berechtigung SYSADM oder DBADM.
- Weitere Voraussetzungen sind in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303 aufgeführt.

Gehen Sie wie folgt vor, um eine Arbeitsaktion zu löschen:

1. Verwenden Sie das Schlüsselwort DROP der Anweisung ALTER WORK ACTION SET. Wenn Sie eine Arbeitsaktion mit dem Schwellenwert CONCURRENTDBCOORDACTIVITIES löschen wollen, müssen Sie die Arbeitsaktion in einer Operation ALTER WORK ACTION SET inaktivieren, ein Commit für diese Änderung durchführen und dann den Schwellenwert in einer zweiten Operation ALTER WORK ACTION SET löschen.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Arbeitsaktion aus der Sicht SYSCAT.WORKACTIONS entfernt. Wenn es sich bei der Arbeitsaktion um eine Schwellenwertarbeitsaktion handelt, wird auch der Schwellenwert aus der Sicht SYSCAT.THRESHOLDS entfernt.

Ein geändertes Arbeitsaktionsset und eine geänderte Arbeitsaktion werden in der Datenbank erst nach dem Durchführen eines Commits für die Änderung wirksam und beeinflussen keine der momentan ausgeführten Datenbankaktivitäten.

Arbeiten mit Arbeitsklassensets und Arbeitsklassen

Erstellen eines Arbeitsklassensets

Mit der Anweisung `CREATE WORK CLASS SET` können Sie ein Arbeitsklassenset erstellen.

Zum Erstellen eines Arbeitsklassensets benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Weitere Voraussetzungen sind in den folgenden Themen aufgeführt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Gehen Sie wie folgt vor, um ein Arbeitsklassenset zu erstellen:

1. Geben Sie die folgenden Merkmale für das Arbeitsklassenset mit der Anweisung `CREATE WORK CLASS SET` an:
 - Einen Namen für das Arbeitsklassenset. Der angegebene Name muss in der Datenbank eindeutig sein.
 - Optional: Eine oder mehrere Arbeitsklassen für das Arbeitsklassenset. Weitere Informationen finden Sie in „Erstellen einer Arbeitsklasse“ auf Seite 114.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird das Arbeitsklassenset zur Sicht `SYSCAT.WORKCLASSETS` hinzugefügt.

Ändern eines Arbeitsklassensets

Nach der Erstellung eines Arbeitsklassensets können die Attribute des Arbeitsklassensets nicht mehr geändert werden. Sie können jedoch Arbeitsklassen im Arbeitsklassenset mit der Anweisung `ALTER WORK CLASS SET` hinzufügen, ändern und löschen.

Zum Ändern eines Arbeitsklassensets benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Weitere Voraussetzungen sind in den folgenden Themen aufgeführt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
 - Namensregeln
1. Wenn Sie eine neue Arbeitsklasse zum Arbeitsklassenset hinzufügen wollen, verwenden Sie das Schlüsselwort `ADD`. Informationen zu den Schlüsselwörtern, die Sie beim Hinzufügen einer Arbeitsklasse angeben können, finden Sie in „Erstellen einer Arbeitsklasse“ auf Seite 114.
 2. Wenn Sie eine Arbeitsklasse ändern wollen, verwenden Sie das Schlüsselwort `ALTER`. Informationen zum Ändern einer Arbeitsklasse finden Sie in „Ändern einer Arbeitsklasse“ auf Seite 117.
 3. Wenn Sie eine Arbeitsklasse löschen wollen, verwenden Sie das Schlüsselwort `DROP`. Informationen zum Löschen einer Arbeitsklasse aus einem Arbeitsklassenset finden Sie in „Löschen einer Arbeitsklasse“ auf Seite 117. Wenn Sie alle Arbeitsklassen aus dem Arbeitsklassenset löschen wollen, können Sie das Arbeitsklassenset selbst löschen. Weitere Informationen finden Sie in „Löschen eines Arbeitsklassensets“ auf Seite 114.

4. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Sicht SYSCAT.WORKCLASSES aktualisiert, um hinzugefügte, geänderte oder gelöschte Arbeitsklassen anzuzeigen.

Löschen eines Arbeitsklassensets

Mit der Anweisung DROP WORK CLASS SET können Sie ein Arbeitsklassenset löschen.

Zum Löschen eines Arbeitsklassensets benötigen Sie die Berechtigung SYSADM oder DBADM.

Sie können ein Arbeitsklassenset nur löschen, wenn es mit keinen Arbeitsaktionssets verknüpft sind. Wenn Sie das Arbeitsklassenset löschen wollen, müssen Sie zuerst die abhängigen Arbeitsaktionssets löschen.

Gehen Sie wie folgt vor, um ein Arbeitsklassenset zu löschen:

1. Verwenden Sie die Anweisung DROP WORK CLASS SET.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird das Arbeitsklassenset aus der Sicht SYSCAT.WORKCLASSETS entfernt. Darüber hinaus werden alle Arbeitsklassen, die Teil des Arbeitsklassensets waren, aus der Sicht SYSCAT.WORKCLASSES entfernt.

Erstellen einer Arbeitsklasse

Mit der Anweisung CREATE WORK CLASS SET oder ALTER WORK CLASS SET können Sie eine Arbeitsklasse erstellen.

Zum Erstellen einer Arbeitsklasse benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Voraussetzungen sind in den folgenden Themen aufgeführt:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Gehen Sie wie folgt vor, um eine Arbeitsklasse zu erstellen:

1. Sie können eine Arbeitsklasse zur gleichen Zeit wie ein neues Arbeitsklassenset erstellen oder eine neue Arbeitsklasse einem vorhandenen Arbeitsklassenset hinzufügen:
 - Verwenden Sie das Schlüsselwort WORK CLASS der Anweisung CREATE WORK CLASS SET, um eine neue Arbeitsklasse zu erstellen, die einem neuen Arbeitsklassenset hinzugefügt wird.
 - Verwenden Sie das Schlüsselwort ADD WORK CLASS der Anweisung ALTER WORK CLASS SET, um eine neue Arbeitsklasse zu erstellen, die einem vorhandenen Arbeitsklassenset hinzugefügt wird.

Geben Sie mindestens eines der folgenden Merkmale für die neue Arbeitsklasse an:

- Einen Namen für die Arbeitsklasse. Dieser Name muss im Arbeitsklassenset eindeutig sein.
- Attribute für die Arbeitsklasse. Diese Attribute werden verwendet, um eine Aktivität mit der Arbeitsklasse zu verknüpfen:

- Den Typ von Arbeit, für den die Arbeitsklasse verwendet wird. Verwenden Sie den Parameter WORK TYPE, um dieses Merkmal anzugeben.

- READ, das nicht-aktualisierende SELECT-Aktivitäten sowie alle XQuery-Aktivitäten darstellt.

Wenn Sie das Schlüsselwort READ angeben, können Sie auch eine optionale Klausel FOR-FROM-TO angeben. Mit dieser Klausel können Sie einen Bereich für den Aufwand der Anweisung in Timerons oder ihre Kardinalität (d. h. die Anzahl der zurückgegebenen Zeilen) angeben. Sie müssen einen numerischen Wert für den ersten Wert angeben. Für den zweiten Wert können Sie entweder einen numerischen Wert oder den Wert UNBOUNDED angeben, wenn Sie keine Obergrenze für den Aufwand oder die Kardinalität der Aktivität erzwingen wollen. Sie können dieses Argument auch für das Schlüsselwort WRITE, das Schlüsselwort DML und das Schlüsselwort ALL angeben.

Wenn Sie z. B. SELECT-Aktivitäten mit einem Aufwand von mindestens 5000 Timerons mit dieser Arbeitsklasse verknüpfen wollen, geben Sie Folgendes an:

```
WORK TYPE READ FOR TIMERONCOST FROM 5000 TO UNBOUNDED
```

- WRITE, das SQL-Aktivitäten darstellt, die Daten in der Datenbank aktualisieren.

Wenn Sie z. B. mit dieser Arbeitsklasse WRITE-Aktivitäten verknüpfen wollen, die zwischen 50 und 100 Zeilen aktualisieren, geben Sie Folgendes an:

```
WORK TYPE WRITE FROM 50 TO 100
```

- CALL, das CALL-Aktivitäten darstellt.

Wenn Sie das Schlüsselwort CALL angeben, können Sie auch das Schlüsselwort ROUTINES IN SCHEMA angeben, wenn nur CALL-Aktivitäten für Routinen in einem bestimmten Schema mit dieser Arbeitsklasse verknüpft werden sollen. Wenn Sie dieser Arbeitsklasse z. B. nur Aufrufe an Routinen im Schema ACCOUNTS zuordnen wollen, geben Sie Folgendes an:

```
WORK TYPE CALL ROUTINES IN SCHEMA ACCOUNTS
```

- DML, das SQL-Aktivitäten darstellt, die von den Schlüsselwörtern READ und WRITE abgedeckt werden.

Wenn Sie z. B. DML-Aktivitäten mit einem Aufwand von 500 bis 1000 Timerons mit dieser Arbeitsklasse verknüpfen wollen, geben Sie Folgendes an:

```
WORK TYPE DML FOR TIMERONCOST FROM 500 TO 1000
```

- DDL, das die folgenden Aktivitäten darstellt:

- ALTER
- CREATE
- COMMENT
- DECLARE GLOBAL TEMPORARY TABLE
- DROP
- FLUSH PACKAGE CACHE
- GRANT
- REFRESH TABLE
- RENAME
- REVOKE
- SET INTEGRITY

Wenn Sie mit dieser Arbeitsklasse z. B. alle DDL-Aktivitäten verknüpfen wollen, geben Sie Folgendes an:

```
WORK TYPE DDL
```

- LOAD, das eine LOAD-Aktivität darstellt.

Wenn Sie mit dieser Arbeitsklasse z. B. alle LOAD-Aktivitäten verknüpfen wollen, geben Sie Folgendes an:

```
WORK TYPE LOAD
```

- ALL, das alle durch die oben aufgelisteten Schlüsselwörter bezeichneten Arbeitstypen darstellt.

Wenn Sie ALL als Arbeitsklassentyp angeben, können Sie auch das Schlüsselwort ROUTINES IN SCHEMA angeben, wenn nur CALL-Aktivitäten für Routinen in einem bestimmten Schema mit dieser Arbeitsklasse verknüpft werden sollen. Sie können auch die Klausel FOR-FROM-TO angeben, wenn alle DML-Aktivitäten, für die ein geschätzter Timeronaufwand oder eine Kardinalität angegeben ist, unter diese Klasse fallen sollen. Wenn Sie z. B. mit dieser Arbeitsklasse sowohl DML-Aktivitäten mit einer Kardinalität von 300 bis 1500 Zeilen als auch Routinen, die vom Schema NEWHIRES aufgerufen werden, verknüpfen wollen, geben Sie Folgendes an:

```
WORK TYPE ALL FOR CARDINALITY FROM 300 TO 1500 ROUTINES IN SCHEMA NEWHIRES
```

Da diese Arbeitsklasse den Typ ALL hat, würde sie auch auf andere Aktivitäten angewendet, die kein Schema oder keine Kardinalität haben, z. B: LOAD-Aktivitäten und DDL-Aktivitäten.

- Optional. Die Position der Arbeitsklasse im Arbeitsklassenset. Die Position der Arbeitsklasse im Arbeitsklassenset bestimmt die Reihenfolge, in der die Arbeitsklasse beim Klassifizieren einer Aktivität in eine Arbeitsklasse ausgewertet wird. Wenn eine Arbeitsklassenzuordnung erfolgt, ermittelt der Datenserver zuerst das Arbeitsklassenset, das dem Objekt (entweder einer Servicesuperklasse oder der Datenbank) zugeordnet ist, und wählt dann die erste übereinstimmende Arbeitsklasse im Arbeitsklassenset aus, mit dem eine Arbeitsaktion verknüpft ist. Geben Sie mit Hilfe des Schlüsselworts POSITION eine der folgenden Optionen an:

- LAST. Die Arbeitsklasse wird am Ende der Liste der Arbeitsklassen im Arbeitsklassenset platziert. Zum Beispiel:

```
WORK TYPE ... POSITION LAST
```

- BEFORE *arbeitsklassenname*. Die Arbeitsklasse soll im Arbeitsklassenset erstellt und vor der angegebenen Arbeitsklasse platziert werden. Zum Beispiel:

```
WORK TYPE ... POSITION BEFORE LARGEDDL
```

- AFTER *arbeitsklassenname*. Die Arbeitsklasse soll im Arbeitsklassenset erstellt und nach der angegebenen Arbeitsklasse platziert werden. Zum Beispiel:

```
WORK TYPE ... POSITION AFTER LARGEDDL
```

- AT *ganzzahl*. Die Arbeitsklasse soll im Arbeitsklassenset an der durch die ganze Zahl angegebenen Position platziert werden. Zum Beispiel:

```
WORK TYPE ... POSITION AT 3
```

2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Arbeitsklasse zur Sicht SYSCAT.WORKCLASSES hinzugefügt.

Ändern einer Arbeitsklasse

Wenn Sie eine Arbeitsklasse ändern müssen, verwenden Sie die Anweisung ALTER WORK CLASS SET.

Zum Ändern einer Arbeitsklasse benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Voraussetzungen sind in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303 aufgeführt.

Gehen Sie wie folgt vor, um eine Arbeitsklasse zu ändern:

1. Verwenden Sie das Schlüsselwort ALTER der Anweisung ALTER WORK CLASS SET, um eines oder mehrere der folgenden Merkmale zu ändern. Eine Erläuterung der unterstützten Werte für diese Merkmale finden Sie in „Erstellen einer Arbeitsklasse“ auf Seite 114.
 - Das Schlüsselwort FOR. Sie können z. B. den Wert für das Schlüsselwort FOR von CARDINALITY in TIMERONCOST ändern.
 - Das Argument FROM *anfangswert* TO *endwert*. Sie können z. B. das Argument von FROM 50 TO 100 in FROM 500 TO 1500 ändern.
 - Das Schlüsselwort SCHEMA für CALL-Aktivitäten. Wenn z. B. die Arbeitsklasse derzeit kein Schema angibt, können Sie eines hinzufügen. Sie können auch das Schlüsselwort ALL angeben, so dass die Arbeitsklasse unabhängig vom Schema der Routine auf alle CALL-Anweisungen angewendet wird. ALL ist die Standardeinstellung.
 - Das Schlüsselwort POSITION. Sie können z. B. eine Arbeitsklasse mit dem Schlüsselwort AT von der letzten Position an eine beliebige Position oder mit dem Schlüsselwort LAST von einer beliebigen Position an die letzte Position versetzen.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Arbeitsklasse in der Sicht SYSCAT.WORKCLASSES aktualisiert.

Löschen einer Arbeitsklasse

Wenn Sie eine Arbeitsklasse nicht mehr benötigen, können Sie sie aus dem Arbeitsklassenset löschen.

Zum Löschen einer Arbeitsklasse benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Voraussetzungen sind in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303 aufgeführt.

Gehen Sie wie folgt vor, um eine Arbeitsklasse zu löschen:

1. Verwenden Sie das Schlüsselwort DROP der Anweisung ALTER WORK CLASS SET. Sie können eine Arbeitsklasse nicht löschen, wenn eine Arbeitsaktion in einem beliebigen Arbeitsaktionsset, das mit dieser Arbeitsklasse verknüpft ist, von der Arbeitsklasse abhängt, die Sie löschen wollen. In dieser Situation müssen Sie zuerst alle abhängigen Arbeitsaktionen löschen, bevor Sie die Arbeitsklasse löschen können.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird die Arbeitsklasse aus der Sicht SYSCAT.WORKCLASSES entfernt.

Teil 3. Überwachung und Steuerung

Kapitel 6. Überwachung und Steuerung

Mit den Workload-Management-Features und -Leistungsmerkmalen können Sie die in der Datenbank ausgeführte Arbeit sowohl überwachen als auch steuern.

Sie können z. B. die folgenden Tasks ausführen:

- Analysieren der Auslastung auf Ihrem System als Hilfe für die Erstellung Ihrer Workload-Management-Anfangskonfiguration.
- Verfolgen und Untersuchen des Verhaltens Ihres Systems, indem Sie die folgenden Arten von Betriebsdaten beziehen:
 - Allgemeine Überwachungsdaten zur Umgebung
 - Informationen für die Analyse der Verschlechterung der Systemleistung
 - Informationen für die Diagnose nicht mehr reagierender Aktivitäten
 - Informationen zur Untersuchung von Konkurrenzsituationen bei Agenten
 - Informationen zur Eingrenzung von Abfragen mit schlechter Leistung

Informationen sind für Aktivitäten, Serviceklassen, Workloads, Arbeitsklassen, Schwellenwertwarteschlangen und Schwellenwertverstöße verfügbar.

- Steuern der Umgebung durch Abbruch von Aktivitäten in der Warteschlange, bei denen Sie Probleme erwarten, oder Abbrechen von ausgeführten Aktivitäten, bei denen Sie eine negative Auswirkung auf das System diagnostiziert haben.

Mit der Workload-Management-Lösung erreichen Sie eine effektivere Fehlerbehebung, da Sie von der Datenbank über Serviceklassen und Workloads bis hinunter zu einzelnen Aktivitäten in der Datenbank immer detailliertere Informationen abrufen können.

Überblick über Überwachungsdaten

Überwachungsdaten sind von Workloads, Arbeitsklassen, Serviceunterklassen, Servicesuperklassen und Schwellenwertwarteschlangen verfügbar. Mit Hilfe dieser Daten können Sie Probleme diagnostizieren und beheben und die Leistung optimieren.

Die folgende Abbildung zeigt die Überwachungsdaten, die für Workloads verfügbar sind. Mit Ereignismonitoren können Sie Workloadstatistikdaten und Informationen zu Aktivitäten erfassen, die in den Workloads ausgeführt werden. Mit Tabellenfunktionen können Sie in Echtzeit auf Workload-Statistikdaten und Informationen zu Workloadvorkommen zugreifen.

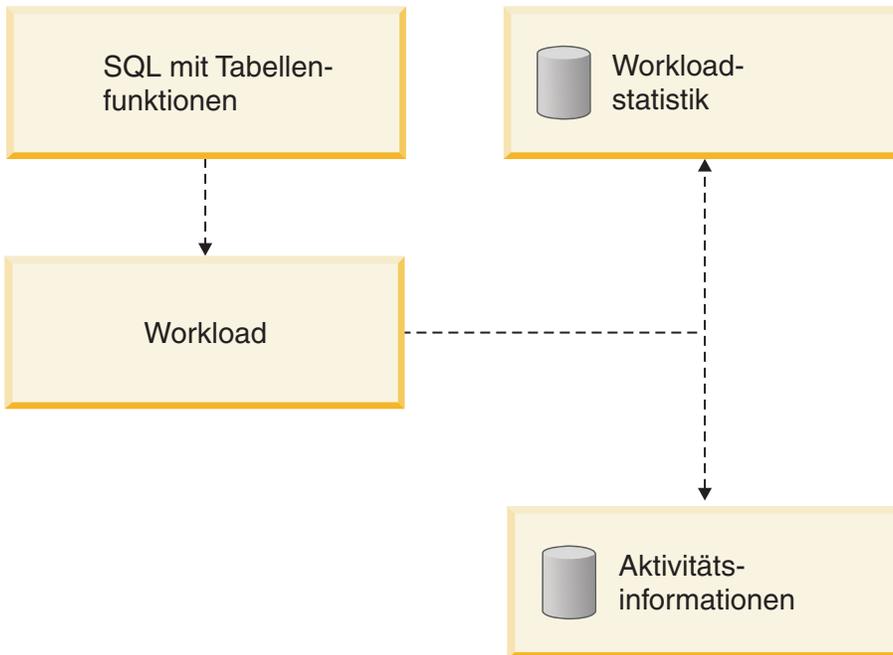


Abbildung 20. Für Workloads verfügbare Überwachungsdaten

Die folgende Abbildung zeigt die Überwachungsdaten, die für Serviceklassen verfügbar sind. Sie können Statistikdaten für Serviceunterklassen und Servicesuperklassen erfassen. Bei Serviceunterklassen können Sie auch zusammengefasste Aktivitäts- und Anforderungsstatistikdaten ermitteln sowie Informationen zu Aktivitäten, die in der Serviceunterklasse ausgeführt werden. Mit Tabellenfunktionen können Sie in Echtzeit auf Statistikdaten zu Servicesuperklassen und Serviceunterklassen sowie auf Informationen zu Agenten in einer bestimmten Serviceklasse zugreifen.

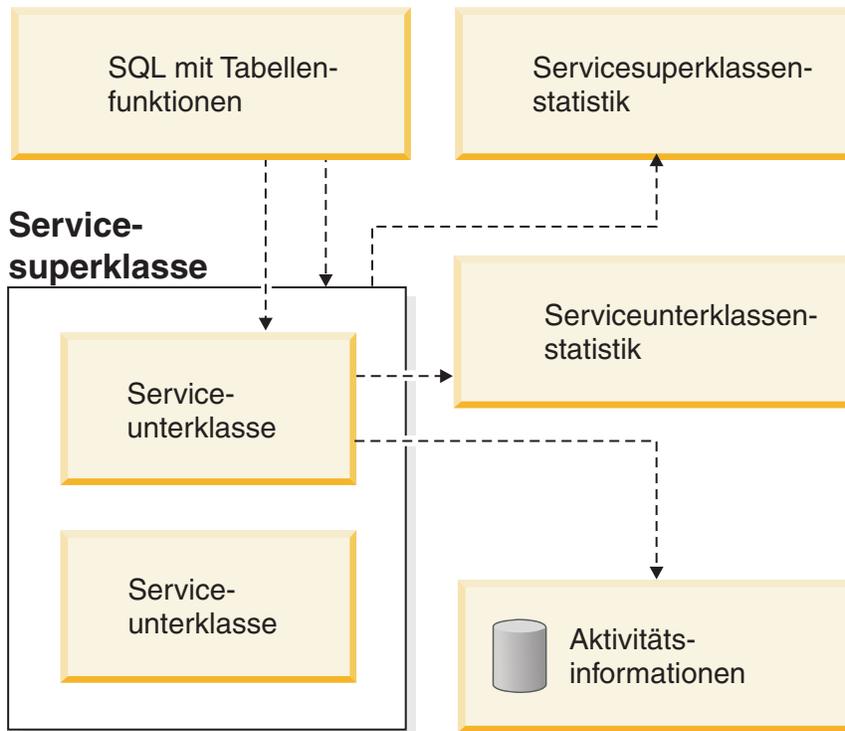


Abbildung 21. Für Serviceklassen verfügbare Überwachungsdaten

Die folgende Abbildung zeigt die Überwachungsdaten, die für Arbeitsklassen verfügbar sind. Sie können Statistikdaten zu Arbeitsklassen sowie Informationen zu Aktivitäten erfassen, die einer bestimmten Arbeitsklasse zugeordnet sind. Mit Tabellenfunktionen können Sie in Echtzeit auf Statistikdaten zu Arbeitsklassen zugreifen.

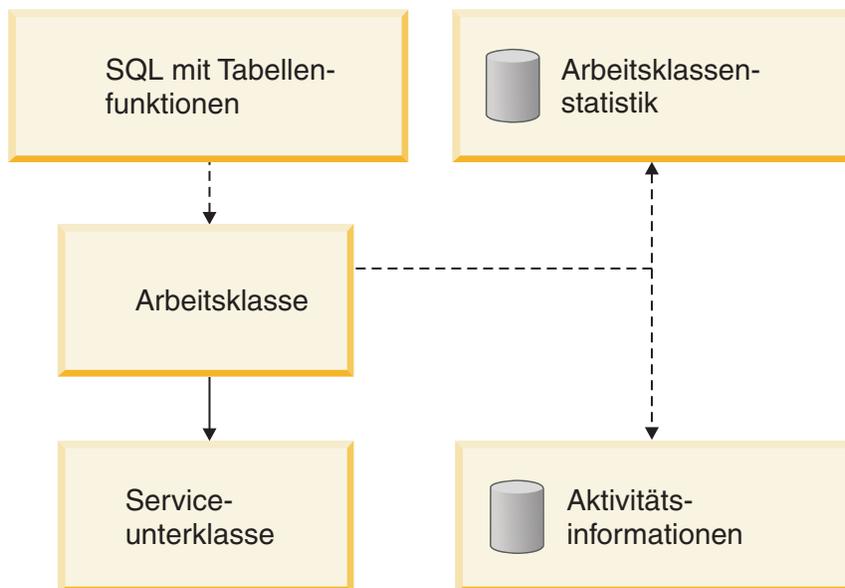


Abbildung 22. Für Arbeitsklassen verfügbare Überwachungsdaten

Die folgende Abbildung zeigt die Überwachungsdaten, die für Schwellenwerte verfügbar sind. Sie können Informationen zu Schwellenwertverstößen, zu den Aktivitäten, die die Schwellenwertverstöße verursacht haben, sowie Statistikdaten zu Warteschlangen (für Warteschlangenschwellenwerte) erfassen. Mit Tabellenfunktionen können Sie in Echtzeit auf Statistikdaten zu Warteschlangenschwellenwerten zugreifen.

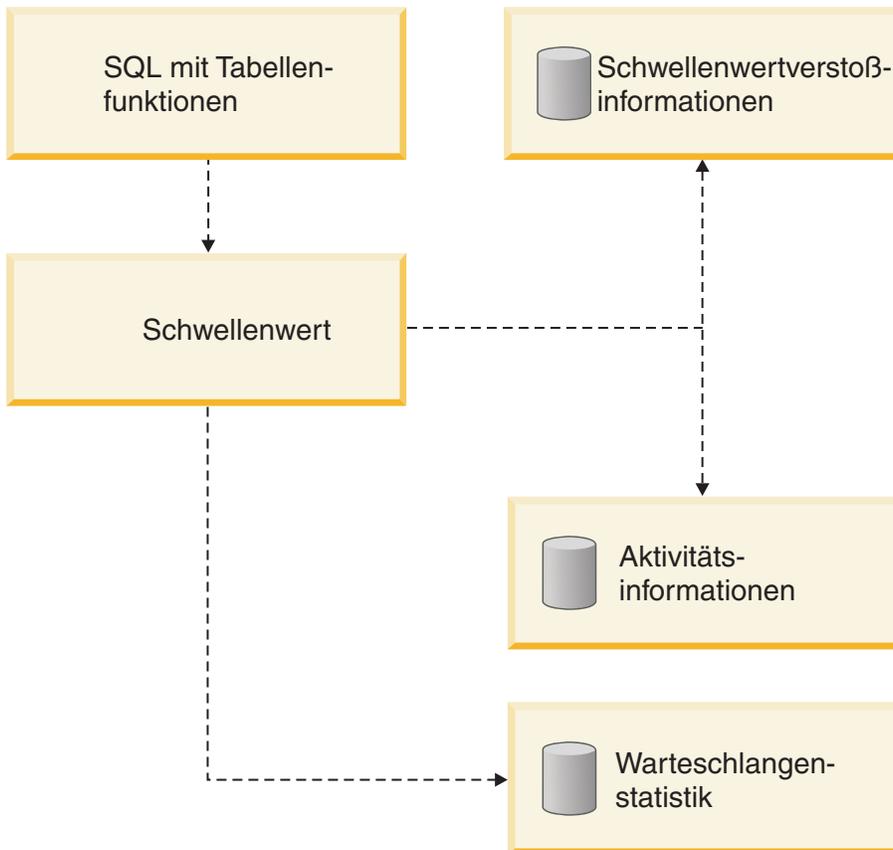


Abbildung 23. Für Schwellenwerte verfügbare Überwachungsdaten

Workload-Management-Funktionen für das Abrufen von Betriebsinformationen

Mit den in diesem Thema beschriebenen Tabellenfunktionen können Sie Betriebsinformationen erhalten.

Die Workload-Management-Tabellenfunktionen sind im Schema SYSPROC verfügbar. Diese Hochleistungsfunktionen können Informationen zur im System anfallenden Arbeit zurückgeben, ohne dadurch die zurzeit ausgeführten Workloads zu beeinträchtigen.

Sie können die folgenden Tabellenfunktionen verwenden, um die auf dem System auftretende Arbeit im Hinblick auf Serviceklassen, Workloadvorkommen, Agenten, Anforderungen und Aktivitäten zu untersuchen. Alle Tabellenfunktionen können Informationen entweder zu einer einzigen Datenbankpartition oder für alle Datenbankpartitionen in einer Umgebung mit partitionierten Datenbanken zurückgeben.

- `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES`(*servicesuperklassenname, serviceunterklassenname, dbpartitionsnummer*). Mit dieser Tabellenfunktion erhalten Sie eine Liste von Workloadvorkommen in einer Datenbank. Ein Workloadvorkommen ist eine Datenbankverbindung mit Attributen, die mit einer Workloaddefinition übereinstimmen. Sie können die Workloadvorkommen entweder für eine bestimmte Serviceklasse oder für alle Serviceklassen auflisten. Weitere Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Untersuchen der Agentenauslastung nach Serviceklasse“ auf Seite 192.
- `WLM_GET_SERVICE_CLASS_AGENTS`(*servicesuperklassenname, serviceunterklassenname, anwendungskennung, dbpartitionsnummer*). Mit dieser Tabellenfunktion erhalten Sie eine Liste von Agenten, die in der Datenbank arbeiten. Sie können alle Agenten auflisten, die in einer bestimmten Serviceklasse ausgeführt werden, oder alle Agenten, die für eine bestimmte Anwendung arbeiten. Mit dieser Tabellenfunktion können Sie auch den Status des Koordinatoragenten und der Subagenten für Anwendungen ermitteln und feststellen, an welchen Anforderungen jeder Agent im System arbeitet. Weitere Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Untersuchen der Agentenauslastung nach Serviceklasse“ auf Seite 192.
- `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES`(*anwendungskennung, dbpartitionsnummer*). Mit dieser Tabellenfunktion erhalten Sie eine Liste aktiver Aktivitäten, die einem Workloadvorkommen auf einer bestimmten Datenbankpartition zugeordnet sind. (Sie können mit Hilfe von Platzhalterzeichen mehrere Anwendungs-IDs und Datenbankpartitionen angeben.) Diese Tabellenfunktion gibt Informationen zu Aktivitäten zurück, die sich in der Warteschlange befinden, im Leerlauf sind oder ausgeführt werden. Diese Tabellenfunktion gibt keine Informationen zu Aktivitäten zurück, deren Ausführung abgeschlossen ist. Weitere Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Zusammenfassen von Daten mit Hilfe von Workload-Management-Tabellenfunktionen“ auf Seite 181 und „Beispiel: Erkennen von blockierten Aktivitäten“ auf Seite 188.
- `WLM_GET_ACTIVITY_DETAILS`(*anwendungskennung, uow-id, aktivitäts-id, dbpartitionsnummer*). Mit dieser Tabellenfunktion erhalten Sie detaillierte Informationen zu einer ablaufenden Aktivität, die Sie durch die eindeutige Kombination der Aktivitäts-ID, UOW-ID und Anwendungs-ID bezeichnen. Sie können diese Tabellenfunktion verwenden, um Informationen zu Aktivitäten zu analysieren, die von der Tabellenfunktion `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES` zurückgegeben werden. Weitere Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Verwenden von Workload-Management-Tabellenfunktionen zum Überwachen des aktuellen Systemverhaltens auf verschiedenen Ebenen“ auf Seite 177.

Integration von Workload-Management-Tabellenfunktionen und des Überwachungsprogramms für Momentaufnahme

Sie können bei der Fehlerermittlung oder Leistungsoptimierung Workload-Management-Tabellenfunktionen in Verbindung mit den Tabellenfunktionen des Überwachungsprogramms für Momentaufnahme verwenden.

Die Workload-Management-Tabellenfunktionen und die Tabellenfunktionen des Überwachungsprogramms für Momentaufnahme nutzen die folgenden Felder gemeinsam. Sie können Joins auf diesen Feldern ausführen, um Daten abzuleiten, die Sie für Aktivitäten zur Leistungsdiagnose und Leistungsoptimierung benötigen.

Tabelle 15. Von Tabellenfunktionen des Workload-Management und des Überwachungsprogramms für Momentaufnahme gemeinsam genutzte Felder

Feld der Workload-Manager-Tabellenfunktion	Feld der Tabellenfunktion des Überwachungsprogramms für Momentaufnahme
agent_tid	agent_pid
application_handle	agent_id agent_id_holding_lock
session_auth_id	session_auth_id
dbpartitionnum	node_number
utility_id	utility_id
workload_id	workload_id

Nehmen wir als Grund für die Verwendung eines Joins zwischen unterschiedlichen Tabellenfunktionen an, dass Sie Basisinformationen zu allen Dienstprogrammen in der Servicesuperklasse BATCH erhalten wollen. Sie können die folgende Abfrage absetzen:

```
SELECT SUBSTR(UTILITY_TYPE,1,4) TYPE,
       UTILITY_PRIORITY PRIORITY,
       SUBSTR(UTILITY_DESCRIPTION,1,12) AS UTILITY_DESCRIPTION,
       SUBSTR(UTILITY_DBNAME,1,8) AS DBNAME,
       UTILITY_STATE,
       SUBSTR(UTILITY_INVOKER_TYPE,1,7) INVOKER,
       SUBSTR(CHAR(WLM.DBPARTITIONNUM),1,4) PART,
       SUBSTR(CLASSES.PARENTSERVICECLASSNAME,1,19) SUPERCLASS_NAME,
       SUBSTR(CLASSES.SERVICECLASSNAME,1,18) SUBCLASS_NAME
FROM SYSIBMADM.SNAPUTIL SNAP,
     TABLE(WLM.GET_WORKLOAD_OCCURRENCE_ACTIVITIES(CAST(NULL AS BIGINT), -2)) WLM,
     SYSCAT.SERVICECLASSES CLASSES
WHERE SNAP.UTILITY_ID = WLM.UTILITY_ID
      AND WLM.SERVICE_CLASS_ID = CLASSES.SERVICECLASSID
      AND CLASSES.SERVICECLASSNAME = 'SYSDEFAULTSUBCLASS'
      AND CLASSES.PARENTSERVICECLASSNAME = 'BATCH'
ORDER BY WLM.DBPARTITIONNUM;
```

Die Ausgabe könnte folgendermaßen aussehen:

TYPE	PRIORITY	UTILITY_DESCRIPTION	DBNAME	UTILITY_STATE	INVOKER	PART	SUPERCLASS_NAME	SUBCLASS_NAME
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	1	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	1	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	1	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	2	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	2	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	2	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	3	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	3	BATCH	SYSDEFAULTSUBCLASS
LOAD	- OFFLINE	LOAD	SAMPLE	EXECUTE	USER	3	BATCH	SYSDEFAULTSUBCLASS

Gespeicherte Workload-Management-Prozeduren

Sie können gespeicherte Prozeduren für das Abbrechen einer Aktivität, das Erfassen von Details zu einer Aktivität und das Zurücksetzen der Statistikdaten zu Workload-Management-Objekten verwenden.

Die folgenden gespeicherten Prozeduren sind verfügbar:

- `WLM_CANCEL_ACTIVITY(anwendungskennung, uow-id, aktivitäts-id)`. Mit dieser gespeicherten Prozedur können Sie eine momentan ausgeführte oder in der Warteschlange befindliche Aktivität abbrechen. Sie geben die Aktivität durch ihre Anwendungskennung, die UOW-ID und die Aktivitäts-ID an. Sie können jeden Typ von Aktivität abbrechen. Die Anwendung mit der abgebrochenen Aktivität empfängt den Fehler SQL4725N.
- `WLM_CAPTURE_ACTIVITY_IN_PROGRESS(anwendungskennung, uow-id, aktivitäts-id)`. Mit dieser gespeicherten Prozedur können Sie Informationen zu einer einzelnen Aktivität, die momentan ausgeführt wird, an den Aktivitätsereignismonitor senden. Diese gespeicherte Prozedur sendet die Informationen unverzüglich, statt zu warten, bis die Aktivität abgeschlossen ist.

Anmerkung: Wenn Sie diese gespeicherte Prozedur verwenden, um Aktivitätsinformationen für eine Prozedur mit INOUT-Parametern zu erfassen, könnten die INOUT-Werte bis zum Zeitpunkt der Erfassung überschrieben worden sein. Diese Situation tritt nicht auf, wenn Sie die Serviceklasse, die Workload, die Arbeitsaktion oder den prädiktiven Schwellenwert, für die/den Sie Aktivitätsdaten erfassen, als `COLLECT ACTIVITY WITH DETAILS AND VALUES` erstellt haben oder wenn Sie die Serviceklasse, die Workload, die Arbeitsaktion oder den prädiktiven Schwellenwert ändern und das Schlüsselwort `COLLECT ACTIVITY DATA` mit dem Schlüsselwort `ON COORDINATOR` oder mit dem Schlüsselwort `ON ALL` sowie mit dem Schlüsselwort `WITH DETAILS AND VALUES` angeben.

- `WLM_COLLECT_STATS()`. Mit dieser gespeicherten Prozedur erfassen Sie Statistikdaten für Workload-Management-Objekte und setzen diese zurück. Alle für Serviceklassen, Workloads, Schwellenwertwarteschlangen und Arbeitsaktionssets verfolgten Statistikdaten werden an den aktiven Statistikereignismonitor (sofern vorhanden) gesendet und zurückgesetzt. Wenn kein aktiver Statistikereignismonitor vorhanden ist, werden die Statistikdaten nur zurückgesetzt, jedoch nicht erfasst.

Workload-Management-Ereignismonitore

Ereignismonitore erfassen Langzeitinformationen oder eine Reihe von Ereignissen für das Debugging. Tabellenfunktionen dagegen erfassen und melden punktuelle Informationen.

Sie können die folgenden Typen von Ereignismonitoren in einer Workload-Management-Konfiguration verwenden:

- **ACTIVITIES.** Dieser Typ von Ereignismonitor erfasst Informationen zu einzelnen Aktivitäten. In einigen Situationen können Sie festlegen, dass der Ereignismonitor Anweisungsinformationen und die für die Eingabevariablen für SQL-Aktivitäten verwendeten Datenwerte ebenfalls erfassen soll. Sie können die vom Aktivitätsereignismonitor erfassten Aktivitäten als Eingabe in Tools wie `db2advis` verwenden. Außerdem können Sie mit einem **ACTIVITIES**-Ereignismonitor Informationen für das Debugging einzelner Aktivitäten erfassen.

Sie können Informationen zu einer Aktivität erfassen, indem Sie `COLLECT ACTIVITY DATA` für die Serviceklasse, Workload oder Arbeitsaktion angeben, zu der eine solche Aktivität gehört, oder für einen Schwellenwert, gegen den eine solche Aktivität verstoßen könnte. Die Informationen werden bei Abschluss der Aktivität erfasst, unabhängig davon, ob die Aktivität erfolgreich abgeschlossen wurde.

- **THRESHOLD VIOLATIONS.** Dieser Typ von Ereignismonitor erfasst Informationen bei jedem von einer Aktivität ausgelösten Schwellenwertverstoß. Die Informationen umfassen die ID, die UOW (Unit of Work - Arbeitseinheit) und die Anwendungskennung, um die Aktivität eindeutig zu bezeichnen, die den

Schwellenwertverstoß verursachte, sowie die Aktion, die auf die Aktivität angewendet wurde (STOP EXECUTION oder CONTINUE).

Wenn Sie COLLECT ACTIVITY DATA für den Schwellenwert angeben und ein Aktivitätsereignismonitor erstellt und aktiv ist, werden auch Informationen zu Aktivitäten erfasst, die gegen den Schwellenwert verstoßen. Diese Informationen werden erfasst, wenn die Aktivität (erfolgreich oder nicht erfolgreich) endet.

Sie können Details zu einem Schwellenwert durch Abfrage der Sicht SYSCAT.THRESHOLDS erhalten.

- **STATISTICS.** Dieser Typ von Ereignismonitor erfasst Statistikdaten, die über einen bestimmten Zeitraum gemessen werden. Im Vergleich zu Anweisungs- oder Aktivitätsereignismonitoren ist der STATISTICS-Ereignismonitor eine weniger aufwendige Methode zur Erfassung von Langzeitinformatoren, da dieser Typ von Ereignismonitor die zusammengefassten Aktivitätsinformationen statt die einzelnen Aktivitäten verwendet und Sie eine einzige Serviceklasse oder Arbeitsklasse als Ziel für diesen Monitor definieren können. Eine Beschreibung zum Senden von Statistikdaten an den Ereignismonitor finden Sie in „Erfassen von Workload-Management-Statistikdaten mit Hilfe eines Ereignismonitors für Statistikdaten“ auf Seite 137.

Im Gegensatz zu Anweisungs-, Verbindungs- und Transaktionsereignismonitoren, haben die Aktivitäts-, Statistik- und Schwellenwertverstoß-Ereignismonitore keine Ereignisbedingungen (d. h. Bedingungen, die im Schlüsselwort WHERE der Anweisung CREATE EVENT MONITOR angegeben werden). Diese Ereignismonitore sind statt dessen auf die Attribute von Serviceklassen, Workloads, Arbeitsklassen und Schwellenwerten angewiesen, um zu ermitteln, ob diese Objekte ihre Aktivitätsinformationen oder zusammengefassten Informationen an diese Monitore senden.

Typischerweise schreiben Ereignismonitore Daten entweder in Tabellen oder in Dateien. Sie müssen diese Tabellen oder Dateien regelmäßig bereinigen, da dies nicht automatisch erfolgt.

Mit dem Script wlmevmon.ddl im Verzeichnis sqllib/misc können Sie drei Ereignismonitore namens DB2ACTIVITIES, DB2STATISTICS und DB2THRESHOLDVIOLATIONS erstellen und aktivieren. Ändern Sie gegebenenfalls das Script, um den Tabellenbereich oder andere Parameter zu ändern.

Statistikmanagement

Statistikdaten für Workload-Management-Objekte

Für Workload-Management-Objekte, einschließlich Serviceklassen, Arbeitsklassen, Workloads und Schwellenwertwarteschlangen, werden Statistikdaten erfasst. Diese Statistikdaten befinden sich im Hauptspeicher und können in Echtzeit mit Tabellenfunktionen für Workload-Management-Statistiken angezeigt werden. Die Statistikdaten können auch erfasst und an einen Statistikereignismonitor gesendet werden, wo sie später für die Langzeitanalyse angezeigt werden können.

Wenn Statistikdaten an den Ereignismonitor gesendet werden, werden die Werte im Speicher zurückgesetzt, um zu verhindern, dass die Daten in anschließenden Erfassungsintervallen doppelt erfasst werden. Da die Tabellenfunktionen für Workload-Management-Statistiken die Werte melden, die sich momentan im Speicher befinden, werden nach einer Sammlung die zurückgesetzten Werte gemeldet. Die Workload-Management-Tabellenfunktionen melden nur eine Teilmenge der

Statistikdaten. Wenn Sie alle Statistikdaten anzeigen wollen, müssen Sie die Statistikdaten erfassen und sie an einen Statistikereignismonitor senden.

Die folgenden Statistikdaten werden für die gegebenen Objekte in jeder Datenbankpartition erfasst, unabhängig vom Wert der Option COLLECT AGGREGATE ACTIVITY DATA oder der Option COLLECT AGGREGATE REQUEST DATA, die für diese Objekte bei ihrer Erstellung oder Änderung angegeben wird.

- Schwellenwertwarteschlangen:
 - Warteschlangenzuweisungen insgesamt (**queue_assignments_total**). Mit Hilfe dieser Statistik können Sie ermitteln, ob eine zu starke Warteschlangennutzung auftritt oder ob die korrekte Anzahl von Aktivitäten in die Warteschlange gestellt wird (d. h. ob der Schwellenwert für gemeinsamen Zugriff zu restriktiv oder nicht restriktiv genug ist).
 - Max. Warteschlangengröße (**queue_size_top**). Mit Hilfe dieser Statistik können Sie die maximale Warteschlangengröße ermitteln und feststellen, ob die Warteschlangengröße ausreichend ist.
 - Wartezeit insgesamt (**queue_time_total**). Mit Hilfe dieser Statistik können Sie ermitteln, wieviel Zeit Aktivitäten in der Warteschlange verbringen und ob diese Zeit zu lang ist.
- Serviceunterklassen:
 - Max. gleichzeitig ablaufende Aktivitäten (**concurrent_act_top**). Mit Hilfe dieser Statistik können Sie den höchsten gemeinsamen Zugriff von Aktivitäten (einschließlich verschachtelter Aktivitäten) ermitteln, die in einer Datenbankpartition für eine Serviceklasse in dem Zeitintervall erreicht wird, für das die Statistikdaten erfasst werden.
 - Abgeschlossene Koordinatoraktivitäten insgesamt (**coord_act_completed_total**). Mit Hilfe dieser Statistik können Sie ermitteln, wieviel Arbeit in einer Serviceklasse ausgeführt wird.
 - Abgebrochene Koordinatoraktivitäten insgesamt (**coord_act_aborted_total**). Mit Hilfe dieser Statistik, die den nicht erfolgreichen Abschluss von Aktivitäten misst, können Sie den Zustand des Systems ermitteln. Aktivitäten können auf Grund von Abbruchbefehlen, Fehlern oder reaktiven Schwellenwerten abgebrochen werden.
 - Zurückgewiesene Koordinatoraktivitäten insgesamt (**coord_act_rejected_total**). Mit Hilfe dieser Statistik, die die Zurückweisung von Aktivitäten misst, können Sie einen Hinweis auf die Zweckmäßigkeit der Zurückweisungsrichtlinie erhalten. Aktivitäten werden als 'Zurückgewiesen' gewertet, wenn sie gegen einen prädiktiven Schwellenwert verstoßen haben, für den die Aktion STOP EXECUTION definiert wurde, oder wenn sie an der Ausführung durch eine Arbeitsaktion gehindert werden.
 - Anzahl aktiver Anforderungen (**num_requests_active**). Mit Hilfe dieser Statistik können Sie die Anzahl der Anforderungen ermitteln, die momentan in einer Serviceklasse ausgeführt werden.
- Servicesuperklassen:
 - Max. gleichzeitig bestehende Verbindungen (**concurrent_connection_top**). Mit Hilfe dieser Statistik können Sie den Schwellenwert für gemeinsamen Zugriff für eine Verbindung optimieren.
- Workloads:
 - Abgeschlossene Workloadvorkommen insgesamt (**wlo_completed_total**). Mit Hilfe dieser Statistik können Sie feststellen, wieviele Vorkommen einer Workload in einem bestimmten Zeitraum abgeschlossen werden.

- Max. gleichzeitig ablaufende Workloadvorkommen (**concurrent_wlo_top**). Mit Hilfe dieser Statistik können Sie die maximale Anzahl gleichzeitig ablaufender Workloadvorkommen bestimmen und den Schwellenwert für gemeinsamen Zugriff eines Workloadvorkommens einrichten oder optimieren, wenn die Anzahl der momentan gleichzeitig ablaufenden Workloadvorkommen zu hoch ist (d. h. zu viele Anwendungen, die derselben Workloaddefinition zugeordnet sind, gleichzeitig auf dem System ausgeführt werden).
- Max. gleichzeitig ablaufende Aktivitäten (**concurrent_act_top**). Mit Hilfe dieser Statistik können Sie den Schwellenwert CONCURRENTWORKLOADACTIVITIES optimieren.
- Abgeschlossene Koordinatoraktivitäten insgesamt (**coord_act_completed_total**). Mit Hilfe dieser Statistik, die den erfolgreichen Abschluss von Aktivitäten misst, können Sie einen Hinweis auf den Status des Systems erhalten.
- Abgebrochene Koordinatoraktivitäten insgesamt (**coord_act_aborted_total**). Mit Hilfe dieser Statistik, die den nicht erfolgreichen Abschluss von Aktivitäten misst, können Sie den Zustand des Systems ermitteln. Aktivitäten können auf Grund eines Abbruchbefehls, von Fehlern oder von reaktiven Schwellenwerten abgebrochen werden.
- Zurückgewiesene Koordinatoraktivitäten insgesamt (**coord_act_rejected_total**). Mit Hilfe dieser Statistik, die die Zurückweisung von Aktivitäten misst, können Sie die Zweckmäßigkeit einer Zurückweisungsrichtlinie ermitteln. Aktivitäten werden als 'Zurückgewiesen' gewertet, wenn sie gegen einen prädiktiven Schwellenwert verstoßen haben, für den die Aktion STOP EXECUTION definiert wurde, oder wenn sie an der Ausführung durch eine Arbeitsaktion gehindert werden.
- Abgeschlossene Workloadvorkommen insgesamt (**wlo_completed_total**). Mit Hilfe dieser Statistik können Sie feststellen, wieviele Vorkommen einer Workload in einem bestimmten Zeitraum abgeschlossen werden.
- Arbeitsklasse (über eine Arbeitsaktion):
 - Aktivitäten insgesamt (**act_total**). Mit Hilfe dieser Statistik können Sie die Effektivität des Arbeitsaktionssets bestimmen und die relativen Prozentsätze der Typen von Aktivitäten im System ermitteln.

Wenn Sie den Wert der Option COLLECT AGGREGATE REQUEST DATA für eine Serviceunterklasse auf BASE einstellen, werden die folgenden Statistikdaten für die Serviceunterklasse erfasst:

- Histogramm der Anforderungsausführungszeit. Die Anforderungsausführungszeit wird in einem Histogramm für jede Datenbankpartition und für alle Anforderungen erfasst. Die Anforderungsausführungszeit gibt näherungsweise den Aufwand von Agenten an, die an Aktivitäten arbeiten (die aus mindestens einer Anforderung bestehen). Anhand dieser Informationen können Sie feststellen, wo Arbeit ausgeführt wird und ob die Verteilung von Arbeit über Partitionen hinweg einheitlich ist. (Koordinatoraktivitätszähler zeigen z. B., dass die meisten Aktivitäten von einer Datenbankpartition ausgehen, der Koordinatoragent jedoch im Rahmen der Verarbeitung der Aktivitäten Anforderungen an eine andere Datenbankpartition sendet, die die meiste Arbeit ausführt.) Das Histogramm der Anforderungsausführungszeit kann hilfreich sein bei der Ermittlung der Größe von Anforderungen, die an eine Datenbankpartition gesendet werden (d. h. ob die Arbeit, die an die Datenbankpartition gesendet wird, im Wesentlichen aus kleinen oder aus großen Anforderungen besteht oder ob es keine bestimmte Verteilung gibt).
- Durchschnittliche Anforderungsausführungszeit (**request_exec_time_avg**). Mit Hilfe dieser Statistik können Sie schnell die durchschnittliche Zeitdauer für die Verarbeitung der einzelnen Anforderungen auf einer Datenbankpartition ermit-

teln und die Histogrammschablone für das entsprechende Histogramm der Anforderungsausführungszeit optimieren.

Wenn Sie den Wert der Option COLLECT AGGREGATE ACTIVITY DATA für eine Serviceunterklasse oder eine Arbeitsklasse (über eine Arbeitsaktion) auf BASE einstellen, werden für jede Datenbankpartition für die entsprechende Serviceklasse oder Arbeitsklasse die folgenden Statistikdaten erfasst oder Histogramme generiert. Anhand der Durchschnittswerte können Sie schnell erkennen, wo Aktivitäten ihre meiste Zeit (z. B. in der Warteschlange oder bei der Ausführung) verbringen, und die Antwortzeit (Laufzeit) ermitteln. Sie können die Durchschnittswerte auch zur Optimierung der Histogrammschablonen verwenden. Sie können also den tatsächlichen Durchschnitt mit dem vom Histogramm berechneten Durchschnitt vergleichen und bei einer Abweichung der Werte die Änderung der Histogrammschablone für das entsprechende Histogramm unter Verwendung einer Gruppe von Binwerten in Erwägung ziehen, die für die Daten besser geeignet sind.

- Durchschnittliche Laufzeit von Koordinatoraktivitäten (**coord_act_lifetime_avg**). Mit Hilfe dieser Statistik können Sie das arithmetische Mittel der Laufzeit für unverschachtelte Koordinatoraktivitäten ermitteln, die einer Serviceklasse oder einer Arbeitsklasse zugeordnet sind.
- Durchschnittliche Ausführungszeit von Koordinatoraktivitäten (**coord_act_exec_avg**). Mit Hilfe dieser Statistik können Sie das arithmetische Mittel der Ausführungszeit für unverschachtelte Koordinatoraktivitäten ermitteln, die einer Serviceklasse oder einer Arbeitsklasse zugeordnet sind.
- Durchschnittliche Wartezeit von Koordinatoraktivitäten (**coord_act_queue_avg**). Mit Hilfe dieser Statistik können Sie das arithmetische Mittel der Wartezeit für unverschachtelte Koordinatoraktivitäten ermitteln, die einer Serviceklasse oder einer Arbeitsklasse zugeordnet sind.
- Max. geschätzter Aufwand (**cost_estimate_top**). Mit Hilfe dieser Statistik können Sie die Schwellenwerte für den geschätzten Aufwand optimieren.
- Max. geschätzte zurückgegebene Zeilen (**rows_returned_top**). Mit Hilfe dieser Statistik können Sie die Schwellenwerte für die tatsächlich zurückgegebenen Zeilen optimieren.
- Max. Tabellenbereich für temporäre Tabellen (**temp_tablespace_top**). Mit Hilfe dieser Statistik können Sie die Schwellenwerte für die Nutzung des Tabellenbereichs für temporäre Tabellen optimieren. Diese Statistikdaten werden nur überwacht, wenn Sie einen Schwellenwert für die Nutzung des Tabellenbereichs für temporäre Tabellen definieren.
- Histogramm der Aktivitätslaufzeit. Dieses Histogramm erfasst die Zeitdifferenz zwischen dem Eingang einer Aktivität und der Endzeit für unverschachtelte Koordinatoraktivitäten. Mit Hilfe dieses Histogramms können Sie eine Ansicht der Gesamtsystemleistung erhalten. Wenn die Aktivität eine Routine ist, die nach Beendigung einen Cursor offen lässt, zählt das Laufzeithistogramm die Laufzeit des Cursors nicht zur Laufzeit der Routine, die das übergeordnete Objekt des Cursors ist.
- Histogramm der Aktivitätsausführungszeit. Dieses Histogramm erfasst die Ausführungszeit für unverschachtelte Koordinatoraktivitäten. Mit Hilfe dieses Histogramms können Sie die Auswirkung von Systemänderungen auf die Ausführungszeit messen. Die Ausführungszeit wird wie folgt berechnet:
 - Bei einem Cursor ist die Ausführungszeit die kombinierte Zeit für die Anforderung zum Öffnen des Cursors, für alle Abrufe und für die Anforderung zum Schließen des Cursors. Zeit, während der der Cursor inaktiv ist, wird nicht zur Ausführungszeit gerechnet.

- Bei Routinen ist die Ausführungszeit die Zeit vom Start bis zum Ende des Routinenaufrufs. Wenn Cursor nach der Beendigung der Routine offen bleiben, werden die Laufzeiten dieser Cursor nicht zur Ausführungszeit der Routine gerechnet.
- Bei allen anderen Aktivitäten ist die Ausführungszeit die Differenz zwischen der Aktivitätslaufzeit und der Zeit, während der die Aktivität in der Warteschlange wartet.
- Histogramm der Aktivitätswartezeit. Dieses Histogramm erfasst die Zeitdauer, während der unverschachtelte Koordinatoraktivitäten in der Warteschlange verbleiben. Mit Hilfe dieses Histogramms können Sie die Auswirkung von Warteschlangenschwellenwerten auf Aktivitäten messen.

Wenn Sie den Wert der Option COLLECT AGGREGATE ACTIVITY DATA für eine Serviceunterklasse oder eine Arbeitsklasse auf EXTENDED einstellen, werden für jede Datenbankpartition für die entsprechende Serviceklasse oder Arbeitsklasse (über eine Arbeitsaktion) die folgenden Systemstatistikdaten erfasst oder Histogramme generiert. Mit Hilfe der Durchschnittswerte können Sie schnell die durchschnittliche Eingangsrate von Aktivitäten (Eingangsrate ist das Gegenteil der Zwischenankunftszeit (Inter-Arrival Time)) und den Aufwand von Aktivitäten (geschätzter Aufwand) ermitteln. Sie können die Durchschnittswerte auch zur Optimierung der Histogrammschablonen verwenden. Sie können also den tatsächlichen Durchschnitt mit dem vom Histogramm berechneten Durchschnitt vergleichen und bei einer Abweichung der Werte die Änderung der Histogrammschablone für das entsprechende Histogramm unter Verwendung einer Gruppe von Binwerten in Erwägung ziehen, die für die Daten besser geeignet sind.

Anmerkung: EXTENDED-Statistikdaten sind hilfreich für eine detailliertere Leistungsmodellierung. Weitere Informationen finden Sie in „Modellierung der Workload-Management-Leistung“ auf Seite 147.

- Durchschnittliche Zwischenankunftszeit von unverschachtelten Koordinatoraktivitäten (**coord_act_interarrival_time_avg**). Mit Hilfe dieser Statistik können Sie das arithmetische Mittel der Zeit zwischen dem Eingang einer Koordinatoraktivität auf Verschachtelungsebene 0, die dieser Serviceklasse oder Arbeitsklasse zugeordnet ist, und der nächsten eingehenden Koordinatoraktivität ermitteln. Der Durchschnittswert wird seit der letzten Zurücksetzung der Statistikdaten berechnet.
- Durchschnittlicher geschätzter Aufwand für Koordinatoraktivitäten (**coord_act_est_cost_avg**). Mit Hilfe dieser Statistikdaten können Sie das arithmetische Mittel des geschätzten Aufwands von DML-Koordinatoraktivitäten auf Verschachtelungsebene 0, die dieser Serviceunterklasse oder Arbeitsklasse zugeordnet sind, seit der letzten Zurücksetzung der Statistikdaten ermitteln.
- Histogramm der Zwischenankunftszeit von Aktivitäten. Dieses Histogramm erfasst die Zwischenankunftszeit für unverschachtelte Koordinatoraktivitäten. Mit Hilfe dieses Histogramms können Sie die Verteilung der Zwischenankunftszeiten für unverschachtelte Koordinatoraktivitäten erhalten. Diese Daten sind für die Modellierung Ihres Systems oder für die Eingabe in Anwendungen zur Leistungsmodellierung hilfreich.
- Histogramm des geschätzten Aufwands für Aktivitäten. Dieses Histogramm erfasst den geschätzten Aufwand für unverschachtelte Koordinatoraktivitäten. Mit Hilfe dieses Histogramms erhalten Sie eine näherungsweise berechnete Servicezeitverteilung. Diese Daten sind für die Modellierung Ihres Systems oder für die Eingabe in Anwendungen zur Leistungsmodellierung hilfreich.

Die folgende Tabelle bietet eine Kurzübersicht der für jedes Workload-Management-Objekt erfassten Statistikdaten. Einige Statistikdaten werden immer für jedes Objekt erfasst. Andere Statistikdaten werden nur erfasst, wenn eine bestimmte Option COLLECT AGGREGATE angegeben wird. Wenn bei zusammengefassten Aktivitätsstatistiken COLLECT AGGREGATE ACTIVITY DATA EXTENDED angegeben wird, werden auch alle zusammengefassten Aktivitätsdaten vom Typ BASE erfasst.

Tabelle 16. Für jedes Workload-Management-Objekt erfasste Statistikdaten

Objekttyp	Standardmäßig erfasste Statistikdaten	Bei Angabe von COLLECT AGGREGATE ACTIVITY DATA BASE erfasste Statistikdaten	Bei Angabe von COLLECT AGGREGATE ACTIVITY DATA EXTENDED erfasste Statistikdaten	Bei Angabe von COLLECT AGGREGATE REQUEST DATA BASE erfasste Statistikdaten
Serviceunterklasse	<ul style="list-style-type: none"> • coord_act_completed_total • coord_act_rejected_total • coord_act_aborted_total • concurrent_act_top • num_requests_active 	<ul style="list-style-type: none"> • cost_estimate_top • rows_returned_top • temp_tablespace_top • coord_act_lifetime_top • request_exec_time_avg • coord_act_lifetime_avg • coord_act_exec_time_avg • coord_act_queue_time_avg • Histogramm der Aktivitätslaufzeit • Histogramm der Aktivitätsausführungszeit • Histogramm der Aktivitätswartezeit • Histogramm der Anforderungsausführungszeit 	<ul style="list-style-type: none"> • coord_act_est_cost_avg • coord_act_interarrival_time_avg • Histogramm der Zwischenankunftszeit von Aktivitäten • Histogramm des geschätzten Aufwands für Aktivitäten 	<ul style="list-style-type: none"> • request_exec_time_avg • Histogramm der Anforderungsausführungszeit
Servicesuperklasse	<ul style="list-style-type: none"> • concurrent_connection_top 	Nicht zutreffend	Nicht zutreffend	
Workload	<ul style="list-style-type: none"> • concurrent_wlo_top • concurrent_act_top • coord_act_completed_total • coord_act_rejected_total • coord_act_aborted_total • wlo_completed_total 	Nicht zutreffend	Nicht zutreffend	
Arbeitsklasse (über eine Arbeitsaktion)	<ul style="list-style-type: none"> • act_total 	<ul style="list-style-type: none"> • cost_estimate_top • rows_returned_top • temp_tablespace_top • coord_act_lifetime_top • coord_act_lifetime_avg • coord_act_exec_time_avg • coord_act_queue_time_avg • Histogramm der Aktivitätslaufzeit • Histogramm der Aktivitätsausführungszeit • Histogramm der Aktivitätswartezeit 	<ul style="list-style-type: none"> • coord_act_est_cost_avg • coord_act_interarrival_time_avg • Histogramm der Zwischenankunftszeit von Aktivitäten • Histogramm des geschätzten Aufwands für Aktivitäten 	
Schwellenwert	Nicht zutreffend	Nicht zutreffend	Nicht zutreffend	

Histogramme beim Workload-Management

Ein *Histogramm* ist eine Sammlung von Bins (Kategorien von Werten), die Container für die Sammlung von getrennten Datenbereichen sind. Histogramme sind nützlich für eine Vielzahl von Aufgaben bei der Workloadanalyse und der Leistungsoptimierung.

DB2-Workload-Management-Histogramme enthalten 41 Bins. Die Anzahl der Bins ist fest. Der 40. Bin enthält den höchsten definierten Wert für das Histogramm. Der 41. Bin dient zur Aufnahme von Werten, die über dem höchsten definierten Wert liegen. Die folgende Abbildung zeigt ein Histogramm der Aktivitätslaufzeiten, dargestellt als Balkendiagramm:

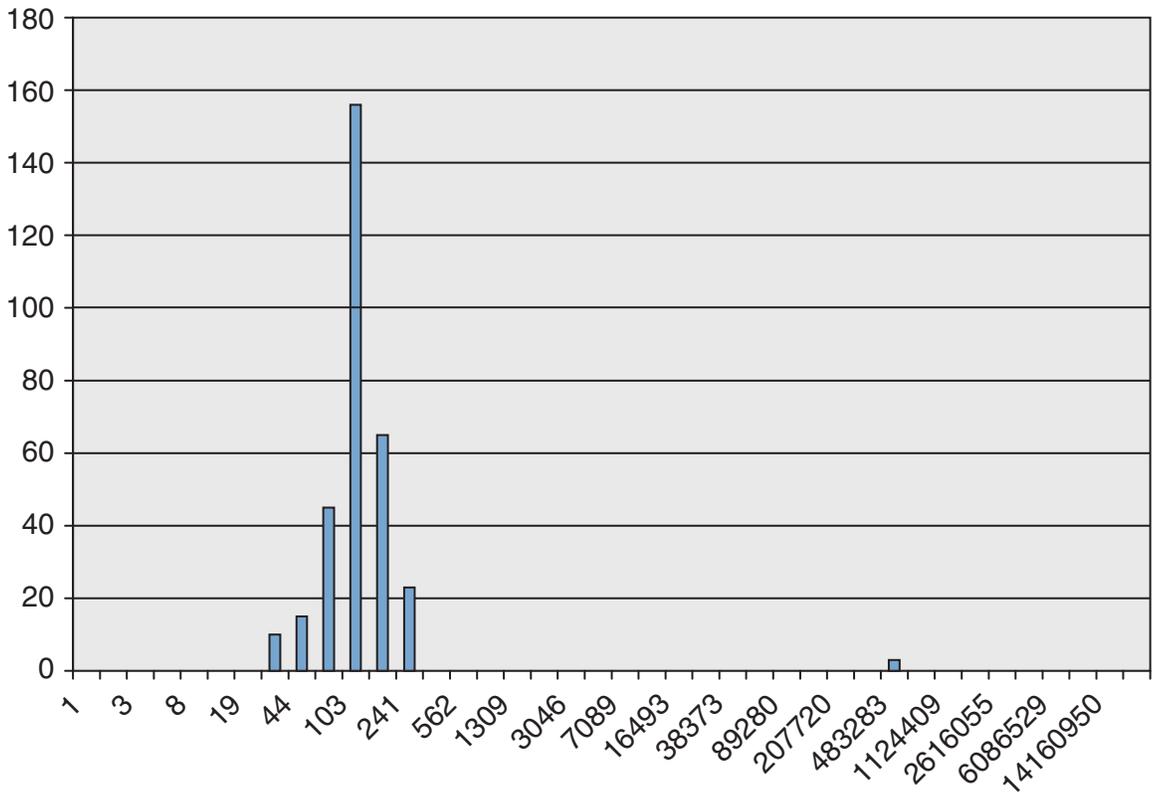


Abbildung 24. Histogramm als Balkendiagramm

Das grafisch dargestellte Histogramm der Aktivitätslaufzeit entspricht den folgenden Daten. Jeder Zähler stellt die Anzahl von Aktivitäten dar, deren Laufzeit (in Millisekunden) in den Bereich zwischen dem unteren Binwert und dem oberen Binwert fiel. 156 Aktivitäten z. B. hatten eine Laufzeit im Bereich von 68 Millisekunden bis 103 Millisekunden.

Unterer Bin	Oberer Bin	Anzahl
0	1	0
1	2	0
2	3	0
3	5	0
5	8	0
8	12	0
12	19	0
19	29	10
29	44	15
44	68	45
68	103	156
103	158	65
158	241	23
241	369	0
369	562	0
562	858	0
858	1309	0
1309	1997	0
1997	3046	0
3046	4647	0
4647	7089	0
7089	10813	0
10813	16493	0
16493	25157	0
25157	38373	0
38373	58532	0

58532	89280	0
89280	136181	0
136181	207720	0
207720	316840	0
316840	483283	3
483283	737162	0
737162	1124409	0
1124409	1715085	0
1715085	2616055	0
2616055	3990325	0
3990325	6086529	0
6086529	9283913	0
9283913	14160950	0
14160950	21600000	0
21600000	Unendlich	0

Sie können Histogramme für eine Reihe unterschiedlicher Zwecke verwenden. Sie können mit ihnen beispielsweise die Verteilung von Werten anzeigen, um Ausreißerwerte zu erkennen oder um Durchschnittswerte und Standardabweichungen zu berechnen. Beispiele für die Verwendung von Histogrammen zum besseren Verständnis und zur besseren Beschreibung Ihrer Workload finden Sie in „Beispiel: Optimieren einer Workload-Management-Konfiguration, wenn keine Kapazitätsplanungsinformationen verfügbar sind“ auf Seite 195 und „Beispiel: Berechnen von Durchschnittswerten und einer Standardabweichung anhand von Histogrammen in einer Workload-Management-Konfiguration“ auf Seite 182.

In einer Umgebung mit partitionierten Datenbanken werden Histogramme auf jeder Datenbankpartition erfasst. Bins in Histogrammen haben auf allen Datenbankpartitionen dieselben Werte. Mit Hilfe der Bins können Sie Informationen auf der Basis einzelner Partitionen analysieren. Sie können auch die Histogramme aller Datenbankpartitionen kombinieren, indem Sie die Anzahlen in den entsprechenden Bins in einem einzigen Histogramm addieren, um eine globale Ansicht der Daten zu erhalten, die Sie dann für Aufgaben wie das Berechnen des globalen Durchschnitts und der globalen Standardabweichung aus dem globalen Histogramm verwenden können.

Histogramme sind für Serviceunterklassen und Arbeitsklassen (über Arbeitsaktionen) verfügbar. Für diese Objekte werden Histogramme erfasst, wenn Sie beim Erstellen oder Ändern des Objekts eine der Klauseln COLLECT AGGREGATE ACTIVITY DATA angeben. Für Arbeitsklassen werden Histogramme erfasst, wenn eine Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA auf die Arbeitsklasse angewendet wird. Die folgenden Histogramme sind verfügbar:

- Laufzeit unverschachtelter Koordinatoraktivitäten (wenn Sie AGGREGATE ACTIVITY DATA BASE oder AGGREGATE ACTIVITY DATA EXTENDED für die Serviceunterklasse oder für eine auf die Arbeitsklasse angewendete Arbeitsaktion angeben).
- Ausführungszeit unverschachtelter Koordinatoraktivitäten (wenn Sie AGGREGATE ACTIVITY DATA BASE oder AGGREGATE ACTIVITY DATA EXTENDED für die Serviceunterklasse oder für eine auf die Arbeitsklasse angewendete Arbeitsaktion angeben).
- Wartezeit unverschachtelter Koordinatoraktivitäten (wenn Sie AGGREGATE ACTIVITY DATA BASE oder AGGREGATE ACTIVITY DATA EXTENDED für die Serviceunterklasse oder für eine auf die Arbeitsklasse angewendete Arbeitsaktion angeben).
- Anforderungsausführungszeit (wenn Sie AGGREGATE REQUEST DATA BASE für die Serviceklasse angeben). Dieses Histogramm findet keine Anwendung bei Arbeitsklassen.

- Histogramm der Zwischenankunftszeit für unverschachtelte Aktivitäten (wenn Sie AGGREGATE ACTIVITY DATA EXTENDED für die Serviceunterklasse oder für eine auf die Arbeitsklasse angewendete Arbeitsaktion angeben).
- Geschätzter Aufwand unverschachtelter DML-Aktivitäten (wenn Sie AGGREGATE ACTIVITY DATA EXTENDED für die Serviceunterklasse oder für eine auf die Arbeitsklasse angewendete Arbeitsaktion angeben).

Alle aktivitätsbezogenen Histogramme erfassen Aktivitäten, die abgeschlossen, abgebrochen oder zurückgewiesen werden.

Sie können optional eine Histogrammschablone angeben, die die oberen Binwerte für jedes der Histogramme beschreibt, die für ein Objekt erfasst werden. Histogrammschablonen werden verwendet, um die oberen Binwerte für ein Histogramm zu ermitteln. Histogrammschablonen sind *einheitenlose* Objekte, die angeben, wie ein bestimmtes Histogramm aussehen sollte (einheitenlos bedeutet, dass der Histogrammschablone keine vordefinierte Maßeinheit zugewiesen ist).

Sie wenden eine Histogrammschablone mit dem entsprechenden Schlüsselwort HISTOGRAM TEMPLATE an, wenn Sie Serviceunterklassen oder Arbeitsaktionen erstellen oder ändern. Wenn Sie keine Histogrammschablone haben, wird die Standardschablone SYSDEFAULTHISTOGRAM verwendet. Wenn die Erfassung mit AGGREGATE ACTIVITY DATA für ein Objekt nicht aktiviert ist, wird die Histogrammschablone ignoriert.

Sie können eine Histogrammschablone erstellen, indem Sie die Anweisung CREATE HISTOGRAM TEMPLATE verwenden und den maximalen oberen Binwert angeben. Alle anderen Bins werden automatisch als exponentiell erhöhte Werte definiert, die sich dem oberen Binwert annähern. Wenn Sie z. B. eine Histogrammschablone mit einem oberen Binwert von 3 000 000 erstellen wollen, würden Sie eine Anweisung wie die folgende absetzen:

```
CREATE HISTOGRAM TEMPLATE TEMPLATE1 HIGH BIN VALUE 3000000
```

Diese Anweisung erstellt eine Histogrammschablone mit den folgenden Binwerten:

Unterer Bin	Oberer Bin
0	1
1	2
2	3
3	4
4	6
6	9
9	13
13	19
19	28
28	41
41	60
60	87
87	127
127	184
184	268
268	389
389	565
565	821
821	1192
1192	1732
1732	2514
2514	3651
3651	5300
5300	7696
7696	11173
11173	16222

16222	23553
23553	34196
34196	49649
49649	72084
72084	104657
104657	151948
151948	220609
220609	320297
320297	465030
465030	675163
675163	980250
980250	1423197
1423197	2066299
2066299	3000000
3000000	Unendlich

Sie können dann diese Histogrammschablone für ein vorhandenes Histogramm verwenden. Wenn Sie z. B. die Histogrammschablone TEMPLATE1 für das Histogramm der Aktivitätslaufzeit der Serviceunterklasse MYSUBCLASS unter der Servicesuperklasse MYSUPERCLASS verwenden wollen, setzen Sie eine Anweisung wie die folgende ab:

```
ALTER SERVICE CLASS MYSUBCLASS UNDER MYSUPERCLASS ACTIVITY LIFETIME HISTOGRAM
TEMPLATE TEMPLATE1
```

Nachdem Sie ein Commit für die Anweisung ALTER SERVICE CLASS durchgeführt haben, hat das Histogramm der Aktivitätslaufzeit, das für diese Serviceunterklasse erfasst wird, obere Binwerte, die von der Histogrammschablone TEMPLATE1 bestimmt sind, statt die Standardwerte für die oberen Binwerte aus der Histogrammschablone SYSDEFAULTHISTOGRAM zu verwenden.

Anmerkung: Eine Änderung der Angabe der Histogrammschablone für eine Serviceklasse oder die Änderung einer Histogrammschablone wird erst nach der Zurücksetzung der Statistikdaten wirksam.

Sie können eine Histogrammschablone mit der Anweisung DROP HISTOGRAM TEMPLATE löschen.

Sie können die Histogrammschablonen durch Abfrage der Sicht SYSCAT.HISTOGRAMTEMPLATES und die entsprechenden oberen Binwerte der Histogrammschablone durch Abfrage der Sicht SYSCAT.HISTOGRAMTEMPLATEBINS anzeigen (der untere Binwert ist immer der obere Binwert des vorherigen Bins bzw. 0 für den ersten Bin).

Erfassen von Workload-Management-Statistikdaten mit Hilfe eines Ereignismonitors für Statistikdaten

Statistikdaten für Workload-Management-Objekte können für eine Langzeitanalyse an einen Statistikereignismonitor gesendet werden.

Anhand von Statistikdaten können Sie das Verhalten Ihres Systems im Lauf der Zeit erkennen (z. B. die durchschnittliche Laufzeit von Aktivitäten, die Wartezeit von Aktivitäten in der Warteschlange, die Verteilung von großen und kleinen Aktivitäten usw.), Schwellenwerte festlegen (z. B. die Obergrenze für gleichzeitig ablaufende Aktivitäten finden) und Probleme erkennen (z. B. ob die von Benutzern erlebte durchschnittliche Laufzeit höher als normal ist). Eine Beschreibung der für jedes Workload-Management-Objekt erfassten Statistikdaten finden Sie in „Statistikdaten für Workload-Management-Objekte“ auf Seite 128.

Sie können automatisch Workload-Management-Statistikdaten an einen Ereignismonitor entweder in einem festen Zeitintervall oder manuell zu einem beliebigen Zeitpunkt senden.

Gehen Sie wie folgt vor, um automatisch Workload-Management-Statistikdaten in einem festen Zeitintervall zu erfassen:

1. Erstellen Sie mit der Anweisung `CREATE EVENT MONITOR` einen `STATISTICS`-Ereignismonitor. Sie könnten z. B. die folgende Anweisung absetzen:
`CREATE EVENT MONITOR STATS1 FOR STATISTICS WRITE TO TABLE`
2. Verwenden Sie die Anweisung `COMMIT`, um die Änderungen festzuschreiben.
3. Aktivieren Sie den Ereignismonitor mit Hilfe der Anweisung `SET EVENT MONITOR STATE`. Statt der Anweisung `SET EVENT MONITOR STATE` können Sie auch die Standardeinstellung `AUTOSTART` für den `STATISTICS`-Ereignismonitor verwenden, um ihn bei der nächsten Aktivierung der Datenbank aktivieren zu lassen. Es kann jedoch jeweils nur ein Ereignismonitor vom Typ `STATISTICS` auf einer Datenbankpartition aktiv sein. Wenn Sie mehrere `STATISTICS`-Ereignismonitore definieren wollen, sollten Sie nicht die Option `AUTOSTART` verwenden.

4. Verwenden Sie die Anweisung `COMMIT`, um die Änderungen festzuschreiben.
5. Optional: Aktivieren Sie die Erfassung zusätzlicher Statistikdaten. Standardmäßig wird nur eine minimale Menge von Statistikdaten für jedes Workload-Management-Objekt erfasst. Eine Beschreibung, welche Statistikdaten standardmäßig für die einzelnen Objekte erfasst werden, finden Sie in „Statistikdaten für Workload-Management-Objekte“ auf Seite 128. Geben Sie die Erfassung von zusammengefassten Aktivitätsdaten für Serviceunterklassen und Arbeitsklassen mit dem Schlüsselwort `COLLECT AGGREGATE ACTIVITY DATA` in den Anweisungen `ALTER SERVICE CLASS` und `ALTER WORK ACTION SET` an. Geben Sie die Erfassung von zusammengefassten Anforderungsdaten für Serviceunterklassen mit dem Schlüsselwort `COLLECT AGGREGATE REQUEST DATA` in der Anweisung `ALTER SERVICE CLASS` an. Führen Sie einen `COMMIT` für die Änderungen durch.

6. Geben Sie ein Erfassungsintervall an, indem Sie den Datenbankkonfigurationsparameter `wlm_collect_int` aktualisieren. Der Parameter `wlm_collect_int` gibt ein Zeitintervall in Minuten an. In jedem Intervall wird die speicherinterne Kopie der Workload-Management-Statistikdaten in den aktiven Statistikereignismonitor geschrieben, und die speicherinternen Statistikdaten werden zurückgesetzt. In einer Umgebung mit partitionierten Datenbanken muss der Parameter `wlm_collect_int` auf der Katalogpartition aktualisiert werden. Dieser Parameter kann dynamisch aktualisiert werden. Zum Beispiel:

```
CONNECT TO aliasname_der_datenbank  
UPDATE DATABASE CONFIGURATION USING WLM_COLLECT_INT 5 IMMEDIATE
```

Nachdem Sie die oben aufgelisteten Schritte ausgeführt haben, werden Workload-Management-Statistikdaten alle `wlm_collect_int` Minuten in den Statistikereignismonitor geschrieben. Jeder in den Statistikereignismonitor geschriebene Datensatz hat einen Wert `STATISTICS_TIMESTAMP` und einen Wert `LAST_WLM_RESET`. Das Zeitintervall von `LAST_WLM_RESET` bis `STATISTICS_TIMESTAMP` definiert das Erfassungsintervall (d. h. das Zeitintervall, in dem die Statistikdaten im Datensatz erfasst wurden).

Wenn der Parameter `wlm_collect_int` auf einen Wert ungleich Null gesetzt wird und es keinen aktiven Statistikereignismonitor gibt, werden die speicherinternen Workload-Management-Statistikdaten dennoch alle `wlm_collect_int` Minuten zurückgesetzt, die Statistikdaten werden jedoch nicht erfasst. Die Daten gehen ver-

loren. Deshalb empfiehlt es sich nicht, einen Wert für `wlm_collect_int` ungleich Null anzugeben, ohne einen Statistikereignismonitor zu aktivieren.

Wenn der Parameter `wlm_collect_int` auf 0 (Standardeinstellung) gesetzt ist, werden keine Statistikdaten automatisch an den Statistikereignismonitor gesendet. Sie können mit der gespeicherten Prozedur `WLM_COLLECT_STATS` manuell Statistikdaten zur späteren Langzeitanalyse an den Statistikereignismonitor senden. Wenn diese Prozedur aufgerufen wird, führt sie dieselben Aktionen aus wie eine automatische intervallgesteuerte Statistikdatensammlung, d. h. die speicherinternen Statistikdaten werden an den Statistikereignismonitor gesendet und dann im Hauptspeicher zurückgesetzt. Wenn kein Statistikereignismonitor aktiv ist, werden die speicherinternen Werte zurückgesetzt, aber keine Daten erfasst. Wenn Sie nur die Statistikdaten zurücksetzen wollen, können Sie die Prozedur `WLM_COLLECT_STATS` aufrufen, während kein Statistikereignismonitor aktiv ist.

Die manuelle Erfassung von Statistikdaten greift nicht in die automatische Erfassung von Statistikdaten ein. Nehmen wir z. B. an, dass Sie `wlm_collect_int` auf 60 eingestellt haben. Jede Stunde werden nun die Statistikdaten an den Statistikereignismonitor gesendet. Nehmen wir weiter an, dass die Statistikdaten zum letzten Mal um 5:30 Uhr erfasst wurden. Sie können die Prozedur `WLM_COLLECT_STATS` um 5:55 Uhr aufrufen, wodurch die speicherinternen Werte der Statistikdaten an den Ereignismonitor gesendet und die Statistikdaten zurückgesetzt werden. Die nächste automatische Statistikerfassung erfolgt dennoch um 6:30 Uhr, eine Stunde nach der letzten automatisierten Erfassung. Das Erfassungsintervall wird nicht von einer manuellen Erfassung und Zurücksetzung von Statistikdaten beeinflusst, die innerhalb des Intervalls durchgeführt wurde.

Anmerkung: Die Tabellenfunktionen für Workload-Management-Statistikdaten melden die aktuellen Werte der speicherinternen Statistikdaten. Wenn Sie die automatische Workload-Management-Statistikerfassung aktiviert haben, werden diese Werte regelmäßig in dem mit dem Datenbankkonfigurationsparameter `wlm_collect_int` definierten Intervall zurückgesetzt. Wenn Sie sich die von den Tabellenfunktionen gemeldeten Statistikdaten ansehen, sollten Sie immer auch die Spalte `LAST_RESET` berücksichtigen. Diese Spalte gibt an, wann die speicherinternen Statistikdaten zuletzt zurückgesetzt wurden. Wenn das Zeitintervall zwischen der letzten Zurücksetzung und der momentanen Zeit nicht groß genug ist, sind eventuell nicht ausreichend Daten vorhanden, um aussagekräftige Schlüsse ziehen zu können.

Anmerkung: Wenn Sie die automatische Erfassung von Workload-Management-Statistikdaten verwenden, müssen Sie Ihre Ereignismonitordateien oder -tabellen regelmäßig bereinigen. Der Ereignismonitor bereinigt die erfassten Daten nicht automatisch, so dass Ihre Dateien oder Tabellen im Lauf der Zeit durch die automatische Erfassung gefüllt werden.

Anmerkung: Wenn eine Datenbank inaktiviert wird, werden die speicherinternen Statistikdaten zurückgesetzt. Beim Inaktivieren der Datenbank werden die Statistikdaten jedoch nicht an den Statistikereignismonitor gesendet. Wenn Sie die seit der letzten Erfassung aufgelaufenen Statistikdaten bei der Inaktivierung nicht verlieren wollen, sollten Sie manuell die Prozedur `WLM_COLLECT_STATS` vor dem Inaktivieren der Datenbank aufrufen.

Anmerkung: Die Prozedur `WLM_COLLECT_STATS` setzt die Statistikdaten anders zurück als der Befehl `RESET MONITOR`. Der Befehl `RESET MONITOR` setzt die Werte der Elemente des Überwachungsprogramms für Momentaufnahme zurück, indem er ihre aktuellen Werte speichert. Nachdem der Befehl `RESET MONITOR`

abgesetzt wurde, meldet die Momentaufnahmenverarbeitung die Differenz zwischen diesen Werten und den aktuellen Werten. Dagegen werden bei der durch die Prozedur WLM_COLLECT_STATS veranlassten Zurücksetzung keine Werte gespeichert, sondern alle Statistikzähler selbst für jedes zutreffende Workload-Management-Objekt zurückgesetzt.

Außerdem hat beim Befehl RESET MONITOR jeder Prozess (Anhang) seine eigene Sicht der Überwachungsdaten. Wenn der Benutzer eine Zurücksetzung ausführt, sind andere Benutzer davon nicht betroffen. Eine Zurücksetzung der Workload-Manager-Statistikdaten wirkt sich dagegen auf alle Benutzer aus.

Workload-Management-Tabellenfunktionen für das Abrufen von Statistikdaten

Mit den in diesem Thema beschriebenen Tabellenfunktionen können Sie Statistikdaten zu Workload-Management-Objekten erhalten.

Die Workload-Management-Tabellenfunktionen sind im Schema SYSPROC verfügbar. Diese Hochleistungsfunktionen können Informationen zur im System anfallenden Arbeit zurückgeben, ohne dadurch die zurzeit ausgeführten Workloads zu beeinträchtigen.

Sie können Statistikdaten für eine Reihe unterschiedlicher Zwecke nutzen, z. B. um zu prüfen, ob Änderungen an der Workload-Management-Konfiguration die gewünschten Auswirkungen haben. Wenn Sie z. B. eine neue Arbeitsklasse zur Klassifizierung von READ-Aktivitäten erstellen, wollen Sie eventuell überprüfen, ob die READ-Aktivitäten tatsächlich unter der neuen Arbeitsklasse klassifiziert werden. Mit Hilfe von Tabellenfunktionen können Sie auch bestimmte Systemprobleme schnell erkennen. Sie können z. B. mit Hilfe von Tabellenfunktionen einen akzeptablen Wert für die durchschnittliche Aktivitätslaufzeit ermitteln und erkennen, wenn dieser Wert über dem üblichen Bereich liegt, was auf ein Problem hindeuten kann, das weiter untersucht werden muss.

Alle Tabellenfunktionen für Statistikdaten geben die Statistikdaten zurück, die seit dem letzten Zurücksetzen der Statistikdaten aufgelaufen sind.

Für die Variable *dbpartitionsnummer* können Sie -2 angeben, dass Daten von allen Datenbankpartitionen erfasst werden sollen, oder -1, dass Daten nur von der Datenbankpartition erfasst werden sollen, mit der die Anwendung verbunden ist, die den Tabellenfunktionsaufruf abgesetzt hat (d. h. die Koordinatorpartition). Wenn Sie diese Tabellenfunktionen von Anwendungsprogrammen aufrufen, sollten Sie die Konstanten SQLM_CURRENT_NODE und SQLM_ALL_NODE in der Headerdatei sqlmon.h verwenden, um die Verwendung von -1 und -2 als Literale zu vermeiden.

- `WLM_GET_SERVICE_SUPERCLASS_STATS(servicesuperklassenname, dbpartitionsnummer)`. Mit dieser Tabellenfunktion können Sie Informationen zur oberen Grenze für gleichzeitig bestehende Verbindungen erhalten, die seit der letzten Zurücksetzung der Statistikdaten berechnet wurde. Sie können Platzhalterzeichen verwenden, um Servicesuperklassen und Datenbankpartitionen einzuschließen.
- `WLM_GET_SERVICE_SUBCLASS_STATS(servicesuperklassenname, serviceunterklassenname, dbpartitionsnummer)`. Mit dieser Tabellenfunktion können Sie zusammengefasste Statistikdaten wie die Anzahl von Aktivitäten und die seit der letzten Zurücksetzung der Statistikdaten berechnete durchschnittliche Ausführungszeit erhalten. Sie können diese Informationen für mehrere Serviceunterklassen über eine oder mehrere Datenbankpartitionen erhalten. Sie können Platzhalterzeichen

verwenden, um Servicesuperklassen, Serviceunterklassen und Datenbankpartitionen einzuschließen. Weitere Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Abrufen von Zeitpunktstatistikdaten aus Serviceklassen“ auf Seite 180, „Beispiel: Zusammenfassen von Daten mit Hilfe von Workload-Management-Tabellenfunktionen“ auf Seite 181, „Beispiel: Analysieren einer serviceklassenbezogenen Leistungsminderung des Systems“ auf Seite 184 und „Beispiel: Untersuchen einer workloadbezogenen Leistungsminderung des Systems“ auf Seite 186.

- `WLM_GET_WORKLOAD_STATS(workloadname, dbpartitionsnummer)`. Mit dieser Tabellenfunktion können Sie zusammengefasste Statistikdaten für eine oder alle Workloads und Datenbankpartitionen erhalten. Sie können Platzhalterzeichen verwenden, um Workloadnamen und Datenbankpartitionen einzuschließen. Die zurückgegebenen Statistikdaten umfassen Informationen zu abgeschlossenen Aktivitäten und zur oberen Grenze, die seit der letzten Zurücksetzung der Statistikdaten berechnet wurden. Weitere Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Untersuchen einer workloadbezogenen Leistungsminderung des Systems“ auf Seite 186.
- `WLM_GET_WORK_ACTION_SET_STATS(arbeitsaktionsset-name, dbpartitionsnummer)`. Mit dieser Tabellenfunktion können Sie zusammengefasste Statistikdaten für mindestens ein Arbeitsaktionsset über mindestens eine Datenbankpartition erhalten. Die zurückgegebenen Statistikdaten umfassen Informationen zur Anzahl der Aktivitäten, die jeder Arbeitsklasse seit der letzten Zurücksetzung der Statistikdaten zugeordnet wurden. Informationen zur Verwendung dieser Tabellenfunktion finden Sie in „Beispiel: Analysieren von Workloads nach Aktivitätstyp“ auf Seite 187.
- `WLM_GET_QUEUE_STATS(schwellenwertvergleichselement, schwellenwertdomäne, schwellenwertname, schwellenwert-id)`. Mit dieser Tabellenfunktion erhalten Sie Informationen zu Schwellenwertwarteschlangen. Diese Informationen sagen Ihnen, wieviele Aktivitäten derzeit in der Warteschlange stehen und wann eine Aktivität zuletzt eine Warteschlange verließ. Statistische Information wie die durchschnittliche Wartezeit einer Aktivität in der Warteschlange und die obere Grenze für die Warteschlangengröße sind ebenfalls enthalten. Anhand dieser Informationen können Sie ermitteln, ob eine bestimmte Warteschlange zu einem Engpass im System wird (d. h. Aktivitäten zu lange in der Warteschlange warten).

Zurücksetzen von Statistikdaten für Workload-Management-Objekte

In diesem Thema wird beschrieben, wie Statistikdaten für Workload-Management-Objekte zurückgesetzt werden können.

Vier Ereignisse setzen die speicherinternen Statistikdaten für jedes Workload-Management-Objekt zurück. (Eine Beschreibung der für jedes Objekt gesammelten Statistikdaten finden Sie in „Statistikdaten für Workload-Management-Objekte“ auf Seite 128.)

- Die gespeicherte Prozedur `WLM_COLLECT_STATS` wird aufgerufen. Details finden Sie in „Erfassen von Workload-Management-Statistikdaten mit Hilfe eines Ereignismonitors für Statistikdaten“ auf Seite 137.
- Der Prozess zum automatischen Erfassen und Zurücksetzen von Workload-Management-Statistikdaten, der vom Datenbankkonfigurationsparameter `wlm_collect_int` gesteuert wird, veranlasst eine Erfassung und Zurücksetzung. Details finden Sie in „Erfassen von Workload-Management-Statistikdaten mit Hilfe eines Ereignismonitors für Statistikdaten“ auf Seite 137.

- Die Datenbank wird reaktiviert. Bei jeder Aktivierung einer Datenbank auf einer Datenbankpartition werden die Statistikdaten für alle Workload-Management-Objekte auf dieser Datenbankpartition zurückgesetzt.
- Das Objekt, für das die Statistikdaten gesammelt werden, wird geändert und für die Änderung wird ein COMMIT durchgeführt. Wenn z. B. eine Serviceunterklasse geändert wird, werden die speicherinternen Statistikdaten für diese Serviceunterklasse beim Durchführen eines Commits für die Änderung zurückgesetzt.

Sie können den Zeitpunkt der letzten Zurücksetzung für ein bestimmtes Workload-Management-Objekt mit den Statistiktabellenfunktionen und über die Zeitmarke in der Spalte LAST_RESET ermitteln. Wenn Sie z. B. den Zeitpunkt der letzten Zurücksetzung für die Serviceunterklasse SYSDEFAULTSUBCLASS unter der Servicesuperklasse SYSDEFAULTUSERCLASS sehen wollen, könnten Sie eine Abfrage wie die folgende absetzen:

```
SELECT LAST_RESET FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS( 'SYSDEFAULTUSERCLASS',
'SYSDEFAULTSUBCLASS', -2)) AS T
```

Alle Tabellenfunktionen für Statistikdaten geben die Statistikdaten zurück, die seit dem letzten Zurücksetzen der Statistikdaten aufgelaufen sind. Eine Zurücksetzung der Statistikdaten erfolgt, wenn eine Datenbank aktiviert oder reaktiviert wird, wenn Sie ein Workload-Management-Objekt ändern (nur die Statistikdaten für dieses Objekt werden zurückgesetzt) und wenn Sie die gespeicherte Prozedur WLM_COLLECT_STATS aufrufen. Statistikdaten werden auch automatisch entsprechend der vom Datenbankkonfigurationsparameter **wlm_collect_int** definierten Zeitspanne zurückgesetzt, wenn Sie diesen Parameter auf einen Wert ungleich Null einstellen.

Der von **wlm_collect_int** definierte Zeitraum wird nicht beeinflusst von einer Statistikrücksetzung, die in dem vom Konfigurationsparameter festgelegten Intervall erfolgt. Wenn Sie z. B. die Tabellenfunktion WLM_COLLECT_STATS 5 Minuten nach dem Start eines von **wlm_collect_int** definierten 20-minütigen Intervalls ausführen, läuft das Intervall dennoch 15 Minuten später ab. Die erfolgte Statistikerfassung und -zurücksetzung verzögert nicht die nächste Statistikerfassung und -zurücksetzung um 5 Minuten.

Eine Änderung der Angabe der Histogrammschablone für eine Serviceklasse oder die Änderung einer Histogrammschablone wird erst nach der Zurücksetzung der Statistikdaten wirksam.

Wenn Sie die Tabellenfunktion WLM_COLLECT_STATS aufrufen, um Statistikdaten zu erfassen und zurückzusetzen, während eine andere Erfassung und Zurücksetzung abläuft (z. B. wenn der Aufruf der Tabellenfunktion das periodische Erfassungs- und Zurücksetzungsintervall von **wlm_collect_int** überlappt oder wenn ein anderer Benutzer gleichzeitig WLM_COLLECT_STATS aufruft), wird die Erfassungs- und Zurücksetzungsanforderung von WLM_COLLECT_STATS ignoriert, und die Warnung SQL1632W wird zurückgegeben.

Überwachen von Schwellenwertverstößen

Bei einem Verstoß gegen einen Workload-Management-Schwellenwert wird ein Schwellenwertverstoßeintrag in den aktiven THRESHOLD VIOLATIONS-Ereignismonitor (sofern vorhanden) geschrieben.

Der Schwellenwertverstoßeintrag umfasst die folgenden Informationen:

- Eine Beschreibung des Schwellenwerts, gegen den verstoßen wurde (ID, Maximalwert usw.).
- Eine ID der Aktivität, die den Schwellenwertverstoß verursachte, einschließlich der ID der Anwendung, die die Aktivität übergab, der eindeutigen Aktivitäts-ID und der UOW-ID.
- Den Zeitpunkt des Schwellenwertverstoßes.
- Die durchgeführte Aktion. Die Aktion gibt an, ob die Aktivität, die den Schwellenwertverstoß verursachte, weiter ausgeführt werden durfte oder gestoppt wurde. Wenn die Aktivität gestoppt wurde, hat die Anwendung, die dies Aktivität übergeben hat, einen Fehler SQL4712N empfangen.

Sie können optional detaillierte Aktivitätsinformationen (einschließlich Anweisungstext) in einen aktiven Aktivitätsereignismonitor schreiben lassen, wenn der Schwellenwertverstoß von einer Aktivität verursacht wurde. Die Aktivitätsinformationen werden nach Abschluss der Aktivität und nicht zum Zeitpunkt des Schwellenwertverstoßes geschrieben. Geben Sie an, dass bei einem Schwellenwertverstoß Aktivitätsinformationen erfasst werden sollen, indem Sie das Schlüsselwort `COLLECT ACTIVITY DATA` in der Anweisung zum Erstellen oder Ändern eines Schwellenwerts oder eines Arbeitsaktionsset verwenden.

Gehen Sie wie folgt vor, um Schwellenwertverstöße zu überwachen:

1. Erstellen Sie mit der Anweisung `CREATE EVENT MONITOR` einen Ereignismonitor vom Typ `THRESHOLD VIOLATIONS`. Zum Beispiel:

```
CREATE EVENT MONITOR VIOLATIONS FOR THRESHOLD VIOLATIONS WRITE TO TABLE
```
2. Verwenden Sie die Anweisung `COMMIT`, um die Änderungen festzuschreiben.
3. Aktivieren Sie den Ereignismonitor mit Hilfe der Anweisung `SET EVENT MONITOR STATE`. Statt der Anweisung `SET EVENT MONITOR STATE` können Sie auch die Standardeinstellung `AUTOSTART` für den `THRESHOLD VIOLATIONS`-Ereignismonitor verwenden, um ihn bei der nächsten Aktivierung der Datenbank aktivieren zu lassen. Es kann jedoch jeweils nur ein Ereignismonitor vom Typ `THRESHOLD VIOLATIONS` auf einer Datenbankpartition aktiv sein. Wenn Sie mehrere `THRESHOLD VIOLATIONS`-Ereignismonitore definieren wollen, sollten Sie nicht die Option `AUTOSTART` verwenden.
4. Verwenden Sie die Anweisung `COMMIT`, um die Änderungen festzuschreiben.

Anmerkung: Wenn Sie Schwellenwerte erstellen, sollten Sie einen Schwellenwertverstoß-Ereignismonitor erstellen und aktivieren, so dass Sie alle Schwellenwertverstöße überwachen können. Ein Schwellenwertverstoß-Ereignismonitor hat keine Auswirkung, sofern nicht gegen Schwellenwerte verstoßen wird.

Erfassen von Daten für einzelne Aktivitäten

Sie können mit einem `ACTIVITIES`-Ereignismonitor Daten für einzelne Aktivitäten erfassen, die in Ihrem System ausgeführt werden. Die erfassten Daten umfassen Elemente wie Anweisungstext und Kompilierungsumgebung und können zur Untersuchung und Diagnose von Problemen sowie zur Eingabe in andere Tools (z. B. Designadvisor) verwendet werden.

Sie können Informationen zu einzelnen Aktivitäten für Serviceunterklassen, Workloads, Arbeitsklassen (über Arbeitsaktionen) und Schwellenwertverstöße erfassen. Sie aktivieren die Aktivitätserfassung mit dem Schlüsselwort `COLLECT ACTIVITY DATA` der Anweisungen `CREATE` und `ALTER` für diese Workload-Management-

Objekte. Beim Abschluss einer Aktivität werden Informationen zur Aktivität an den aktiven ACTIVITIES-Ereignismonitor gesendet, wenn folgende Bedingungen erfüllt sind:

- Die Aktivität wurde von einer Anwendung übergeben, die einer Workload zugeordnet ist, für die COLLECT ACTIVITY DATA angegeben wurde. Oder:
 - Die Aktivität wird in einer Serviceunterklasse ausgeführt, für die COLLECT ACTIVITY DATA angegeben ist. Oder:
 - Auf die Aktivität wurde die Arbeitsaktion COLLECT ACTIVITY DATA angewendet. Oder:
 - Die Aktivität verstößt gegen einen Schwellenwert, der mit der Aktion COLLECT ACTIVITY DATA definiert wurde.

Das Schlüsselwort COLLECT ACTIVITY DATA steuert auch den Umfang der Informationen, die an den ACTIVITIES-Ereignismonitor gesendet werden. Wenn das Schlüsselwort die Option WITH DETAILS angibt, werden Anweisungsinformationen (wie der Anweisungstext) erfasst. Wenn das Schlüsselwort die Option WITH DETAILS AND VALUES angibt, werden auch Datenwerte erfasst.

Auf eine Aktivität werden eventuell mehrere Schlüsselwörter COLLECT ACTIVITY DATA angewendet. Beispielsweise wird die Aktivität in einer Serviceunterklasse ausgeführt, für die COLLECT ACTIVITY DATA angegeben ist, und verstößt während der Ausführung möglicherweise gegen einen Schwellenwert mit der Aktion COLLECT ACTIVITY DATA. In dieser Situation wird die Aktivität nur einmal erfasst. Das Schlüsselwort COLLECT, das die größte zu erfassende Datenmenge angibt, wird auf die Aktivität angewendet. Wenn z. B. sowohl COLLECT ACTIVITY DATA WITHOUT DETAILS und COLLECT ACTIVITY DATA WITH DETAILS auf eine Aktivität angewendet werden, wird die Aktivität mit detaillierten Informationen erfasst.

Gehen Sie wie folgt vor, um die Erfassung von Aktivitäten für ein bestimmtes Workload-Management-Objekt zu aktivieren:

1. Erstellen Sie mit der Anweisung CREATE EVENT MONITOR einen ACTIVITIES-Ereignismonitor.
2. Verwenden Sie die Anweisung COMMIT, um die Änderungen festzuschreiben.
3. Aktivieren Sie den Ereignismonitor mit Hilfe der Anweisung SET EVENT MONITOR STATE. Statt der Anweisung SET EVENT MONITOR STATE können Sie auch die Standardeinstellung AUTOSTART für den ACTIVITIES-Ereignismonitor verwenden, um ihn bei der nächsten Aktivierung der Datenbank aktivieren zu lassen. Es kann jedoch jeweils nur ein Ereignismonitor vom Typ ACTIVITIES auf einer Datenbankpartition aktiv sein. Wenn Sie mehrere ACTIVITIES-Ereignismonitore definieren wollen, sollten Sie nicht die Option AUTOSTART verwenden.
4. Verwenden Sie die Anweisung COMMIT, um die Änderungen festzuschreiben.
5. Definieren Sie die Objekte, für die Sie Aktivitäten erfassen wollen, mit der Anweisung ALTER SERVICE CLASS, ALTER WORK ACTION SET, ALTER THRESHOLD oder ALTER WORKLOAD, und geben Sie das Schlüsselwort COLLECT ACTIVITY DATA an.
6. Verwenden Sie die Anweisung COMMIT, um die Änderungen festzuschreiben.

Anmerkung: Die Erfassung einzelner Aktivitäten ist aufwendiger als die Erfassung von Workload-Management-Statistikdaten. Sie sollten versuchen, eine Aktivitätssammlung einzurichten, die so wenig Aktivitäten wie möglich erfasst. Wenn Sie z. B. Aktivitäten untersuchen müssen, die von einer bestimmten Anwendung über-

geben wurden, sollten Sie diese Anwendung eingrenzen, indem Sie eine Workload oder eine Serviceklasse speziell für diese Anwendung erstellen und die Aktivitätserfassung nur für diese Workload oder Serviceklasse aktivieren.

Sie wissen vielleicht nicht immer im voraus, dass Sie eine Aktivität erfassen wollen. Beispielsweise haben Sie eine Abfrage, deren Ausführung sehr lange dauert, und Sie wollen Informationen zu ihr für die spätere Analyse erfassen. In dieser Situation ist es zu spät, das Schlüsselwort COLLECT ACTIVITY DATA für die Workload-Management-Objekte anzugeben, da die Aktivität bereits in das System eingegangen ist. In diesem Fall können Sie die gespeicherte Prozedur WLM_CAPTURE_ACTIVITY_IN_PROGRESS verwenden. Die gespeicherte Prozedur WLM_CAPTURE_ACTIVITY_IN_PROGRESS sendet Informationen zu einer momentan ausgeführten Aktivität an den aktiven ACTIVITIES-Ereignismonitor. Sie geben die zu erfassende Aktivität durch die Anwendungskennung, die UOW-ID und die Aktivitäts-ID an. Informationen zur Aktivität werden sofort beim Aufruf der Prozedur an den ACTIVITIES-Ereignismonitor gesendet; Sie müssen nicht auf den Abschluss der Aktivität warten.

Wenn Sie die gespeicherte Prozedur WLM_CAPTURE_ACTIVITY_IN_PROGRESS verwenden, um Aktivitätsinformationen für eine Prozedur mit INOUT-Parametern zu erfassen, könnten die INOUT-Werte bis zum Zeitpunkt der Erfassung überschrieben worden sein. Diese Situation tritt nicht auf, wenn die Aktivität als Ergebnis des Schlüsselworts COLLECT ACTIVITY DATA WITH DETAILS AND VALUES für eine Serviceunterklasse, eine Workload, eine Arbeitsaktion oder einen prädiktiven Schwellenwert erfasst wird.

Importieren von Aktivitäten in den Designadvisor

Sie können von einem Aktivitätsereignismonitor erfasste Aktivitäten in den Designadvisor importieren, um leichter Entscheidungen über die Datenbankobjekte treffen zu können, auf die diese Aktivitäten zugreifen.

In den Designadvisor importierte Aktivitäten müssen mit den Optionen COLLECT ACTIVITY DATA WITH DETAILS oder COLLECT ACTIVITY DATA WITH DETAILS AND VALUES erfasst worden sein. Die Option COLLECT ACTIVITY DATA WITHOUT DETAILS ist nicht ausreichend. Sie erfasst nicht den Anweisungstext, der für den Designadvisor erforderlich ist.

Führen Sie für den Import von Aktivitätsinformationen aus den Aktivitätsereignismonitortabellen in den Designadvisor den Befehl db2advis mit dem Parameter **-wlm** gefolgt von weiteren Parametern aus:

1. Name des Aktivitätsereignismonitors
2. Optional: Name der Workload oder der Serviceklasse
3. Optional: Startzeit und Endzeit

Wenn Sie z. B. Informationen zu allen Aktivitäten importieren wollen, die vom Ereignismonitor DB2ACTIVITIES in der Datenbank SAMPLE erfasst wurden, verwenden Sie den folgenden Befehl:

```
db2advis -d SAMPLE -wlm DB2ACTIVITIES
```

Anmerkung: Sie können Informationen von Aktivitätsereignismonitortabellen nur über die Befehlszeilenschnittstelle des Designadvisors importieren.

Abbrechen von Aktivitäten

Wenn eine Aktivität zu viele Ressourcen verbraucht oder sich in einer Endlosschleife befindet, wollen Sie sie möglicherweise abbrechen. Das Abbrechen einer Aktivität ist eine schonendere Methode als das erzwungene Beenden der Anwendung, die die Aktivität übergeben hat. Eine abgebrochene Aktivität gibt SQL4725N an den Benutzer zurück, beendet jedoch nicht die Verbindung und wirkt sich nicht auf andere Aktivitäten dieses Benutzers aus. Durch das erzwungene Beenden der Anwendung werden sowohl die Verbindung als auch die Aktivitäten dieses Benutzers beendet.

Sie können eine Aktivität nur dann explizit abbrechen, wenn eine Koordinatoraktivität momentan an einer Anforderung für die Aktivität arbeitet. Wenn Sie eine Aktivität abbrechen, die sich im Leerlauf befindet (d. h. für die keine Anforderungen verarbeitet werden), erhält die Aktivität den Status CANCEL_PENDING und wird beim Empfang der nächsten Anforderung abgebrochen. Wenn Sie z. B. versuchen, eine Cursoraktivität zwischen Abrufen abzuberechnen, wird der Fehler SQL4725N erst beim nächsten Abruf nach der Abbruchanforderung an den Benutzer zurückgegeben.

Alle Benutzeraktivitäten können abgebrochen werden, auch das Dienstprogramm LOAD und gespeicherte Prozeduren.

Gehen Sie wie folgt vor, um eine Aktivität abzuberechnen:

1. Geben Sie die Aktivität an, die Sie abbrechen wollen. Mit der Tabellenfunktion WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES können Sie die Aktivitäten ermitteln, die in einer Anwendung ausgeführt werden. Sie können auch die Tabellenfunktion WLM_GET_ACTIVITY_DETAILS verwenden, um zusätzliche Details zu einer bestimmten Aktivität zu erhalten, wenn die Informationen in WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES nicht ausreichen, um die von den Aktivitäten ausgeführte Arbeit zu ermitteln.
2. Brechen Sie die Aktivität mit Hilfe der gespeicherten Prozedur WLM_CANCEL_ACTIVITY ab. Die gespeicherte Prozedur verwendet die folgenden Argumente: *anwendungskennung*, *uow-id* und *aktivitäts-id*. Ein Beispiel für die Verwendung dieser gespeicherten Prozedur finden Sie in „Beispiel: Erkennen von blockierten Aktivitäten“ auf Seite 188.

Richtlinien für das Erfassen von Informationen zu einer Rogue-Aktivität und Untersuchen dieser Aktivität

Dieses Thema bietet Richtlinien zur Erfassung von Informationen und zur Untersuchung einer Rogue-Aktivität.

Erstellen Sie zuerst eine Gruppe von Kriterien dafür, was Sie als Rogue-Aktivität betrachten würden. Zum Beispiel:

- Eine Aktivität, die in einer Serviceklasse für Aktivitäten mit einem geringen geschätzten Aufwand ausgeführt wird und deren Ausführung länger als 1 Stunde dauert
- Eine Aktivität, die ungewöhnlich viele Zeilen zurückgibt
- Eine Aktivität, die einen sehr großen Tabellenbereich für temporäre Tabellen verbraucht

Erstellen Sie dann Schwellenwerte, die diese Kriterien beschreiben und eine Aktion COLLECT ACTIVITY DATA WITH DETAILS enthalten. Bei einem Schwellenwert-

verstoß werden Informationen zu der Aktivität, die den Schwellenwertverstoß verursachte, bei Abschluss der Aktivität an den aktiven ACTIVITIES-Ereignismonitor gesendet.

Wenn Sie z. B. Informationen zu Datenbankaktivitäten erfassen wollen, deren Ausführung länger als 3 Stunden dauert, erstellen Sie einen Schwellenwert wie den folgenden:

```
CREATE THRESHOLD LONGRUNNINGACTIVITIES FOR DATABASE ACTIVITIES ENFORCEMENT DATABASE  
WHEN ACTIVITYTOTALTIME > 3 HOURS COLLECT ACTIVITY DATA WITH DETAILS CONTINUE
```

Sie können dann die Informationen analysieren, die in den Ereignismonitor geschrieben werden. Für DML-Aktivitäten können auch der Anweisungstext und die Kompilierungsumgebung in den Ereignismonitor geschrieben werden, so dass Sie darauf DB2 explain ausführen können, um die Leistung der Aktivität weiter zu untersuchen.

Modellierung der Workload-Management-Leistung

Die Auslastung auf Ihrem System kann als Menge von Aktivitäten modelliert werden, die im System mit einer Geschwindigkeit eingehen, die durch eine Eingangsratenverteilung für Aktivitäten (oft gemessen als Kehrwert, die Verteilung der *Zwischenankunftszeit* (Inter-Arrival Time)) und der Zeitdauer gesteuert wird, während der die Aktivitäten zur Ausführung im System verbleiben, die einer Servicezeitverteilung folgt.

Die Zwischenankunftszeit ist die Zeit zwischen dem Eingang einer Aktivität und dem Eingang der nächsten Aktivität. Die Servicezeit ist die Zeit, während der eine Aktivität zur Ausführung im System verbleibt. Wenn Sie z. B. eine Abfrage zum Zeitpunkt 0 Sekunden übergeben, diese Abfrage 2 Sekunden in einer Warteschlange verbleibt und zum Zeitpunkt 5 Sekunden abgeschlossen wird, beträgt die Servicezeit $5 - 2 = 3$ Sekunden. Bei der Servicezeit wird davon ausgegangen, dass keine andere Arbeit auf dem System ausgeführt wird (d. h. sie ist nicht die beobachtete Ausführungszeit, sondern vielmehr die Zeit, die benötigt würde, um die Aktivität isoliert auszuführen). Die Servicezeitverteilung kann für DML-Aktivitäten näherungsweise berechnet werden anhand des geschätzten Aufwands in Timerons, bei dem sowohl die CPU- als auch die E/A-Zeit für eine Aktivität berücksichtigt wird.

Sie können ein Workloadmodell für Ihr System durch Messung der Zwischenankunftszeitverteilung und der Servicezeitverteilung der Aktivitäten im System erstellen. Sie erhalten die Zwischenankunftszeitverteilungen und die (anhand des geschätzten Aufwands) näherungsweise berechneten Servicezeitverteilungen durch erweiterte zusammengefasste Aktivitätsstatistiken für Serviceunterklassen oder Arbeitsklassen (über Arbeitsaktionen) und einen Statistikereignismonitor. Diese Statistikdaten werden standardmäßig nicht erfasst. Weitere Informationen finden Sie in „Statistikdaten für Workload-Management-Objekte“ auf Seite 128.

Arbeiten mit Histogrammen

Erstellen einer Histogrammschablone

Mit der Anweisung CREATE HISTOGRAM TEMPLATE können Sie eine Histogrammschablone erstellen. Histogrammschablonen werden von Serviceunterklassen und Arbeitsaktionen verwendet, um die Binwerte für die Statistikdaten festzulegen, die mit Hilfe von Histogrammen gepflegt werden.

Zum Erstellen einer Histogrammschablone benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Informationen zu Voraussetzungen finden Sie in den folgenden Abschnitten:

- Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303
- Namensregeln

Einige Statistikdaten zu DB2-Serviceunterklassen, Arbeitsklassenaktivitäten und Anforderungen werden mit Hilfe von Histogrammen erfasst. Alle Histogramme haben eine feste Anzahl Bins, und jeder Bin stellt einen Bereich dar, in dem Aktivitäten oder Anforderungen gezählt werden. Der für die Bins verwendete Einheiten-typ hängt vom Typ des erstellten Histogramms ab. Die Histogrammschablone beschreibt den höchsten Wert des vorletzten Bins im Histogramm, der sich auf die Werte aller Bins im Histogramm auswirkt. Weitere Informationen zu Histogrammen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.

Gehen Sie wie folgt vor, um eine Histogrammschablone zu erstellen:

1. Setzen Sie die Anweisung CREATE HISTOGRAM TEMPLATE ab, und geben Sie den Namen der Histogrammschablone, die Sie erstellen wollen, sowie einen Wert für das Schlüsselwort HIGH BIN VALUE an, um den höchsten Wert für den zweitletzten Bin festzulegen.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird das Histogramm zum Sicht SYSCAT.HISTOGRAMTEMPLATES hinzugefügt, und die Bins werden zur Sicht SYSCAT.HISTOGRAMTEMPLATEBINS hinzugefügt.

Ändern einer Histogrammschablone

Mit der Anweisung ALTER HISTOGRAM TEMPLATE können Sie eine vorhandene Histogrammschablone ändern. Histogrammschablonen werden von Serviceunterklassen und Arbeitsaktionen verwendet, um die Binwerte für die Statistikdaten festzulegen, die mit Hilfe von Histogrammen gepflegt werden.

Zum Ändern einer Histogrammschablone benötigen Sie die Berechtigung SYSADM oder DBADM.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Einige Statistikdaten zu DB2-Serviceunterklassen, Arbeitsklassenaktivitäten und Anforderungen werden mit Hilfe von Histogrammen erfasst. Alle Histogramme haben eine feste Anzahl Bins, und jeder Bin stellt einen Bereich dar, in dem Aktivitäten oder Anforderungen gezählt werden. Der für die Bins verwendete Einheiten-typ hängt vom Typ des Histogramms ab, das Sie erstellen. Die Histogrammschablone beschreibt den höchsten Wert des vorletzten Bins im Histogramm, der sich auf die Werte aller Bins im Histogramm auswirkt. Weitere Informationen zu Histogrammen finden Sie in „Histogramme beim Workload-Management“ auf Seite 133.

Gehen Sie wie folgt vor, um eine Histogrammschablone zu ändern:

1. Setzen Sie die Anweisung `ALTER HISTOGRAM TEMPLATE` ab, und geben Sie den Namen der Histogrammschablone, die Sie ändern wollen, sowie einen Wert für den Parameter `HIGH BIN VALUE` an, um den höchsten Wert für den zweitletzten Bin zu ändern.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird der obere Binwert für das Histogramm in der Sicht `SYSCAT.HISTOGRAMTEMPLATEBINS` aktualisiert. Die Änderung wird erst bei der nächsten Zurücksetzung der Workload-Management-Statistikdaten wirksam. Weitere Informationen finden Sie in „Zurücksetzen von Statistikdaten für Workload-Management-Objekte“ auf Seite 141.
3. Optional: Führen Sie die gespeicherte Prozedur `WLM_COLLECT_STATS` aus, um die Statistikdaten zu erfassen und zurückzusetzen, so dass die neue Histogrammschablone sofort verwendet werden kann.

Löschen einer Histogrammschablone

Wenn Sie eine Histogrammschablone nicht mehr benötigen, können Sie sie löschen.

Zum Löschen einer Histogrammschablone benötigen Sie die Berechtigung `SYSADM` oder `DBADM`.

Weitere Informationen zu Voraussetzungen finden Sie in Anhang A, „Aspekte von DDL-Anweisungen für Workload-Management“, auf Seite 303.

Die Histogrammschablone `SYSDEFAULTHISTOGRAM` kann nicht gelöscht werden.

Sie können eine Histogrammschablone nicht löschen, wenn eine Serviceunterklasse oder eine Arbeitsaktion darauf verweist. Sie können die Serviceunterklassen und Arbeitsaktionen, die auf eine Histogrammschablone verweisen, durch Abfrage der Sicht `SYSCAT.HISTOGRAMTEMPLATESUSE` anzeigen.

Gehen Sie wie folgt vor, um eine Histogrammschablone zu löschen:

1. Verwenden Sie die Anweisung `DROP HISTOGRAM TEMPLATE`.
2. Führen Sie ein Commit für Ihre Änderungen durch. Beim Durchführen eines Commits für Ihre Änderungen wird das Histogramm aus der Sicht `SYSCAT.HISTOGRAMTEMPLATES` entfernt, und die zugehörigen Bins werden aus der Sicht `SYSCAT.HISTOGRAMTEMPLATEBINS` entfernt.

Teil 4. Beispiele

Kapitel 7. Workload-Management-Beispiele

Beispiel: Verwenden von Serviceklassen

Das folgende Beispiel zeigt, wie die Datenbankauslastung mit Hilfe von Serviceklassen gesteuert werden kann.

Dieses Beispiel stammt aus dem fiktiven International Beer Emporium. International Beer Emporium ist ein mittelständisches Unternehmen mit im Wesentlichen fünf Abteilungen: Vertrieb, Buchhaltung, Entwicklung, Test und Produktion. Alle fünf Abteilungen greifen gemeinsam auf dieselbe Produktkatalogdatenbank zu.

Erstimplementierung der Workload-Management-Lösung

Die Produktkatalogdatenbank läuft meistens zufrieden stellend. Manchmal beklagen sich die Benutzer jedoch, dass ihre Anwendungen keine Verbindung zur Datenbank herstellen können, da die maximale Anzahl der Verbindungen überschritten wurde. Nach der Migration auf DB2 Version 9.5 beschließt Bob, der Datenbankadministrator, Serviceklassen auszuprobieren. Bob will die Verwendungsmuster der Produktkatalogdatenbank aller fünf Abteilungen ermitteln und herausfinden, warum seiner Datenbank gelegentlich die Verbindungen ausgehen. Im Folgenden sind die Schritte aufgeführt, die Bob zum Einrichten von Serviceklassen ausführt:

1. Zuerst erstellt er Servicesuperklassen für jede der Abteilungen (für jede Servicesuperklasse wird automatisch auch die Standardserviceunterklasse erstellt):
 - SALES wird für die Vertriebsabteilung erstellt:
CREATE SERVICE CLASS SALES
 - ACCOUNTING wird für die Buchhaltungsabteilung erstellt:
CREATE SERVICE CLASS ACCOUNTING
 - ENGINEERING wird für die Entwicklungsabteilung erstellt:
CREATE SERVICE CLASS ENGINEERING
 - TESTING wird für die Testabteilung erstellt:
CREATE SERVICE CLASS TESTING
 - PRODUCTION wird für die Produktionsabteilung erstellt:
CREATE SERVICE CLASS PRODUCTION
2. Bob erstellt Sitzungsbenutzergruppen mit entsprechenden Berechtigungs-IDs für jede der Abteilungen:
 - Eine Sitzungsbenutzergruppe mit der Berechtigungs-ID SALESGRP wird erstellt. Diese Gruppe umfasst die Berechtigungs-IDs aller Benutzer in der Vertriebsabteilung.
 - Eine Sitzungsbenutzergruppe mit der Berechtigungs-ID ACCTNGRP wird erstellt. Diese Gruppe umfasst die Berechtigungs-IDs aller Benutzer in der Buchhaltungsabteilung.
 - Eine Sitzungsbenutzergruppe mit der Berechtigungs-ID ENGINGRP wird erstellt. Diese Gruppe umfasst die Berechtigungs-IDs aller Benutzer in der Entwicklungsabteilung.
 - Eine Sitzungsbenutzergruppe mit der Berechtigungs-ID TESTGRP wird erstellt. Diese Gruppe umfasst die Berechtigungs-IDs aller Benutzer in der Testabteilung.

- Eine Sitzungsbenutzergruppe mit der Berechtigungs-ID PRODGRP wird erstellt. Diese Gruppe umfasst die Berechtigungs-IDs aller Benutzer in der Produktionsabteilung.
3. Bob erstellt Workloads, um Verbindungen von jeder Gruppe der zugehörigen Serviceklasse zuzuordnen:
- Workload WL_SALES wird erstellt, und ihre Sitzungsbenutzergruppe wird auf SALESGRP eingestellt. WL_SALES ordnet ihre Verbindungen der Service-superklasse SALES zu:

```
CREATE WORKLOAD WL_SALES SESSION_USER GROUP ('SALESGRP') SERVICE CLASS SALES
```
 - Workload WL_ACCOUNTING wird erstellt, und ihre Sitzungsbenutzergruppe wird auf ACCTNGRP eingestellt. WL_ACCOUNTING ordnet ihre Verbindungen der Servicesuperklasse ACCOUNTING zu:

```
CREATE WORKLOAD WL_ACCOUNTING SESSION_USER GROUP ('ACCTNGRP') SERVICE CLASS ACCOUNTING
```
 - Workload WL_ENGINEERING wird erstellt, und ihre Sitzungsbenutzergruppe wird auf ENGINGRP eingestellt. WL_ENGINEERING ordnet ihre Verbindungen der Serviceklasse ENGINEERING zu:

```
CREATE WORKLOAD WL_ENGINEERING SESSION_USER GROUP ('ENGINGRP') SERVICE CLASS ENGINEERING
```
 - Workload WL_TEST wird erstellt, und ihre Sitzungsbenutzergruppe wird auf TESTGRP eingestellt. WL_TEST ordnet ihre Verbindungen der Serviceklasse TESTING zu:

```
CREATE WORKLOAD WL_TEST SESSION_USER GROUP ('TESTGRP') SERVICE CLASS TESTING
```
 - Workload WL_PRODUCTION wird erstellt, und ihre Sitzungsbenutzergruppe wird auf PRODGRP eingestellt. WL_PRODUCTION ordnet ihre Verbindungen der Serviceklasse PRODUCTION zu:

```
CREATE WORKLOAD WL_PRODUCTION SESSION_USER GROUP ('PRODGRP') SERVICE CLASS PRODUCTION
```

Bob nutzt die Standardeinstellungen für Serviceklassen und Workloads. Er will die Datenbanknutzungsmuster beobachten, bevor er Steuerungen auf die Serviceklassen anwendet. Daraus ergeben sich die folgenden Definitionen für die Service-superklassen:

Tabelle 17. Serviceklassendefinitionen

Serviceklasse
SALES
ACCOUNTING
ENGINEERING
TESTING
PRODUCTION
SYSDEFAULTUSERCLASS
SYSDEFAULTMAINTENANCECLASS
SYSDEFAULTSYSTEMCLASS

Wenn die Workload-Management-Lösung wie oben beschrieben implementiert ist, wird Arbeit von jeder Abteilung zur eigenen Servicesuperklasse weitergeleitet. Arbeit von Abteilungen, für die keine speziellen Vorgaben gemacht wurden, wird der Standard-superklasse SYSDEFAULTUSERCLASS zugeordnet. Mit dieser Konfiguration kann Bob die Arbeit in den einzelnen Serviceklassen überwachen, um das Datenbanknutzungsmuster der Abteilungen zu ermitteln.

Erste Verbesserung der Workload-Management-Implementierung

Nach der letzten Verbindungsspitze fragt Bob die Servicesuperklassenstatistik mit der Tabellenfunktion `WLM_GET_SERVICE_SUPERCLASS_STATS` ab und untersucht den oberen Grenzwert für jede Servicesuperklasse. Bob erkennt, dass die obere Grenze für Verbindungen für alle Abteilungen mit Ausnahme der Testabteilung nahe 100 liegt. Die Statistikdaten für die Testabteilung zeigen jedoch, dass das Testteam einmal über 800 Verbindungen herstellte.

Einmal pro Monat führt die Testabteilung ihre allmonatliche intensive Produktprüfung durch. Zu dieser Zeit stellt die Abteilung bis zu 1000 gleichzeitig bestehende Verbindungen her. Da der Konfigurationsparameter des Datenbankmanagers **max_connections** auf 1000 eingestellt ist, nutzt die Testabteilung die meisten der verfügbaren Verbindungen zur Datenbank. Wenn das System 1000 Verbindungen erreicht hat, werden alle weiteren Verbindungsversuche abgewiesen.

Auf Grund von Speichereinschränkungen im System können die Konfigurationswerte **max_connections** und **maxagents** auf dem Datenserver nicht erhöht werden, um weitere Verbindungen zuzulassen.

Um zu verhindern, dass die Testabteilung alle Verbindungen nutzt, beschließt Bob, die Anzahl der Verbindungen von der Testabteilung zu beschränken und sicherzustellen, dass jede der anderen vier Abteilungen ausreichend viele Verbindungen zur Datenbank erhalten kann, um ihre Geschäftsziele erfüllen zu können.

Die anderen vier Abteilungen benötigen in der Regel jeweils nicht mehr als 150 gleichzeitig bestehende Verbindungen. Außerdem bemerkt Bob auch, dass die Standardbenutzer-, die Standardverwaltungs- und die Standardssystemservicesuperklasse nur selten Verbindungen enthalten. Er beschließt deshalb, dass 100 Verbindungen für diese Standardservicesuperklassen ausreichen sollten. Nachdem 700 Verbindungen (600 für die vier Abteilungen und 100 für die Standardklassen) aus dem Pool **max_connections** mit 1 000 verfügbaren Verbindungen zugeordnet wurden, stehen noch 300 Verbindungen für die Testabteilung zur Verfügung. Durch die Beschränkung der Testabteilung auf maximal 300 Verbindungen, sollten Verbindungen von Benutzern anderer Abteilungen nicht abgewiesen werden.

Zur Beschränkung der Testgruppe auf maximal 300 gleichzeitig bestehende Verbindungen erstellt Bob einen Schwellenwert `MAXSERVICECLASSCONNECTIONS` von 300 für die Serviceklasse `TESTING`.

```
CREATE THRESHOLD MAXSERVICECLASSCONNECTIONS FOR SERVICE CLASS TESTING ACTIVITIES
ENFORCEMENT DATABASE PARTITION
WHEN TOTALSCPARTITIONCONNECTIONS > 300 STOP EXECUTION
```

Nach der Implementierung dieser Änderung sieht die Workload-Management-Konfiguration wie folgt aus:

Tabelle 18. Konfiguration nach dem Hinzufügen des Schwellenwerts für die Servicesuperklasse TESTING

Serviceklasse	Schwellenwert MAXSERVICECLASSCONNECTIONS
SALES	Nicht zutreffend
ACCOUNTING	Nicht zutreffend
ENGINEERING	Nicht zutreffend
TESTING	300

Tabelle 18. Konfiguration nach dem Hinzufügen des Schwellenwerts für die Servicesuperklasse TESTING (Forts.)

Serviceklasse	Schwellenwert MAXSERVICECLASSCONNECTIONS
PRODUCTION	Nicht zutreffend
SYSDEFAULTUSERCLASS	Nicht zutreffend
SYSDEFAULTMAINTENANCECLASS	Nicht zutreffend

Da die Serviceklasse TESTING nur maximal 300 gleichzeitig bestehende Verbindungen enthalten kann, werden alle Verbindungsanforderungen über diesen Schwellenwert hinaus abgewiesen. Auf die anderen Serviceklassen wird kein Schwellenwert MAXSERVICECLASSCONNECTIONS angewendet, so dass diese Serviceklassen die verbleibenden 700 verfügbaren Verbindungen zum Datenserver gemeinsam nutzen können. Da es keine Konkurrenzsituation für Verbindungen zwischen diesen Serviceklassen gibt, legt Bob keine Verbindungsschwellenwerte für diese fest.

Zweite Verbesserung der Workload-Management-Implementierung

Obwohl Verbindungen von der Vertriebs-, Buchhaltungs-, Entwicklungs- und Produktionsabteilung nicht mehr abgewiesen werden, beklagen sich Benutzer aus diesen Abteilungen immer noch über die schlechte Leistung, wenn die Testabteilung intensive Produkttests durchführt. Bob untersucht die Abfragen der Testabteilung während des Produkttestzyklus und erkennt, dass die Abfragen komplizierte Joins mit großen Datenmengen enthalten. Diese Abfragen generieren eine beträchtliche Vorablesezugriffsaktivität, wodurch die Verarbeitung von Vorablesezugriffsanforderungen von Verbindungen anderer Abteilungen verhindert wird. Bob beschließt, die Vorablesezugriffspriorität der Verbindungen von der Testabteilung zu verringern, und ändert die Vorablesezugriffspriorität der Serviceklasse TESTING auf LOW:

```
ALTER SERVICE CLASS TESTING PREFETCH PRIORITY LOW
```

Die Workload-Management-Konfiguration sieht nun wie folgt aus:

Tabelle 19. Konfiguration nach dem Ändern der Vorablesezugriffspriorität für die Servicesuperklasse TESTING

Serviceklasse	Schwellenwert MAXSERVICECLASSCONNECTIONS	Vorablesezugriffspriorität
SALES	Nicht zutreffend	DEFAULT
ACCOUNTING	Nicht zutreffend	DEFAULT
ENGINEERING	Nicht zutreffend	DEFAULT
TESTING	300	LOW
PRODUCTION	Nicht zutreffend	DEFAULT
SYSDEFAULTUSERCLASS	Nicht zutreffend	DEFAULT
SYSDEFAULTMAINTENANCECLASS	Nicht zutreffend	DEFAULT

Indem die Vorablesezugriffspriorität der Serviceklasse TESTING auf LOW gesetzt wird, werden Vorablesezugriffsanforderungen von Verbindungen von der Testabteilung erst bedient, nachdem alle anderen Vorablesezugriffsanforderungen von den anderen Abteilungen verarbeitet wurden.

Diese Änderung erhöht den Abfragedurchsatz der anderen Abteilungen und verringert den Durchsatz der Testabteilung in ihrer Produkttestphase.

Dritte Verbesserung der Workload-Management-Implementierung

Nachdem das Problem mit dem Vorablesezugriff behoben ist, teilt die Entwicklungsabteilung Bob mit, dass sie einige Verbindungen für eine experimentelle Anwendung namens Brewmeister benötigt. Da es sich um eine experimentelle Anwendung handelt, will Bob sicherstellen, dass sie nicht zu viele Datenbankverbindungen verbraucht und dass Abfragen von der Anwendung nicht um Vorablesefunktionen konkurrieren, wenn das System ausgelastet ist. Um diese Ziele zu erreichen, erstellt er eine neue Serviceunterklasse unter der Servicesuperklasse ENGINEERING für die experimentelle Anwendung sowie eine Workload, um Verbindungen von der Anwendung der neuen Serviceunterklasse zuzuordnen. Bob aktualisiert die Serviceklasse und die Workloads wie folgt:

- Die Serviceunterklasse EXPERIMENT wird unter der Servicesuperklasse ENGINEERING erstellt:

```
CREATE SERVICE CLASS EXPERIMENT UNDER ENGINEERING
```
- Der Schwellenwert MAXSERVICECLASSCONNECTIONS von 50 wird für die Serviceunterklasse EXPERIMENT erstellt:

```
CREATE THRESHOLD MAXSERVICECLASSCONNECTIONS FOR SERVICE CLASS EXPERIMENT UNDER ENGINEERING ACTIVITIES ENFORCEMENT DATABASE WHEN TOTALDBPARTITIONCONNECTIONS > 50 STOP EXECUTION
```
- Die Workload WL_EXPERIMENT wird erstellt, um Verbindungen von der Anwendung BREWMEISTER der Serviceunterklasse EXPERIMENT zuzuordnen:

```
CREATE WORKLOAD WL_EXPERIMENT APPLNAME ('BREWMEISTER') SERVICE CLASS EXPERIMENT UNDER ENGINEERING
```
- Die Vorablesezugriffspriorität für die Serviceunterklasse EXPERIMENT wird auf LOW gesetzt:

```
ALTER SERVICE CLASS EXPERIMENT UNDER ENGINEERING PREFETCH PRIORITY LOW
```

Die Workload-Management-Konfiguration sieht nun wie folgt aus:

Tabelle 20. Konfiguration mit Serviceunterklasse EXPERIMENT

Serviceklasse	Schwellenwert MAXSERVICECLASSCONNECTIONS	Vorablesezugriffspriorität
SALES	Nicht zutreffend	DEFAULT
ACCOUNTING	Nicht zutreffend	DEFAULT
ENGINEERING	Nicht zutreffend	DEFAULT
EXPERIMENT	50	LOW
TESTING	300	LOW
PRODUCTION	Nicht zutreffend	DEFAULT
SYSDEFAULTUSERCLASS	Nicht zutreffend	DEFAULT
SYSDEFAULTMAINTENANCECLASS	Nicht zutreffend	DEFAULT

Mit dieser Konfiguration kann die Anwendung BREWMEISTER nur 50 gleichzeitig bestehende Verbindungen zur Datenbank halten. Darüber hinaus werden Vorablesezugriffsanforderungen von dieser Anwendung an die Vorablesewarteschlange mit niedriger Priorität gesendet. Die Entwicklungsabteilung kann jetzt auf sichere Weise mit der Anwendung experimentieren und kann davon ausgehen, dass nicht unbeabsichtigt das gesamte Datenbanksystem überlastet wird.

Beispiel: Workloadzuordnung

Am Anfang der ersten UOW (Unit of Work - Arbeitseinheit) nach dem Herstellen der Datenbankverbindung ordnet der Datenserver die Verbindung einer Workload zu, indem er die Verbindungsattribute jeder aktivierten Workload auswertet. Die Workloadneuauswertung erfolgt am Anfang jeder UOW, wenn der Wert eines Verbindungsattributs oder der Workloaddefinition selbst sich innerhalb der UOW ändert.

Die folgende Abbildung zeigt eine Workloadzuordnung. Benutzer in der Gruppe Marketing, die Abfragen über AppA übergeben, werden der Workload APPAQUERIES zugeordnet. Obwohl PAYROLL vor APPAQUERIES positioniert ist, werden sie nicht der Workload PAYROLL zugeordnet, da die Definition der Workload PAYROLL für den Schlüsselwert SESSION_USER GROUP den Wert Finance definiert. Benutzer in der Gruppe Finance, die Abfragen über AppA übergeben, werden der Workload FINANCE zugeordnet. Sie werden nicht der Workload PAYROLL zugeordnet, obwohl diese genauer ist und in ihrer Definition sowohl AppA als auch Finance angibt, da die Workload FINANCE vor der Workload PAYROLL positioniert ist. Benutzer in der Gruppe Marketing, die Abfragen über AppB übergeben, werden der Workload SYSDEFAULTUSERWORKLOAD zugeordnet, da keines der Verbindungsattribute, die in den Definitionen der Workloads FINANCE, PAYROLL oder APPAQUERIES angegeben sind, der Anwendung AppB oder der Gruppe Marketing entspricht.

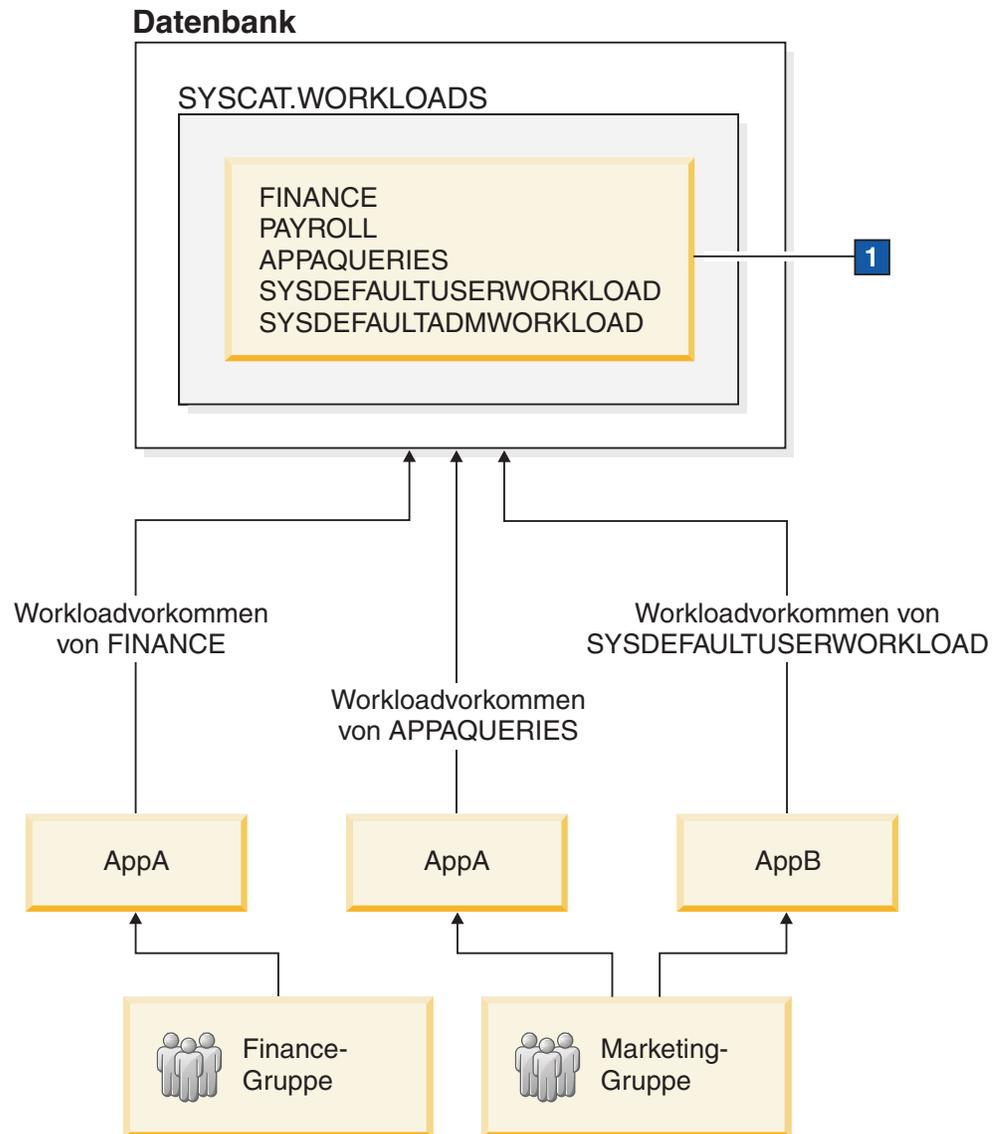


Abbildung 25. Beispiel einer Workloadzuordnung

1 In der Abbildung oben lauten die Anweisungen CREATE WORKLOAD folgendermaßen:

```
CREATE WORKLOAD PAYROLL APPLNAME ('AppA') SESSION_USER GROUP ('FINANCE')
SERVICE CLASS SC1
```

```
CREATE WORKLOAD APPAQUERIES APPLNAME('AppA') POSITION LAST SERVICE CLASS SC2
```

```
CREATE WORKLOAD FINANCE SESSION_USER GROUP ('FINANCE') SERVICE CLASS SC1 POSITION
BEFORE PAYROLL
```

In einer dreistufigen Client/Server-Umgebung wird die Datenbankverbindung vom Anwendungsserver für die Clients hergestellt. Der Anwendungsserver kann die API `sqleseti` (Clientinformationen festlegen) verwenden, um Clientinformationen an den DB2-Datenserver zu übergeben; andernfalls werden nur die Informationen vom Anwendungsserver übergeben, und diese Informationen sind wahrscheinlich gleich für alle Clientanforderungen, die über diesen Anwendungsserver geleitet werden. Wenn der Datenserver UOWs von unterschiedlichen Clients unterschiedlichen Workloads (und unterschiedlichen Serviceklassen) zuordnet, verwendet er die Clientinformationsattribute (d. h. Clientbenutzer-ID, Clientanwendungsname, Clientworkstationname und Clientabrechnungszeichenfolge) als Bedingungen für die Zuordnung einer UOW zu einer Workload.

Die folgende Abbildung zeigt ein Beispiel einer dreistufigen Umgebung, in der Abfragen von unterschiedlichen Benutzeranwendungen (`marketing.exe`, `auditing.exe` und `reporting.exe`) über einen Anwendungsserver übergeben werden, der eine Verbindung zur Datenbank mit dem Sitzungsbenutzer `APPUSER` herstellt. Drei Workloads sind definiert: eine für Abfragen, die von `marketing.exe` übergeben werden, eine für Abfragen, die von `reporting.exe` übergeben werden, und eine für die restlichen Abfragen. Wie in der Abbildung dargestellt, ruft der Anwendungsserver zum Zuordnen von Abfragen von `marketing.exe` die API `sqleseti` auf, um den Wert des Sonderregisters `CURRENT CLIENT_APPLNAME` auf `marketing.exe` einzustellen. Analog dazu ruft der Server zum Zuordnen von Abfragen von `reporting.exe` zur Workload `REPORTING` die API `sqleseti` auf, um den Wert des Sonderregisters `CURRENT CLIENT_APPLNAME` auf `reporting.exe` einzustellen. Beachten Sie in der Abbildung, dass keine erneute Workloadzuordnung erfolgt, wenn der Server die API `sqleseti` aufruft, um das Sonderregister `CURRENT CLIENT_USERID` auf `Lidia` einzustellen (ohne jegliche andere Änderungen, d. h. der Clientanwendungsname ist noch auf `reporting.exe` eingestellt). Dies liegt daran, dass es keine Workload gibt, die spezifisch mit der Einstellung `Lidia` für `CURRENT CLIENT_USERID` definiert ist.

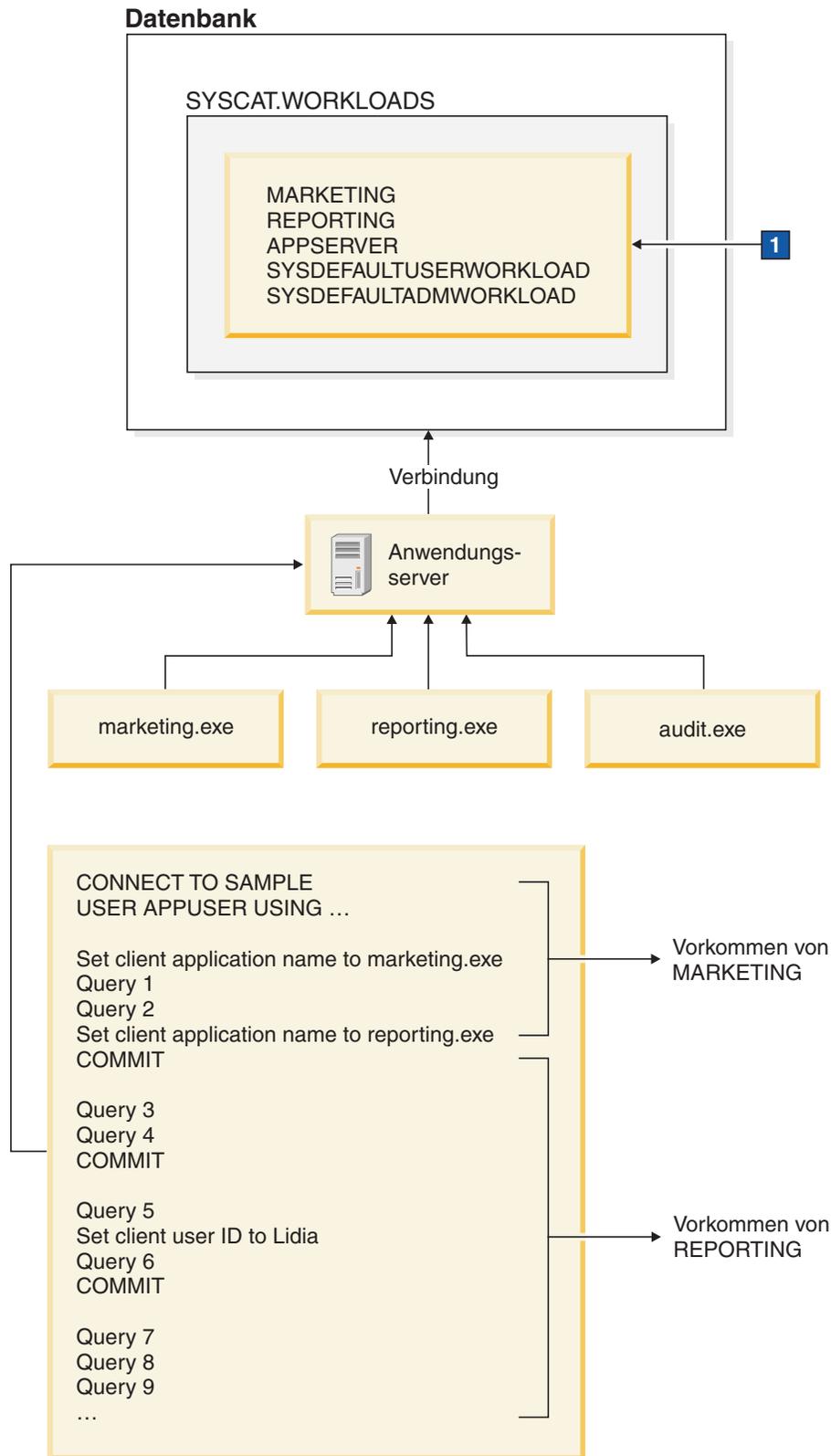


Abbildung 26. Beispiel einer Workloadzuordnung in einer dreistufigen Umgebung

Die folgenden Anweisungen werden verwendet, um die im Feld **1** in der Abbildung oben angegebenen Workloads zu definieren:

```
CREATE WORKLOAD MARKETING SESSION_USER ('APPUSER')
CURRENT CLIENT_APPLNAME ('marketing.exe') SERVICE CLASS SC2
POSITION AT 1
```

```
CREATE WORKLOAD REPORTING SESSION_USER ('APPUSER')
CURRENT CLIENT_APPLNAME ('reporting.exe') SERVICE CLASS SC4
POSITION AFTER MARKETING
```

```
CREATE WORKLOAD APPSERV SESSION_USER ('APPUSER')
SERVICE CLASS SC1
```

Beispiel: Workloadzuordnung, wenn Workloadattribute Einzelwerte haben

Das Beispiel in diesem Thema zeigt, wie der Datenserver Workloads zuordnet. In diesem Beispiel ist für jedes Workloadverbindungsattribut nur ein Wert angegeben.

Nehmen wir an, dass im Katalog die folgenden Workloads vorhanden sind:

Tabelle 21. Workloads im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER_GROUP	SESSION_USER_ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNAME	CURRENT_CLIENT_ACCTNG
1	REPORTS	AppA								
2	INVENTORY REPORT	AppB	LYNN		ACCOUNTING	TELEMKTR				
3	SALES REPORT	AppC	KATE	KATE		SALESREP				
4	AUDIT REPORT	AppB			ACCOUNTING	FINANALYST				
5	EXPENSE REPORT	AppA	TIM			EXPENSE APPROVER				
6	AUDIT RESULT			LYNN			LYNN			Audit Group

Nehmen wir an, dass eine Datenbankverbindung mit den folgenden Attributen hergestellt wurde:

Tabelle 22. Datenbankverbindungsattribute

APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER_GROUP	SESSION_USER_ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNAME	CURRENT_CLIENT_ACCTNG
AppA	TIM	TIM	FINANCE	FINANALYST, EXPENSE APPROVER	NULL	NULL	NULL	Business account

Bei der Übergabe der ersten UOW (Unit of Work - Arbeitseinheit) prüft der Datenserver jede Workload im Katalog, beginnend mit der ersten Workload in der Liste, und verarbeitet die Workloads in aufsteigender Reihenfolge, bis er eine Workload mit übereinstimmenden Attributen findet. Wenn eine übereinstimmende Workload gefunden wurde, wird die UOW unter einem Vorkommen dieser Workload ausgeführt. Bei der Ermittlung, welcher Workload die Verbindung zugeordnet werden soll, vergleicht der Datenserver die Verbindungsattribute in einer definierten Reihenfolge.

Der Datenserver prüft zuerst die Workload REPORTS auf Übereinstimmung. Die Workload REPORTS ist die erste Workload in der Liste.

Tabelle 23. Workload REPORTS im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER_GROUP	SESSION_USER_ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNAME	CURRENT_CLIENT_ACCTNG
1	REPORTS	AppA								

Der Datenserver prüft die Verbindungsattribute in der folgenden definierten Reihenfolge:

1. APPLNAME. Der Wert von APPLNAME, AppA, für die Datenbankverbindung entspricht dem Wert von APPLNAME für die Workload REPORTS.
2. SYSTEM_USER, das in der Workloaddefinition nicht angegeben ist. Jeder Wert (einschließlich eines Nullwerts) wird als Übereinstimmung betrachtet.
3. SESSION_USER, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.
4. SESSION_USER GROUP, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.
5. SESSION_USER ROLE, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.
6. CURRENT CLIENT_USERID, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.
7. CURRENT CLIENT_APPLNAME, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.
8. CURRENT CLIENT_WRKSTNNAME, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.
9. CURRENT CLIENT_ACCTNG, das in der Workloaddefinition nicht angegeben ist. Jeder Wert wird als Übereinstimmung betrachtet.

Auf Grund der expliziten und impliziten Übereinstimmungen zwischen den Verbindungsattributen der Workload REPORTS und den von der Verbindung übergebenen Informationen wählt der Datenserver in dieser Situation die Workload REPORTS als potenzielle Übereinstimmung aus. Nach der Auswahl einer Workload prüft der Datenserver dann, ob der Sitzungsbenutzer das Zugriffsrecht USAGE für die Workload hat. Nehmen wir an, dass der Sitzungsbenutzer TIM das Zugriffsrecht USAGE für die Workload REPORTS hat. Diese Workload wird dann für die Verbindung verwendet. Wenn jedoch TIM nicht das Zugriffsrecht USAGE für die Workload REPORTS hat, fährt der Datenserver fort, indem er die Workload INVENTORYREPORT auf eine Übereinstimmung überprüft.

Angenommen, TIM soll der Workload EXPENSEREPORT zugeordnet werden, da für diese Workload zusätzliche Verbindungsattribute angegeben sind. In dieser Situation würden Sie die Auswertungsreihenfolge der Workloads ändern, so dass EXPENSEREPORT vor REPORTS in der Workloadliste steht:

```
ALTER WORKLOAD EXPENSEREPORT POSITION AT 1
```

Mit der folgenden SQL-Anweisung könnten Sie dasselbe Ergebnis erzielen:

```
ALTER WORKLOAD EXPENSEREPORT BEFORE REPORTS
```

Damit die Anweisung ALTER WORKLOAD wirksam wird, müssen Sie sofort nach der Anweisung ALTER WORKLOAD eine Anweisung COMMIT absetzen. Die Anweisung ALTER WORKLOAD hat die folgende Auswirkung auf den Katalog:

Tabelle 24. Workloads im Katalog nach der Neupositionierung der Workload EXPENSEREPORT

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
1	EXPENSE REPORT	AppA	TIM			EXPENSE APPROVER				
2	REPORTS	AppA								
3	INVENTORY REPORT	AppB	LYNN		ACCOUNTING	TELEMKTR				
4	SALES REPORT	AppC	KATE	KATE		SALESREP				
5	AUDIT REPORT	AppB			ACCOUNTING	FINANALYST				
6	AUDIT RESULT			LYNN			LYNN			Audit Group

Wenn TIM noch nicht das Zugriffsrecht USAGE für die Workload EXPENSEREPORT hat, müssen Sie die folgenden Anweisungen absetzen (die Anweisung COMMIT stellt sicher, dass die Anweisung GRANT wirksam wird):

```
GRANT USAGE ON WORKLOAD EXPENSEREPORT TO USER TIM
COMMIT
```

Am Anfang der nächsten UOW erfolgt die erneute Workloadzuordnung, und der Datenserver ordnet die Verbindung von TIM der Workload EXPENSEREPORT zu. Außerdem werden auch neue UOWs, die von anderen Verbindungen mit denselben Attributen übergeben werden, der Workload EXPENSEREPORT zugeordnet.

Beispiel: Workloadzuordnung für eine UOW, wenn mehrere Workloads vorhanden sind

Das Beispiel in diesem Thema zeigt, wie der Datenserver die Workloadauswertung durchführt, um die Verbindung einer vorhandenen Workload zuzuordnen.

Nehmen wir an, dass die folgenden Workloads im Katalog definiert sind:

Tabelle 25. Workloads im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER_GROUP	SESSION_USER_ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
1	EXPENSE REPORT	AppB	TIM			EXPENSE APPROVER				
2	REPORTS	AppB								
3	INVENTORYREPORT	AppA	LYNN		ACCOUNTING	TELEMKTR				
4	SALES REPORT	AppC	KATE	KATE		SALESREP				
5	AUDIT REPORT	AppA			ACCOUNTING	FINANALYST				
6	AUDIT RESULT			LYNN			LYNN			Audit Group

Nehmen wir an, dass eine Datenbankverbindung mit den folgenden Attributen hergestellt wird:

Tabelle 26. Datenbankverbindungsattribute

APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER_GROUP	SESSION_USER_ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
AppA	LYNN	LYNN	ACCOUNTING	FINANALYST, SALESREP	LYNN	NULL	wrkstrn2	Audit group

Bei der Übergabe der ersten UOW (Unit of Work - Arbeitseinheit) prüft der Datenserver jede Workload im Katalog in aufsteigender Auswertungsreihenfolge und stoppt, wenn er eine Workload findet, deren Verbindungsattribute den Verbindungsattributen entsprechen, die von der Verbindung angegeben werden. Beim Prüfen der Workloads vergleicht der Datenserver die Verbindungsattribute in einer definierten Reihenfolge.

Zuerst prüft der Datenserver die Workload EXPENSEREPORT:

Tabelle 27. Workload EXPENSEREPORT im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER_GROUP	SESSION_USER_ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
1	EXPENSEREPORT	AppB	TIM			EXPENSE APPROVER				

Da das Attribut APPLNAME in der Workloaddefinition AppB lautet, das von der Verbindung übergebene Attribut APPLNAME jedoch AppA lautet, ist keine Übereinstimmung möglich. Der Datenserver fährt mit der Workload REPORTS fort, dem zweiten Eintrag in der Liste:

Tabelle 28. Workload REPORTS im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT CLIENT_USERID	CURRENT CLIENT_APPLNAME	CURRENT CLIENT_WRKSTNNAME	CURRENT CLIENT_ACCTNG
2	REPORTS	AppB								

Auch hier lautet das Attribut APPLNAME in der Workloaddefinition AppB und bietet keine Übereinstimmung mit AppA. Der Datenserver fährt mit der dritten Workload in der Liste fort, der Workload INVENTORYREPORT:

Tabelle 29. Workload INVENTORYREPORT im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT CLIENT_USERID	CURRENT CLIENT_APPLNAME	CURRENT CLIENT_WRKSTNNAME	CURRENT CLIENT_ACCTNG
3	INVENTORYREPORT	AppA	LYNN		ACCOUNTING	TELEMKTR				

Der Datenserver prüft auf eine Übereinstimmung zwischen den übergebenen Verbindungsattributen und der Workload INVENTORYREPORT. Die Attribute werden in der folgenden Reihenfolge geprüft:

1. APPLNAME. Sowohl die Workloaddefinition als auch die Verbindung haben den Wert AppA. Es gibt also eine Übereinstimmung.
2. SYSTEM_USER. Sowohl die Workloaddefinition als auch die Verbindung haben den Wert LYNN. Es gibt also eine Übereinstimmung.
3. SESSION_USER. Die Verbindung übergab den Wert LYNN. Da das Attribut SESSION_USER für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert, einschließlich eines Nullwerts, als Übereinstimmung.
4. SESSION_USER GROUP. Sowohl die Workloaddefinition als auch die Verbindung haben den Wert ACCOUNTING. Es gibt also eine Übereinstimmung.
5. SESSION_USER ROLE. Die Workloaddefinition gibt den Wert TELEMKTR an. Die Verbindung übergab jedoch die Werte FINANALYST und SALESREP. Für dieses Attribut gibt es keine Übereinstimmung.

Der Datenserver versucht nicht weiter, die Workload INVENTORYREPORT mit den Verbindungsattributen abzugleichen, und fährt mit der vierten Workload in der Liste fort, der Workload SALESREPORT:

Tabelle 30. Workload SALESREPORT im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT CLIENT_USERID	CURRENT CLIENT_APPLNAME	CURRENT CLIENT_WRKSTNNAME	CURRENT CLIENT_ACCTNG
4	SALESREPORT	AppC	KATE	KATE		SALESREP				

Da das Attribut APPLNAME der Workloaddefinition von SALESREPORT den Wert AppC hat, gibt es keine Übereinstimmung mit der Verbindung (die den Wert AppA für APPLNAME übergab). Der Datenserver fährt dann mit der fünften Workload in der Liste fort, der Workload AUDITREPORT:

Tabelle 31. Workload AUDITREPORT im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT CLIENT_USERID	CURRENT CLIENT_APPLNAME	CURRENT CLIENT_WRKSTNNAME	CURRENT CLIENT_ACCTNG
5	AUDITREPORT	AppA			ACCOUNTING	FINANALYST				

Der Datenserver vergleicht die Attribute der Workload AUDITREPORT und der Verbindung in der definierten Reihenfolge:

1. APPLNAME. Sowohl die Workloaddefinition als auch die Verbindung haben den Wert AppA. Es gibt also eine Übereinstimmung.

2. SYSTEM_USER. Die Verbindung übergab den Wert LYNN. Da das Attribut SYSTEM_USER für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
3. SESSION_USER. Die Verbindung übergab den Wert LYNN. Da das Attribut SESSION_USER für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
4. SESSION_USER GROUP. Sowohl die Workload als auch die Verbindung haben den Wert ACCOUNTING für dieses Attribut. Es gibt also eine Übereinstimmung.
5. SESSION_USER ROLE. Sowohl die Workload als auch die Verbindung haben den Wert FINANALYST für dieses Attribut. Es gibt also eine Übereinstimmung.
6. CURRENT CLIENT_USERID. Da das Attribut CURRENT CLIENT_USERID für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
7. CURRENT CLIENT_APPLNAME. Da das Attribut CURRENT CLIENT_APPLNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
8. CURRENT CLIENT_WRKSTNNAME. Da das Attribut CURRENT CLIENT_WRKSTNNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
9. CURRENT CLIENT_ACCTNG. Da das Attribut CURRENT CLIENT_ACCTNG für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.

Nachdem der Datenserver alle Verbindungsattribute verarbeitet und eine übereinstimmende Workload gefunden hat, prüft er, ob der Sitzungsbenutzer das Zugriffsrecht USAGE für die Workload hat. Nehmen wir an, dass LYNN das Zugriffsrecht USAGE für die Workload AUDITREPORT nicht besitzt. In dieser Situation wird diese Workload der Verbindung nicht zugeordnet, obwohl alle Verbindungsattribute übereinstimmen. Der Datenserver fährt mit der sechsten Workload in der Auswertungsliste fort, der Workload AUDITRESULT:

Tabelle 32. Workload AUDITRESULT im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT CLIENT_USERID	CURRENT CLIENT_APPLNAME	CURRENT CLIENT_WRKSTNNAME	CURRENT CLIENT_ACCTNG
6	AUDITRESULT			LYNN			LYNN			Audit Group

Der Datenserver vergleicht die Attribute der Workload AUDITRESULT und der Verbindung in der definierten Reihenfolge:

1. APPLNAME. Da das Attribut APPLNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
2. SYSTEM_USER. Da das Attribut SYSTEM_USER für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
3. SESSION_USER. Sowohl die Workload als auch die Verbindung haben den Wert LYNN für dieses Attribut. Es gibt also eine Übereinstimmung.
4. SESSION_USER GROUP. Da das Attribut SESSION_USER GROUP für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
5. SESSION_USER ROLE. Da das Attribut SESSION_USER ROLE für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
6. CURRENT CLIENT_USERID. Sowohl die Workload als auch die Verbindung haben den Wert LYNN für dieses Attribut. Es gibt also eine Übereinstimmung.

7. CURRENT CLIENT_APPLNAME. Da das Attribut CURRENT CLIENT_APPLNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
8. CURRENT CLIENT_WRKSTNNAME. Da das Attribut CURRENT CLIENT_WRKSTNNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
9. CURRENT CLIENT_ACCTNG. Sowohl die Workload als auch die Verbindung haben den Wert Audit Group für dieses Attribut. Es gibt also eine Übereinstimmung.

Nachdem der Datenserver alle Verbindungsattribute verarbeitet und eine übereinstimmende Workload gefunden hat, prüft er, ob der Sitzungsbenutzer das Zugriffsrecht USAGE für die Workload hat. Nehmen wir in dieser Situation an, dass der Sitzungsbenutzer LYNN das Zugriffsrecht USAGE für die Workload AUDITRESULT besitzt. Da alle Verbindungsattribute übereinstimmen und der Sitzungsbenutzer das Zugriffsrecht USAGE hat, wird die Verbindung der Workload AUDITRESULT zugeordnet.

Beispiel: Workloadzuordnung, wenn Workloadattribute mehrere Werte haben

Das Beispiel in diesem Thema zeigt, wie der Datenserver Workloads zuordnet. In diesem Beispiel lassen einige der Workloaddefinitionen mehrere Werte für ein Verbindungsattribut zu.

Nehmen wir an, dass die folgenden Workloads im Katalog definiert sind:

Tabelle 33. Workloads im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
1	ITEMINQ		KYLE, GEORGE		RETAIL, SALES					
2	DAILY TRANS REPORT	AppC		KYLE, CAROL	SALES, ACCOUNTING					
3	SALES SUMMARY	AppA, AppB				ACCOUNTANT, FINANALYST				

Nehmen wir an, dass eine Datenbankverbindung mit den folgenden Attributen hergestellt wurde:

Tabelle 34. Datenbankverbindungsattribute

APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
AppC	LINDA	KYLE	SALES	ACCOUNTANT	LINDA	NULL	NULL	Business Account

Bei der Übergabe der ersten UOW (Unit of Work - Arbeitseinheit) prüft der Datenserver jede Workload im Katalog in aufsteigender Auswertungsreihenfolge und stoppt, wenn er eine Workload findet, deren Verbindungsattribute den Verbindungsattributen entsprechen, die von der Verbindung angegeben werden. Beim Prüfen der Workloads vergleicht der Datenserver die Verbindungsattribute in einer definierten Reihenfolge.

Zuerst prüft der Datenserver die Workload ITEMINQ:

Tabelle 35. Workload ITEMINQ im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT_CLIENT_USERID	CURRENT_CLIENT_APPLNAME	CURRENT_CLIENT_WRKSTNNAME	CURRENT_CLIENT_ACCTNG
1	ITEMINQ		KYLE, GEORGE		RETAIL, SALES					

Der Datenserver prüft auf eine Übereinstimmung zwischen den übergebenen Verbindungsattributen und der Workload ITEMNQ. Die Attribute werden in der folgenden Reihenfolge geprüft:

1. APPLNAME. Da das Attribut APPLNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert, einschließlich eines Nullwerts, als Übereinstimmung.
2. SYSTEM_USER. Die Verbindung übergab den Wert LINDA. Die Workload ITEMNO hat jedoch die Werte KYLE und GEORGE. Für dieses Attribut gibt es keine Übereinstimmung.

Der Datenserver versucht nicht weiter, die Workload ITEMNMO mit der Verbindung abzugleichen, und fährt mit der zweiten Workload in der Liste fort, der Workload DAILYTRANSREPORT:

Tabelle 36. Workload DAILYTRANSREPORT im Katalog

Auswertungsreihenfolge	Workloadname	APPLNAME	SYSTEM_USER	SESSION_USER	SESSION_USER GROUP	SESSION_USER ROLE	CURRENT CLIENT_USERID	CURRENT CLIENT_APPLNAME	CURRENT CLIENT_WRKSTNNAME	CURRENT CLIENT_ACCTNG
2	DAILYTRANSREPORT	AppC		KYLE, CAROL	SALES, ACCOUNTING					

Der Datenserver vergleicht die Attribute der Workload DAILYTRANSREPORT und der Verbindung in definierter Reihenfolge:

1. APPLNAME. Sowohl die Workloaddefinition als auch die Verbindung haben den Wert AppC. Es gibt also eine Übereinstimmung.
2. SYSTEM_USER. Da das Attribut SYSTEM_USER für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert, einschließlich eines Nullwerts, als Übereinstimmung.
3. SESSION_USER. Der von der Verbindung übergebene Wert SESSION_USER lautet KYLE. Dies ist eine Übereinstimmung mit einem der Werte der Workload SESSION_USER. Hätte die Verbindung CAROL übergeben, wäre auch dies eine Übereinstimmung, da sowohl KYLE als auch CAROL als Teil der Workloaddefinition von DAILYTRANSREPORT angegeben sind.
4. SESSION_USER GROUP. Der von der Verbindung übergebene Wert SESSION_USER GROUP lautet SALES. Dieser Wert entspricht dem Wert SALES, der für das Attribut der Workload SESSION_USER GROUP angegeben ist. Hätte die Verbindung ACCOUNTING übergeben, wäre auch dies eine Übereinstimmung, da sowohl SALES als auch ACCOUNTING in der Workloaddefinition angegeben sind.
5. SESSION_USER ROLE. Da das Attribut SESSION_USER ROLE für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
6. CURRENT CLIENT_USERID. Da das Attribut CURRENT CLIENT_USERID für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
7. CURRENT CLIENT_APPLNAME. Da das Attribut CURRENT CLIENT_APPLNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
8. CURRENT CLIENT_WRKSTNNAME. Da das Attribut CURRENT CLIENT_WRKSTNNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.
9. CURRENT CLIENT_ACCTNG. Da das Attribut CURRENT CLIENT_WRKSTNNAME für die Workload nicht festgelegt ist, gilt jeder von der Verbindung übergebene Wert als Übereinstimmung.

Nachdem der Datenserver alle Verbindungsattribute verarbeitet und eine übereinstimmende Workload für die Verbindung gefunden hat, prüft er, ob der Sitzungsbenuer das Zugriffsrecht USAGE für die Workload hat. Nehmen wir in dieser Situation an, dass der Sitzungsbenuer KYLE das Zugriffsrecht USAGE für die Workload DAILYTRANSREPORT hat. Da alle Verbindungsattribute übereinstimmen und der Sitzungsbenuer das Zugriffsrecht USAGE hat, wird die Verbindung der Workload DAILYTRANSREPORT zugeordnet.

Beispiel: Verwenden von Schwellenwerten

Sie können Schwellenwerte für eine Vielzahl von Zwecken verwenden. In dem Szenario in diesem Thema werden Schwellenwerte verwendet, um die Anzahl ausgeführter großer Jobs zu steuern, unterschiedliche Ausführungszeiten für unterschiedliche Anwendungen zuzulassen und das Verhalten einer in Entwicklung befindlichen Anwendung zu steuern.

Eine Möglichkeit, eine Workload-Management-Lösung einzurichten, besteht darin, die Datenbankressourcen über die verschiedenen Abteilungen in einem Unternehmen aufzuteilen und zu verwalten. Nehmen wir z. B. an, dass die Vertriebsabteilung zwei Hauptberichte führt, die aus den monatlichen und den jährlichen Verkäufen bestehen. Nehmen wir weiter an, dass die Personalabteilung alle zwei Wochen eine Anwendung der Lohnbuchhaltung ausführt und dass das Entwicklungsteam auf Wunsch des Managements an einem neuen Typ von Bericht arbeitet.

In dieser Situation erstellen Sie für jede dieser Anwendungen eine Workloaddefinition, um die jeweilige Anwendung der zutreffenden Servicesuperklasse zuzuordnen. Die Datenbankkataloge enthalten daher die folgenden Workloaddefinitionen:

- MonthlySales, der Servicesuperklasse Sales zugeordnet
- YearlySales, der Servicesuperklasse Sales zugeordnet
- Payroll, der Servicesuperklasse Human Resources zugeordnet
- NewReport, der Servicesuperklasse Development zugeordnet

Schwellenwert für die Anzahl großer Jobs

Da der Bericht YearlySales sehr groß ist, soll nicht mehr als ein Vorkommen dieser Anwendung auf einmal in der Datenbank ausgeführt werden. Sie erstellen daher einen Schwellenwert, um die maximale Anzahl gleichzeitiger Vorkommen dieser Workload auf 1 zu setzen.

Sie können eine ähnliche Lösung erzielen, indem Sie die Anwendung YearlySales mit einer Serviceunterklasse YearlySalesReports (unter der Servicesuperklasse Sales) verknüpfen und den Schwellenwert für maximalen gemeinsamen Zugriff für die Serviceunterklasse auf 1 setzen.

In beiden Fällen können Sie die Schwellenwertaktion auf STOP EXECUTION setzen, um zu verhindern, dass mehr als ein Vorkommen der Workload ausgeführt wird. Sie können auch Aktivitätsinformationen erfassen, wenn Sie weitere Informationen zu den Bedingungen erfassen wollen, unter denen ein Schwellenwertverstoß auftritt.

Schwellenwert für die Aktivitätslaufzeiten

Da erwartet wird, dass alle Anwendungen in maximal einer Stunde ausgeführt werden, erstellen Sie einen Schwellenwert mit einer Datenbankdomäne, der nicht zulässt, dass eine Aktivität länger als 1 Stunde ausgeführt wird. Die einzige Aus-

nahme dieser Regel ist der Jahresbericht, dessen Ausführung bis zu 5 Stunden dauern kann. Sie erstellen daher eine neue Serviceunterklasse unter der Servicesuperklasse Sales und nennen diese YearlySalesReports. Dann ordnen Sie die Workload YearlySales dieser Serviceunterklasse zu, erstellen einen Schwellenwert für die Gesamtzeit der Aktivitäten von 5 Stunden und weisen diesen Schwellenwert der Serviceunterklasse zu. Der neue Wert von 5 Stunden gilt jetzt für die Serviceunterklasse YearlySalesReports. Für alle anderen Elemente in der Datenbank gilt jedoch weiterhin der globale Wert von 1 Stunde.

Schwellenwert für die Anzahl der Koordinator- und verschachtelten Aktivitäten

Die Anwendung NewReport nutzt sehr viele gespeicherte Prozeduren und benutzerdefinierte Funktionen und ist noch nicht vollständig getestet. Sie erzeugt daher große Mengen von Aktivitäten, die sich auf den Rest des Systems auswirken. Nach Beratung mit dem Entwickler erfahren Sie, dass dieser neue Bericht nicht mehr als 20 Aktivitäten insgesamt generieren sollte. Sie definieren daher einen Schwellenwert für Workloadaktivitäten in der Workload NewReport und setzen diesen auf 20. Anfangs legen Sie als Schwellenwertaktion STOP EXECUTION und COLLECT ALL fest, um unerwünschte Nebeneffekte durch das Starten sehr vieler Aktivitäten auszuschließen und dem Entwickler bei der Problemerkennung zu helfen.

Wenn die Anwendung stabiler wird, kommt sie in die Optimierungsphase. In dieser Phase versucht der Entwickler, die Anzahl der von der Anwendung generierten Aktivitäten von 15 bis 20 auf 15 zu verringern. Jetzt ändern Sie den Schwellenwert, indem Sie die obere Begrenzung auf 15 und die Schwellenwertaktion auf CONTINUE setzen. Diese Schwellenwertdefinition hilft bei der Ermittlung und Behebung von Situationen, in denen die Anzahl der generierten Aktivitäten 15 überschreitet, die höhere Stabilität der Anwendung aber nicht das Stoppen der Ausführung erforderlich macht.

Beispiel: Schwellenwerte CONCURRENTWORKLOADOCCURRENCES, TOTALDBPARTITIONCONNECTIONS und TOTALSPARTITIONCONNECTIONS

Das Beispiel in diesem Abschnitt zeigt, wie die zusammengefassten Schwellenwerte CONCURRENTWORKLOADOCCURRENCES, TOTALDBPARTITIONCONNECTIONS und TOTALSPARTITIONCONNECTIONS interagieren.

Nehmen wir an, dass eine Datenbankverbindung auf der Datenbankpartition 1 hergestellt wird. Der Schwellenwert TOTALDBPARTITIONCONNECTIONS wird einmal ausgewertet, um zu entscheiden, ob diese Verbindung eine Verbindung zur Datenbank herstellen darf. Wenn die Auswertung dieses Schwellenwerts erfolgreich ist, wird TOTALDBPARTITIONCONNECTIONS nicht mehr für dieselbe Verbindung ausgewertet, die jetzt in der Datenbank ist. Die Verbindung wechselt mehrmals zwischen Workload A und Workload B, da die relevanten Verbindungsattribute geändert wurden. Bei jedem Wechsel der Verbindung zwischen Workloads wird die Verbindung möglicherweise in eine andere Serviceklasse versetzt. Wenn ein Vorkommen einer der beiden Workloads gestartet wird, wird der Schwellenwert CONCURRENTWORKLOADOCCURRENCES ausgewertet. Wenn nicht gegen diesen Schwellenwert verstoßen wird, führt die Verbindung einen Join zur entsprechenden Serviceklasse durch. Zu diesem Zeitpunkt wird der letzte Schwellenwert

(TOTALSPARTITIONCONNECTIONS) ausgewertet. Jetzt wird die Verbindung eventuell vor dem Eingang in die Serviceklasse in die Warteschlange gestellt, bevor sie Arbeit absetzen kann.

Beispiel: Verwalten bestimmter Aktivitätstypen mit Hilfe von Arbeitsklassensets

Das folgende Beispiel zeigt, wie mit Hilfe eines Arbeitsklassensets CALL- und DML-Aktivitäten verwaltet werden können.

Nehmen wir an, dass jeden Tag eine große Anzahl von Anwendungen auf Ihrer Datenbank NONAME ausgeführt werden und dass in letzter Zeit einige Leistungsprobleme aufgetreten sind. Im Hinblick auf einige dieser Probleme beschließen Sie, dass Sie in der Lage sein müssen, die Anzahl großer Abfragen (d. h. aller Abfragen mit einem geschätzten Aufwand von mehr als 9 999 Timerons oder einer geschätzten Kardinalität von mehr als 9 999 Zeilen), die gleichzeitig in der Datenbank ausgeführt werden können, zu steuern. Sie wollen auch informiert werden, wenn die Anzahl der CALL-Anweisungen, die eine beliebige Routine im Schema MYSCHEMA aufrufen, 5 übersteigt.

Zur Steuerung der Anzahl großer Abfragen und CALL-Anweisungen, die in der Datenbank ausgeführt werden können, gehen Sie wie folgt vor:

1. Erstellen Sie das Arbeitsklassenset MYWORKCLASSSET, das drei Arbeitsklassen enthält: eine für Abfragen mit einem hohen geschätzten Aufwand, eine für Abfragen mit einer hohen geschätzten Kardinalität und eine für CALL-Anweisungen, die Prozeduren im Schema MYSCHEMA aufrufen. Zum Beispiel:

```
CREATE WORK CLASS SET MYWORKCLASSSET (WORK CLASS LARGEESTIMATEDCOST
WORK TYPE DML FOR TIMERONCOST FROM 10000 TO UNBOUNDED, WORK CLASS
LARGECARDINALITY WORK TYPE DML FOR CARDINALITY FROM 10000 TO UNBOUNDED,
WORK CLASS CALLSTATEMENTS WORK TYPE CALL ROUTINES IN SCHEMA MYSCHEMA)
```

2. Erstellen Sie ein Arbeitsaktionsset DATABASEACTIONS, das drei Arbeitsaktionen enthält, die auf die Arbeitsklassen im Arbeitsklassenset MYWORKCLASSSET auf Datenbankebene angewendet werden sollen.

```
CREATE WORK ACTION SET DATABASEACTIONS FOR DATABASE USING WORK CLASS SET
LARGEQUERIES (WORK ACTION ONECONCURRENTQUERY ON WORK CLASS LARGEESTIMATEDCOST
WHEN CONCURRENTDBCOORDACTIVITIES > 1 AND QUEUEDACTIVITIES > 1 STOP EXECUTION,
WORK ACTION TWOCONCURRENTQUERIES ON WORK CLASS LARGECARDINALITY
WHEN CONCURRENTDBCOORDACTIVITIES > 2 AND QUEUEDACTIVITIES > 3 STOP EXECUTION,
WORK ACTION FIVECALLS ON WORK CLASS CALLSTATEMENTS
WHEN CONCURRENTDBCOORDACTIVITIES > 5 COLLECT ACTIVITY DATA CONTINUE)
```

Darüber hinaus werden täglich mehrere große Verwaltungsanwendungen auf der Datenbank ausgeführt. Diese Anwendungen sollen in einem einzelnen Ressourcenpool ausgeführt werden. Dazu erstellen Sie eine Servicesuperklasse namens ADMINAPPS für diese Anwendungen. Für jede Anwendung erstellen Sie eine Workload, um sie der Servicesuperklasse ADMINAPPS zuzuordnen.

Da es wichtig ist, dass die Abfragen (SELECT-Anweisungen) schnell ausgeführt werden, beschließen Sie, für diese Abfragen eine Serviceunterklasse namens SELECTS in der Servicesuperklasse ADMINAPPS zu erstellen.

Gehen Sie wie folgt vor, um die SELECT- und XQuery-Anweisungen der Serviceunterklasse SELECTS zuzuordnen:

1. Erstellen Sie ein Arbeitsklassenset SELECTDML, das eine Arbeitsklasse für alle SELECT-Anweisungen enthält, die die Datenbank nicht aktualisieren:

```
CREATE WORK CLASS SET SELECTDML (WORK CLASS SELECTCLASS WORK TYPE READ)
```

- Erstellen Sie ein Arbeitsaktionsset ADMINAPPSACTIONS. Dieses Arbeitsaktionsset enthält eine Arbeitsaktion, die auf die Arbeitsklasse im Arbeitsklassenset SELECTDML auf der Ebene der Servicesuperklasse angewendet werden soll.

```
CREATE WORK ACTION SET ADMINAPPSACTIONS FOR SERVICE CLASS ADMINAPPS USING WORK
CLASS SET SELECTDML
(WORK ACTION MAPSELECTS ON WORK CLASS SELECTCLASS MAP ACTIVITY TO SELECTS)
```

Beispiel: Verwenden einer Arbeitsklasse mit dem Schlüsselwort ALL

Dieses Beispiel zeigt, wie Sie mit einer als ALL definierten Arbeitsklasse arbeiten können, die potenziell alle erkannten Aktivitäten in der Datenbank abdeckt.

Wenn eine Arbeitsklasse mit dem Typ ALL in Verbindung mit einer Zuordnungsarbeitsaktion verwendet wird, werden alle erkannten Datenbankaktivitäten der in der Arbeitsaktion angegebenen Serviceunterklasse zugeordnet. Wenn eine Arbeitsklasse mit dem Typ ALL in Verbindung mit einer Schwellenwertarbeitsaktion verwendet wird, bestimmt der Schwellenwerttyp, auf welche Datenbankaktivitäten der Schwellenwert angewendet wird. Betrachten wir das folgende Beispiel.

Nehmen wir an, dass Sie ein Arbeitsklassenset namens Example mit den folgenden Arbeitsklassen erstellen. Die Arbeitsklassen werden in der folgenden Reihenfolge ausgewertet:

- SMALLDML, für alle SQL-Anweisungen vom Typ DML mit einem geschätzten Aufwand von weniger als 1000 Timerons
- LOADUTIL, für das Dienstprogramm LOAD
- DDLACTIVITY, für die gesamte DDL-Aktivität
- ALLACTIVITY, für die gesamte Datenbankaktivität

ALLACTIVITY wird als letzte Arbeitsklasse ausgewertet. Sie umfasst Datenbankaktivitäten, die keiner der ersten drei Arbeitsklassen entsprechen.

Der DDL-Code zum Erstellen dieses Arbeitsklassensets lautet:

```
CREATE WORK CLASS SET EXAMPLE
(WORK CLASS SMALLDML WORK TYPE DML FOR TIMERONCOST FROM 0 TO 999,
WORK CLASS LOADUTIL WORK TYPE LOAD, WORK CLASS DDLACTIVITY WORK TYPE DDL,
WORK CLASS ALLACTIVITY WORK TYPE ALL)
```

Nehmen wir an, dass Sie eine Servicesuperklasse namens EXAMPLESERVICECLASS haben und dass diese zwei Serviceunterklassen namens SMALLACTIVITY und OTHERACTIVITY enthält. Sie wollen das System so einrichten, dass alle kleinen Datenbankaktivitäten in der Serviceunterklasse SMALLACTIVITY und alle anderen erkannten Datenbankaktivitäten mit Ausnahme des Dienstprogramms LOAD in der Serviceunterklasse OTHERACTIVITY ausgeführt werden. Sie wollen das Dienstprogramm LOAD keiner anderen Serviceunterklasse neu zuordnen. Dieses soll statt dessen in der Standardserviceunterklasse ausgeführt werden.

Dazu richten Sie das Arbeitsaktionsset SERVICECLASSACTIONS für die Servicesuperklasse EXAMPLESERVICECLASS ein. Das Arbeitsaktionsset SERVICECLASSACTIONS enthält die folgenden Arbeitsaktionen.

Tabelle 37. Arbeitsaktionsset SERVICECLASSACTIONS

Arbeitsaktion	Angewendet auf Arbeitsklasse	Aktion
MAPDML	SMALLDML	Wird der Serviceunterklasse SMALLACTIVITY zugeordnet
MAPOTHER	ALLACTIVITY	Wird der Serviceunterklasse OTHERACTIVITY zugeordnet
COUNTLOAD	LOADUTIL	Zählt die Anzahl der LOAD-Aktivitäten

Der DDL-Code zum Erstellen dieses Arbeitsaktionssets lautet:

```
CREATE WORK ACTION SET SERVICECLASSACTIONS FOR SERVICE CLASS EXAMPLESERVICECLASS
USING WORK CLASS SET EXAMPLE
(WORK ACTION MAPDML ON WORK CLASS SMALLDML MAP ACTIVITY TO SMALLACTIVITY,
WORK ACTION MAPOTHER ON WORK CLASS ALLACTIVITY MAP ACTIVITY TO OTHERACTIVITY,
WORK ACTION COUNTLOAD ON WORK CLASS LOADUTIL COUNT ACTIVITY)
```

Mit dieser Konfiguration werden alle kleinen DML-Anweisungen unter der Serviceunterklasse SMALLACTIVITY ausgeführt. Die Arbeitsaktion COUNTLOAD wird auf die Arbeitsklasse LOADUTIL angewendet, die unter der Standardserviceunterklasse ausgeführt wird. Alle anderen erkannten Datenbankaktivitäten werden unter der Serviceunterklasse OTHERACTIVITY ausgeführt. Wäre keine Arbeitsaktion auf die Arbeitsklasse LOADUTIL angewendet worden, würden LOAD-Aktivitäten unter die Arbeitsklasse ALLACTIVITY fallen, und die Arbeitsaktion MAPOTHER würde auf sie angewendet. (Wenn auf eine Arbeitsklasse keine Arbeitsaktion angewendet wird, werden keine Aktivitäten unter dieser Arbeitsklasse klassifiziert.)

Anmerkung: Befände sich die Arbeitsklasse ALLACTIVITY an erster Stelle in der Auswertungsreihenfolge, würden alle erkannten Aktivitäten der Serviceunterklasse OTHERACTIVITY zugeordnet.

Nehmen wir jetzt an, dass Sie ein Arbeitsaktionsset für die Datenbank definieren und Schwellenwerte anwenden wollen, die steuern, welche Aktivitäten gleichzeitig auf dem System ausgeführt werden dürfen. Sie könnten ein Arbeitsaktionsset namens DATABASEACTIONS mit den folgenden Arbeitsaktionen erstellen. Der DML-Code zum Erstellen dieses Arbeitsaktionssets lautet:

```
CREATE WORK ACTION SET DATABASEACTIONS FOR DATABASE USING WORK CLASS SET EXAMPLE
(WORK ACTION CONCURRENTSMALLDML ON WORK CLASS SMALLDML
WHEN CONCURRENTDBCOORDACTIVITIES > 1000 AND QUEUEDACTIVITIES > 10000
COLLECT ACTIVITY DATA STOP EXECUTION,
WORK ACTION CONCURRENTLOAD ON WORK
CLASS LOADUTIL WHEN CONCURRENTDBCOORDACTIVITIES > 2 AND QUEUEDACTIVITIES > 10 COLLECT
ACTIVITY DATA STOP EXECUTION,
WORK ACTION CONCURRENTOTHER ON WORK CLASS ALLACTIVITY
WHEN CONCURRENTDBCOORDACTIVITIES > 100 AND QUEUEDACTIVITIES > 100 COLLECT
ACTIVITY DATA STOP EXECUTION,
WORK ACTION MAXCOSTALLOWED ON WORK CLASS ALLACTIVITY
WHEN ESTIMATEDSQLCOST > 1000000 COLLECT ACTIVITY DATA STOP EXECUTION)
```

Tabelle 38. Arbeitsaktionsset DATABASEACTIONS

Arbeitsaktion	Angewendet auf Arbeitsklasse	Schwellenwerttyp und -wert	Aktion
CONCURRENTSMALLDML	SMALLDML	Gemeinsamer Zugriff: bis zu 1000 Anweisungen; Warteschlange: bis zu 10 000 Anweisungen	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
CONCURRENTLOAD	LOADUTIL	Gemeinsamer Zugriff: bis zu 2 Vorkommen; Warteschlange: bis zu 10 Vorkommen	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
CONCURRENTOTHER	ALLACTIVITY	Gemeinsamer Zugriff: bis zu 100 Aktivitäten; Warteschlange: bis zu 100 Aktivitäten	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
MAXCOSTALLOWED	ALLACTIVITY	Geschätzter SQL-Aufwand: bis zu 1 000 000 Timerons	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten

Wenn diese Arbeitsaktionen angewendet werden, können bis zu 1000 kleine SQL-Anweisungen vom Typ DML (auf Grund der Arbeitsklasse SMALLDML) gleichzeitig ausgeführt werden, und bis zu 10 000 dieser Anweisungen können in die Warteschlange gestellt werden. Nur zwei Vorkommen des Dienstprogramms LOAD können gleichzeitig ausgeführt werden, und bis zu 10 Vorkommen können in die Warteschlange gestellt werden. Nur 100 Aktivitäten, die nicht den Typ LOAD haben und keine kleinen DML-Anweisungen sind, dürfen gleichzeitig ausgeführt werden, und nur 100 dieser Aktivitäten können in die Warteschlange gestellt werden. In allen Situationen kann bei einem Verstoß gegen einen Warteschlangenschwellenwert die Datenbankaktivität nicht ausgeführt werden, und es wird eine Fehlermeldung zurückgegeben.

Darüber hinaus wird die Arbeitsaktion MAXCOSTALLOWED auf die Arbeitsklasse ALLACTIVITY angewendet. Dies bedeutet, dass eine Datenbankaktivität mit einem geschätzten Aufwand (d. h. DML- und XQueries-Anweisungen) von mehr als 1 000 000 Timerons nicht ausgeführt werden darf. Obwohl die Arbeitsaktion MAXCOSTALLOWED auf die Arbeitsklasse ALLACTIVITY angewendet wird, wirkt sich diese Arbeitsaktion nur auf Datenbankaktivitäten mit einem geschätzten Aufwand von mehr als 1 000 000 Timerons aus. Diese Arbeitsaktion betrifft keine Aktivitäten, die keinen geschätzten Aufwand haben, z. B. DDL.

Beispiel: Verwenden von Arbeitsaktionssets und Datenbankschwellenwerten

Dieses Beispiel zeigt unterschiedliche Strategien zur Verwendung von Arbeitsaktionssets und Schwellenwerten zur Steuerung der von DB2-Aktivitäten verbrauchten Ressourcen. Vor dem Erstellen von Workload-Management-Objekten müssen Sie verstanden haben, wie diese eingesetzt werden.

Nehmen wir an, dass Sie ein Arbeitsklassenset namens ALLSQL haben und dass dieses die folgenden Arbeitsklassen in der aufgeführten Reihenfolge enthält:

1. SMALLDML, für alle SQL-Anweisungen vom Typ DML mit einem geschätzten Aufwand von weniger als 1 000 Timerons
2. MEDDML, für alle SQL-Anweisungen vom Typ DML mit einem geschätzten Aufwand zwischen 1 000 und 20 000 Timerons
3. LARGEDML, für alle SQL-Anweisungen vom Typ DML mit einem geschätzten Aufwand von mehr als 20 000 Timerons
4. ALLDDL, für alle SQL-Anweisungen vom Typ DML
5. ALLACTIVITY, für die gesamte Datenbankaktivität

Auf diese Arbeitsklassen wurden bereits Arbeitsaktionen angewendet, z. B. COUNT ACTIVITY, COLLECT und Schwellenwerte (die keine ACTIVITYTOTALTIME-Schwellenwerte sind).

Nehmen wir an, dass Sie die Ausführung von großen DML-Aktivitäten auf maximal 5 Stunden begrenzen wollen. Die Ausführung aller anderen SQL-Aktivitäten darf nicht länger als 30 Minuten dauern. Die folgenden beiden Beispiele zeigen Methoden, mit denen dieses Ziel erreicht werden kann.

Methode 1

Eine Methode besteht darin, für jede Arbeitsklasse eine Arbeitsaktion mit dem Schwellenwert ACTIVITYTOTALTIME zu erstellen:

Tabelle 39. Schwellenwert ACTIVITYTOTALTIME für jede Arbeitsklasse

Arbeitsaktion	Angewendet auf Arbeitsklasse	Schwellenwerttyp und -wert	Aktionen
SMALLDMLTIMEALLOWED	SMALLDML	ACTIVITYTOTALTIME < 31 MINUTES	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
MEDDMLTIMEALLOWED	MEDDML	ACTIVITYTOTALTIME < 31 MINUTES	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
LARGDMLTIMEALLOWED	LARGEDML	ACTIVITYTOTALTIME < 5 HOURS	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
ALLDDLTIMEALLOWED	ALLDDL	ACTIVITYTOTALTIME < 31 MINUTES	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten
ALLACTIVITYTIMEALLOWED	ALLACTIVITY	ACTIVITYTOTALTIME < 31 MINUTES	<ul style="list-style-type: none"> • Stoppen der Ausführung • Erfassen von Aktivitätsdaten

Methode 2

Eine weitere Methode wäre, nur eine Arbeitsklasse (LARGEDML) zu verwenden und dann ein Arbeitsaktionsset für die Datenbank mit einer Arbeitsaktion (LARGEDMLTIMEALLOWED) zu erstellen, die auf die Arbeitsklasse angewendet wird.

Tabelle 40. Arbeitsaktion LARGEDMLTIMEALLOWED angewendet auf die Arbeitsklasse LARGEDML

Arbeitsaktion	Angewendet auf Arbeitsklasse	Schwellenwerttyp und -wert	Aktion
LARGEDMLTIMEALLOWED	LARGEDML	ACTIVITYTOTALTIME < 5 HOURS	<ul style="list-style-type: none">• Stoppen der Ausführung• Erfassen von Aktivitätsdaten

Sie würden dann einen Schwellenwert ACTIVITYTOTALTIME von weniger als 31 Minuten auf die Datenbank anwenden. Mit dieser Methode wird der 5-Stunden-Schwellenwert nur auf Aktivitäten angewendet, die der Arbeitsklasse LARGEDML entsprechen. Andere Aktivitäten werden nicht klassifiziert. Für sie gilt der Datenbankschwellenwert ACTIVITYTOTALTIME von weniger als 31 Minuten.

Beispiel: Festlegen des Typs der auszuführenden Arbeit mit Hilfe von Arbeitsaktionssets

Durch die Verwendung von Arbeitsklassensets, Arbeitsklassen, Arbeitsaktionssets, Arbeitsaktionen und einigen Workload-Management-Überwachungsfunktionen können Sie die unterschiedlichen Typen von Arbeit auf Ihrem System sowie die Verteilung der Arbeit ermitteln.

Zum Durchführen dieser Aufgabe müssen Sie zuerst ein Arbeitsklassenset erstellen, das Arbeitsklassen für die unterschiedlichen Typen von Arbeit enthält, an denen Sie interessiert sind. Wenn Sie z. B. wissen wollen, wieviele READ-Aktivitäten, WRITE-Aktivitäten, DDL-Aktivitäten und LOAD-Aktivitäten auf Ihrem System ausgeführt werden, erstellen Sie ein Arbeitsklassenset, ACTIVITYTYPES (siehe folgendes Beispiel):

```
CREATE WORK CLASS SET ACTIVITYTYPES (WORK CLASS READWC WORK TYPE READ, WORK CLASS WRITEWC WORK TYPE WRITE, WORK CLASS DDLWC WORK TYPE DDL, WORK CLASS LOADWC WORK TYPE LOAD)
```

Als Nächstes erstellen Sie ein Arbeitsaktionsset auf Datenbankebene, COUNTACTIONS, das Sie auf das Arbeitsklassenset ACTIVITYTYPES anwenden. Das Arbeitsaktionsset enthält eine Arbeitsaktion COUNT ACTIVITY für jede Arbeitsklasse im Arbeitsklassenset ACTIVITYTYPES (siehe folgendes Beispiel):

```
CREATE WORK ACTION SET COUNTACTIONS FOR DATABASE USING WORK CLASS SET ACTIVITYTYPES (WORK ACTION COUNTREAD ON WORK CLASS READWC COUNT ACTIVITY, WORK ACTION COUNTWRITE ON WORK CLASS WRITEWC COUNT ACTIVITY, WORK ACTION COUNTDDL ON WORK CLASS DDLWC COUNT ACTIVITY, WORK ACTION COUNTLOAD ON WORK CLASS LOADWC COUNT ACTIVITY)
```

Nachdem ausreichend Zeit vergangen ist, können Sie die Anzahl der einzelnen ausgeführten Aktivitätstypen mit der Tabellenfunktion WLM_GET_WORK_ACTION_SET_STATS ermitteln:

```
SELECT SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,  
LAST_RESET,  
SUBSTR(WORK_CLASS_NAME,1,15) AS WORK_CLASS_NAME,
```

```

SUBSTR(CHAR(ACT_TOTAL),1,14) AS TOTAL_ACTS
FROM TABLE(WLM_GET_WORK_ACTION_SET_STATS(CAST(NULL AS VARCHAR(128)), -2))
AS WASSTATS WHERE WORK_ACTION_SET_NAME = 'COUNTACTIONS'
ORDER BY WORK_CLASS_NAME, PART

```

Beispiel: Verwenden von Workload-Management-Tabellenfunktionen zum Überwachen des aktuellen Systemverhaltens auf verschiedenen Ebenen

Das Workload-Management-Tool bietet eine Vielzahl von Tabellenfunktionen, mit denen Sie Daten zu Ihrer Workload-Management-Konfiguration erhalten können.

Bei der Installation von DB2 Version 9.5 wird ein Satz von Standardworkloads und -serviceklassen erstellt. Bevor Sie sich entscheiden, wie Sie Ihre eigene Workload-Management-Lösung implementieren wollen, können Sie mit Hilfe der Tabellenfunktionen die im System ausgeführte Arbeit im Hinblick auf die Standardworkloadvorkommen, Serviceklassen und Aktivitäten beobachten.

Sie können beginnen, indem Sie die Liste der Workloadvorkommen in einer Serviceklasse abrufen. Verwenden Sie dazu die Tabellenfunktion `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES`. Im folgenden Beispiel wird eine leere Zeichenfolge für *servicesuperklassenname* und *serviceunterklassenname* und `-2` (ein Platzhalterzeichen) für *dbpartitionsnummer* übergeben:

```

SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(COORD_PARTITION_NUM),1,4) AS COORDPART,
       SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHNDL,
       SUBSTR(CHAR(WORKLOAD_NAME),1,22) AS WORKLOAD_NAME,
       SUBSTR(CHAR(WORKLOAD_OCCURRENCE_ID),1,6) AS WLO_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES('', '', -2)) AS SCINFO
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART, APPHNDL, WORKLOAD_NAME, WLO_ID

```

Nehmen wir an, dass das System vier Datenbankpartitionen hat und dass zwei Anwendungen Aktivitäten in der Datenbank ausführen, wenn Sie die Abfrage absetzen. Die Ergebnisse ähneln der folgenden Ausgabe:

SUPERCLASS_NAME	SUBCLASS_NAME	PART	COORDPART	APPHNDL	WORKLOAD_NAME	WLO_ID
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	0	0	1	SYSDEFAULTUSERWORKLOAD	1
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	0	0	2	SYSDEFAULTUSERWORKLOAD	2

Die Ergebnisse zeigen, dass beide Workloadvorkommen der Workload `SYSDEFAULTUSERWORKLOAD` zugeordnet wurden. Außerdem zeigen die Ergebnisse, dass beide Workloadvorkommen der Serviceunterklasse `SYSDEFAULTSUBCLASS` in der Servicesuperklasse `SYSDEFAULTUSERCLASS` zugeordnet wurden und dass beide Workloadvorkommen aus derselben Koordinatorpartition (Partition 0) stammen.

Als Nächstes können Sie die Tabellenfunktion `WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES` erneut verwenden, um die Verbindungsattribute der beiden Workloadvorkommen zu ermitteln:

```

SELECT SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHNDL,
       SUBSTR(CHAR(WORKLOAD_NAME),1,22) AS WORKLOAD_NAME,
       SUBSTR(CHAR(WORKLOAD_OCCURRENCE_ID),1,6) AS WLO_ID,
       SUBSTR(CHAR(SYSTEM_AUTH_ID),1,9) AS SYSAUTHID,
       SUBSTR(CHAR(APPLICATION_NAME),1,15) AS APPLNAME
FROM TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES('', '', 0)) AS SCINFO
ORDER BY APPHNDL, WORKLOAD_NAME, WLO_ID

```

APPHNDL	WORKLOAD_NAME	WLO_ID	SYSAUTHID	APPLNAME
1	SYSDEFAULTUSERWORKLOAD 1	1	LYNN	accountspay
2	SYSDEFAULTUSERWORKLOAD 2	2	KATE	businessobjects

Dann können Sie mit der Tabellenfunktion WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES die aktuellen Aktivitäten eines der beiden Workloadvorkommen anzeigen:

```
SELECT SUBSTR(CHAR(COORD_PARTITION_NUM),1,5) AS COORD,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(UOW_ID),1,5) AS UOWID,
       SUBSTR(CHAR(ACTIVITY_ID),1,5) AS ACTID,
       SUBSTR(CHAR(PARENT_UOW_ID),1,8) AS PARUOWID,
       SUBSTR(CHAR(PARENT_ACTIVITY_ID),1,8) AS PARACTID,
       SUBSTR(ACTIVITY_TYPE,1,9) AS ACTTYPE,
       SUBSTR(CHAR(NESTING_LEVEL),1,7) AS NESTING
FROM TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(1, -2)) AS WLOACTS
ORDER BY PART, UOWID, ACTID
```

COORD	PART	UOWID	ACTID	PARUOWID	PARACTID	ACTTYPE	NESTING
0	0	1	3	-	-	CALL	0
0	0	1	5	1	3	READ_DML	1
0	1	1	5	-	-	READ_DML	1
0	2	1	5	-	-	READ_DML	1
0	3	1	5	-	-	READ_DML	1

Die Abfrageergebnisse zeigen, dass das Workloadvorkommen 1 zwei Aktivitäten ausführt. Eine Aktivität ist eine gespeicherte Prozedur (angegeben durch den Aktivitätstyp CALL), und die andere Aktivität ist eine DML-Aktivität, die einen Lesevorgang ausführt (z. B. eine Anweisung SELECT). Die DML-Aktivität ist im Aufruf der gespeicherten Prozedur verschachtelt. Sie können erkennen, dass die DML-Aktivität verschachtelt ist, da die ID der übergeordneten UOW (Unit of Work - Arbeitseinheit) und die ID der übergeordneten Aktivität der DML-Aktivität der UOW-ID und der Aktivitäts-ID der CALL-Aktivität entsprechen. Sie können auch erkennen, dass die DML-Aktivität auf den Datenbankpartitionen 0, 1, 2 und 3 ausgeführt wird. Die Informationen zur übergeordneten ID sind nur auf der Koordinatorpartition verfügbar.

Sie können weitere Informationen zu einer gerade ausgeführten Aktivität mit der Tabellenfunktion WLM_GET_ACTIVITY_DETAILS erhalten. Diese Tabellenfunktion gibt Aktivitätsinformationen wie Name/Wert-Paare für jede Datenbankpartition zurück. Im folgenden Beispiel zeigt die Tabellenfunktion nur eine aus 11 Elementen bestehende Untergruppe von Name/Wert-Paaren für jede Datenbankpartition an. Eine vollständige Liste der Name/Wert-Paare finden Sie in Tabellenfunktion 'WLM_GET_ACTIVITY_DETAILS'.

```
SELECT SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(NAME, 1, 20) AS NAME,
       SUBSTR(VALUE, 1, 30) AS VALUE
FROM TABLE(WLM_GET_ACTIVITY_DETAILS(1, 1, 5, -2)) AS ACTDETAIL
WHERE NAME IN ('APPLICATION_HANDLE',
              'COORD_PARTITION_NUM',
              'LOCAL_START_TIME',
              'UOW_ID',
              'ACTIVITY_ID',
              'PARENT_UOW_ID',
              'PARENT_ACTIVITY_ID',
              'ACTIVITY_TYPE',
              'NESTING_LEVEL',
              'INVOCATION_ID',
              'ROUTINE_ID')
```

ORDER BY PART

PART	NAME	VALUE
0	APPLICATION_HANDLE	1
0	COORD_PARTITION_NUM	0
0	UOW_ID	1
0	ACTIVITY_ID	5
0	PARENT_UOW_ID	1
0	PARENT_ACTIVITY_ID	3
0	ACTIVITY_TYPE	READ_DML
0	NESTING_LEVEL	0
0	INVOCATION_ID	1
0	ROUTINE_ID	0
0	LOCAL_START_TIME	2005-11-25-18.52.49.343000
1	APPLICATION_HANDLE	1
1	COORD_PARTITION_NUM	0
1	UOW_ID	1
1	ACTIVITY_ID	5
1	PARENT_UOW_ID	-
1	PARENT_ACTIVITY_ID	-
1	ACTIVITY_TYPE	READ_DML
1	NESTING_LEVEL	0
1	INVOCATION_ID	1
1	ROUTINE_ID	0
1	LOCAL_START_TIME	2005-11-25-18.52.49.598000

Die drei bisher aufgeführten Tabellenfunktionen bieten eine abstrakte Beschreibung der auf dem System ausgeführten Arbeit. Die von diesen Tabellenfunktionen bereitgestellten Informationen zum Status der Arbeit ist auf einen Aktivitätsstatus wie EXECUTING beschränkt. Wenn Sie präziser untersuchen wollen, was genau zu einem bestimmten Zeitpunkt in einer Serviceklasse geschieht, können Sie die Tabellenfunktion WLM_GET_SERVICE_CLASS_AGENTS ausführen.

Im folgenden Beispiel wird WLM_GET_SERVICE_CLASS_AGENTS durch Übergabe von 1 für *anwendungskennung* und -2 (ein Platzhalterzeichen) für *dbpartitionsnummer* aufgerufen:

```
SELECT SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHANDLE,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(AGENT_TID),1,9) AS AGENT_TID,
       SUBSTR(CHAR(AGENT_TYPE),1,11) AS AGENTTYPE,
       SUBSTR(CHAR(AGENT_STATE),1,10) AS AGENTSTATE,
       SUBSTR(CHAR(REQUEST_TYPE),1,14) AS REQTYPE,
       SUBSTR(CHAR(UOW_ID),1,6) AS UOW_ID,
       SUBSTR(CHAR(ACTIVITY_ID),1,6) AS ACT_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS('',' ', 1, -2)) AS SCDETAILS
ORDER BY APPHANDLE, PART, AGENT_TID
```

APPHANDLE	PART	AGENT_TID	AGENTTYPE	AGENTSTATE	REQTYPE	UOW_ID	ACT_ID
1	0	3	COORDINATOR	ACTIVE	FETCH	1	5
1	0	4	PDBSUBAGENT	ACTIVE	SUBSECTION:1	1	5
1	1	2	PDBSUBAGENT	ACTIVE	SUBSECTION:2	1	5

Die Ergebnisse zeigen einen Koordinatoragenten und einen Subagenten auf der Datenbankpartition 0 sowie einen Subagenten auf der Datenbankpartition 1, der für eine Aktivität mit einer UOW-ID von 1 und einer Aktivitäts-ID von 5 aktiv ist. Die Informationen zum Koordinatoragenten geben an, dass es sich bei der Anforderung um eine Abrufanforderung handelt.

Beispiel: Abrufen von Zeitpunktstatistikdaten aus Serviceklassen

Jede Aktivität wird vor der Ausführung einer Serviceklasse zugeordnet. Sie können das System mit den Tabellenfunktionen für die Serviceklassenstatistik und durch Abfrage aller Serviceklassen auf allen Datenbankpartitionen überwachen, um Zeitpunktstatistikdaten zu erhalten.

Mit der folgenden Anweisung können Sie Statistikdaten für Serviceklassen wie die durchschnittliche Aktivitätslaufzeit erhalten. Wenn eine leere Zeichenfolge für ein Argument für die Tabellenfunktion `WLM_GET_SERVICE_SUBCLASS_STATS` übergeben wird, wird das Ergebnis nicht durch dieses Argument eingeschränkt. Der Wert des letzten Arguments, *dbpartitionsnummer*, ist -2 (ein Platzhalterzeichen), d. h. dass Daten von allen Datenbankpartitionen zurückgegeben werden.

Anmerkung: Informationen zur Laufzeit können nur für die Serviceklassen zurückgegeben werden, für die `COLLECT AGGREGATE ACTIVITY DATA` definiert ist.

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CAST(COORD_ACT_LIFETIME_AVG / 1000 AS DECIMAL(9,3)) AS AVGLIFETIME,
       CAST(COORD_ACT_LIFETIME_STDDEV / 1000 AS DECIMAL(9,3)) AS STDDEVLIFETIME,
       SUBSTR(CAST(LAST_RESET AS VARCHAR(30)),1,16) AS LAST_RESET
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS('', '', -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART
```

SUPERCLASS_NAME	SUBCLASS_NAME	PART	AVGLIFETIME	STDDEVLIFETIME	LAST_RESET
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 0	0	691.242	34.322	2006-07-24-11.44
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 1	1	644.740	22.124	2006-07-24-11.44
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 2	2	612.431	43.347	2006-07-24-11.44
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 3	3	593.451	28.329	2006-07-24-11.44

Sie können auch die Tabellenfunktion `WLM_GET_SERVICE_SUBCLASS_STATS` verwenden, um die obere Grenze für den gemeinsamen Zugriff von Koordinatoraktivitäten zu erhalten, die in der Serviceklasse auf jeder Datenbankpartition ausgeführt werden:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CONCURRENT_ACT_TOP AS ACTHIGHWATERMARK
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS('', '', -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART
```

SUPERCLASS_NAME	SUBCLASS_NAME	PART	ACTHIGHWATERMARK
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 0	0	10
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 1	1	0
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 2	2	0
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 3	3	0

Durch Überprüfung der durchschnittlichen Laufzeit und der Anzahl abgeschlossener Aktivitäten können Sie die Ausgabe der Tabellenfunktion `WLM_GET_SERVICE_SUBCLASS_STATS` verwenden, um eine zusammengefasste Ansicht der Workload auf jeder Datenbankpartition in der Datenbank zu erhalten. Signifikante Abweichungen bei den von einer Tabellenfunktion zurückgegebenen oberen Grenzen und Durchschnittswerten können auf eine Änderung in der Workload auf dem System hindeuten.

Beispiel: Zusammenfassen von Daten mit Hilfe von Workload-Management-Tabellenfunktionen

Sie können verschiedene Zusammenfassungen von Tabellendaten in einer Workload-Management-Konfiguration durchführen, um das System zu überwachen und mögliche Probleme zu erkennen.

Im Folgenden sehen Sie Beispiele der Datenerfassung und -zusammenfassung, die Sie zur Feststellung von Problemen durchführen können.

Erkennen von Erhöhungen der durchschnittlichen Laufzeiten, da Abfragen zu lange in der Warteschlange verbleiben

Sie können eine Situation erkennen, in der sich die durchschnittlichen Abfragelaufzeiten erhöhen, da die Abfragen zu lange in der Warteschlange verbleiben. Dazu zeigen Sie die Durchschnittswartezeit für Koordinatoraktivitäten für jede Serviceklasse über das gesamte System an.

Im Folgenden sehen Sie ein Beispiel, das den Prozentsatz der Zeit, während der eine durchschnittliche Abfrage in der Warteschlange wartet, für Koordinatoraktivitäten für jede Serviceklasse zusammengefasst über alle Datenbankpartitionen anzeigt:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       CASE WHEN (SUM(COORD_ACT_COMPLETED_TOTAL) = 0) THEN
         0
       ELSE
         SUM(COORD_ACT_QUEUE_TIME_AVG * COORD_ACT_COMPLETED_TOTAL) /
         SUM(COORD_ACT_COMPLETED_TOTAL)
       END AS AVG_QUEUE_TIME,
       CASE WHEN (SUM(COORD_ACT_COMPLETED_TOTAL) = 0) THEN
         0
       ELSE
         SUM(COORD_ACT_LIFETIME_AVG * COORD_ACT_COMPLETED_TOTAL) /
         SUM(COORD_ACT_COMPLETED_TOTAL)
       END AS AVG_LIFE_TIME,
       CASE WHEN (SUM(COORD_ACT_COMPLETED_TOTAL) = 0) THEN
         0
       ELSE CASE WHEN
         (CAST(SUM(COORD_ACT_LIFETIME_AVG * COORD_ACT_COMPLETED_TOTAL) /
          SUM(COORD_ACT_COMPLETED_TOTAL) AS INTEGER) = 0) THEN
           0
         ELSE
           100 * (SUM(COORD_ACT_QUEUE_TIME_AVG * COORD_ACT_COMPLETED_TOTAL) /
            SUM(COORD_ACT_COMPLETED_TOTAL)) /
            (SUM(COORD_ACT_LIFETIME_AVG * COORD_ACT_COMPLETED_TOTAL) /
            SUM(COORD_ACT_COMPLETED_TOTAL))
         END
       END AS PERCENT_TIME_QUEUED
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS(' ', ' ', -2)) AS STATS
GROUP BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME
ORDER BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME
```

SUPERCLASS_NAME	SUBCLASS_NAME	AVG_QUEUE_TIME	AVG_LIFE_TIME	PERCENT_TIME_QUEUED
SYSDEFAULTMAINTENAN	SYSDEFAULTSUBCLASS	+0.0000000000000000E+000	+0.0000000000000000E+000	+0.0000000000000000E+000
SYSDEFAULTSYSTEMCLA	SYSDEFAULTSUBCLASS	+0.0000000000000000E+000	+0.0000000000000000E+000	+0.0000000000000000E+000
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	+2.3286010000000000E+005	+8.2342142400000000E+005	+2.8280000000000000E-001

Die Ergebnisse zeigen, dass der Prozentsatz der Zeit, während der eine durchschnittliche Aktivität in der Warteschlange verbleibt, 28% beträgt. Wenn die bisherige Erfahrung mit dem System und der Workload darauf hindeutet, dass dies zu

hoch oder zu niedrig ist, kann sich eine Anpassung Ihrer Schwellenwerte auf diesen Prozentsatz auswirken.

Erkennen von plötzlichen Anstiegen der Anzahl der in einer Workload ausgeführten Abfragen

Nehmen wir an, Sie haben eine Workload namens WL1. Sie können eine Situation erkennen, in der eine große Anzahl von Abfragen in der Workload ausgeführt wird. Zeigen Sie dazu die Gesamtzahl der ausgeführten unverschachtelten Koordinatoraktivitäten für die Workload über das gesamte System an:

```
SELECT SUBSTR(WORKLOAD_NAME,1,22) AS WLNAME,  
COUNT(*) AS TOTAL_EXE_ACT  
FROM TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES(' ', ' ', -2)) AS APPS,  
TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(APPS.APPLICATION_HANDLE, -2)) AS APPACTS  
WHERE WORKLOAD_NAME = 'WL1' AND  
APPS.DBPARTITIONNUM = APPS.COORD_PARTITION_NUM AND  
ACTIVITY_STATE = 'EXECUTING' AND  
NESTING_LEVEL = 0  
GROUP BY WORKLOAD_NAME
```

WLNAME	TOTAL_EXE_ACT
WL1	5

Beispiel: Berechnen von Durchschnittswerten und einer Standardabweichung anhand von Histogrammen in einer Workload-Management-Konfiguration

Eine Verwendungsmöglichkeit von Histogrammen ist die Ermittlung der Standardabweichung für Aktivitätslaufzeiten. Das Beispiel in diesem Thema zeigt, wie Bins für die Berechnung dieser Statistikdaten verwendet werden.

Eine Berechnung der durchschnittlichen Laufzeit für jede Aktivität ist eine nützliche Information. Der Durchschnitt allein beschreibt jedoch nicht präzise, wie diese vom Benutzer erlebt wird. Wenn die Streuung in der Aktivitätslaufzeit groß ist, können die von Ihnen unterstützten Benutzer manchmal sehr schnelle Abfragen erleben (was erstrebenswert ist) und manchmal sehr langsame (was eventuell nicht akzeptabel ist). Wenn Sie ein Ziel für die Aktivitätslaufzeiten definieren, ist nicht nur die durchschnittliche Laufzeit der Aktivitäten wichtig, sondern auch die Standardabweichung der Aktivitätslaufzeit. Sie müssen die Streuung verstehen und kontrollieren, um sicherzustellen, dass Ihre Benutzer tatsächlich den beobachteten Durchschnittswert erleben.

In einer Workload-Management-Konfiguration werden Statistikdaten auf jeder Datenbankpartition erfasst. Das folgende Beispiel zeigt, wie Sie die durchschnittliche Aktivitätslaufzeit für eine einzige Datenbankpartition ermitteln können.

Nehmen wir an, Sie haben eine Umgebung mit einer einzigen Partition sowie ein Histogramm mit den folgenden Bins. In den echten Histogrammen sind mehr Bins enthalten. In diesem Beispiel ist die Anzahl der Bins jedoch zur Vereinfachung auf 8 beschränkt.

- Bin 1 - 0 bis 2 Sekunden
- Bin 2 - 2 bis 4 Sekunden
- Bin 3 - 4 bis 8 Sekunden
- Bin 4 - 8 bis 16 Sekunden

Bin 5 - 16 bis 32 Sekunden
 Bin 6 - 32 bis 64 Sekunden
 Bin 7 - 64 bis 128 Sekunden
 Bin 8 - 128 Sekunden bis Unendlich

Sie können einen Näherungswert des Durchschnitts berechnen, indem Sie annehmen, dass die durchschnittliche Antwortzeit für eine Abfrage, die in einen Bin mit dem Bereich x bis y fällt, $(x + y)/2$ ist. Sie können dann diese Zahl mit der Anzahl der Abfragen im Bin multiplizieren, die Summe über alle Bins bilden und dann die Summe durch die Gesamtzahl dividieren. Im vorstehenden Beispiel nehmen wir an, dass sich eine durchschnittliche Antwortzeit für jeden Bin wie folgt ergibt:

Bin 1 durchschnittliche Laufzeit = $(0+2)/2 = 1$
 Bin 2 durchschnittliche Laufzeit = $(2+4)/2 = 3$
 Bin 3 durchschnittliche Laufzeit = $(4+8)/2 = 6$
 Bin 4 durchschnittliche Laufzeit = $(8+16)/2 = 12$
 Bin 5 durchschnittliche Laufzeit = $(16+32)/2 = 24$
 Bin 6 durchschnittliche Laufzeit = $(32+64)/2 = 48$
 Bin 7 durchschnittliche Laufzeit = $(64+128)/2 = 96$

Nehmen wir an, dass das folgende Histogramm im Abrechnungszeitraum erfasst wurde:

Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8
Anzahl							
20	30	80	10	5	3	2	0

Zur Berechnung der durchschnittlichen Laufzeit muss Bin 8 leer sein. Bin 8 ist nur vorhanden, damit Sie erkennen können, wann Sie die obere Begrenzung Ihres Bereichs ändern müssen. Aus diesem Grund müssen Sie die Obergrenze für den Bereich angeben.

Sie können die durchschnittliche Laufzeit für die Datenbankpartition 1 wie folgt näherungsweise berechnen:

$$\begin{aligned} \text{durchschnittliche Laufzeit} &= (20 \times 1 + 30 \times 3 + 80 \times 6 + 10 \times 12 + 5 \times 24 + 3 \times 48 + 2 \times 96) / 150 \\ &= (20 + 90 + 480 + 120 + 120 + 144 + 192) / 150 \\ &= 1166 / 150 \\ &= 7,77 \text{ Sekunden} \end{aligned}$$

Sie können die Standardabweichung der Laufzeit wie folgt näherungsweise berechnen:

$$\text{Standardabweichung} = [(20 \times (1 - 7,77)^2 + 30 \times (3 - 7,77)^2 + \dots) / 150]^{1/2}$$

In Umgebungen mit partitionierten Datenbanken können Durchschnitte und Standardabweichungen berechnet werden, indem zuerst ein kombiniertes Histogramm über alle Datenbankpartitionen durch Addition der Anzahl aller Bins über die Datenbankpartitionen berechnet wird.

Nehmen wir z. B. an, dass die Datenbank zwei Partitionen hat, die Binsgrößen im Histogramm wie oben beschrieben sind und das Histogramm die folgenden Daten enthält:

Datenbankpartition	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8
Anzahl	Anzahl	Anzahl	Anzahl	Anzahl	Anzahl	Anzahl	Anzahl	Anzahl
1	20	30	80	10	5	3	2	0
2	1	5	20	20	4	0	0	0

Da die Binsgrößen über alle Datenbankpartitionen gleich sind, ist das Gesamthistogramm einfach zu berechnen:

Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8
Anzahl							
21	35	100	30	9	3	2	0

Aus dem kombinierten Histogramm können Sie den Gesamtdurchschnitt der Laufzeit und die Gesamtstandardabweichung auf ähnliche Weise berechnen wie bei einer Umgebung mit eine Partition:

$$\begin{aligned} \text{Durchschnittliche Laufzeit} &= (21 \times 1 + 35 \times 3 + 100 \times 6 + 30 \times 12 + 9 \times 24 + 3 \times 48 + 2 \times 96) / 200 \\ &= (21 + 105 + 600 + 360 + 216 + 144 + 192) / 200 \\ &= 1638 / 200 \\ &= 8,19 \text{ Sekunden} \end{aligned}$$

$$\text{Standardabweichung} = [(21 \times (1 - 8,19)^2 + 35 \times (3 - 8,19)^2 + \dots) / 200]^{1/2}$$

Beispiel: Analysieren einer serviceklassenbezogenen Leistungsminderung des Systems

Wenn Sie eine Leistungsminderung des Systems bemerken (z. B. dauert die Ausführung einiger Anwendungen länger als erwartet) und Sie nicht sicher sind, ob das Problem mit der Konfiguration der Serviceklassen zusammenhängt, können Sie mit Tabellenfunktionsdaten das Problem untersuchen und erforderlichenfalls beheben.

Zuerst erhalten Sie eine komplexe Übersicht, was in den Serviceklassen abläuft. Diese Übersicht sollte die durchschnittliche Aktivitätslaufzeit, die Anzahl der normal abgeschlossenen Aktivitäten und die obere Grenze für gleichzeitig ablaufende Koordinatoraktivitäten im System enthalten. Dazu können Sie eine allgemeine Abfrage, bei der nur ein einziger Wert zurückgegeben wird (Aggregation - Zusammenfassung), über Serviceklassen und Datenbankpartitionen mit den Daten aus der Tabellenfunktion WLM_GET_SERVICE_CLASS_STATS erstellen. Definieren Sie für das erste und das zweite Argument leere Zeichenfolgen, und stellen Sie das dritte Argument auf -2 (ein Platzhalterzeichen) ein, um anzugeben, dass Daten für alle Serviceklassen auf allen Datenbankpartitionen zusammengestellt werden sollen. Ihre Abfrage könnte der folgenden ähneln:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(SUM(COORD_ACT_COMPLETED_TOTAL)),1,13) AS ACTSCOMPLETED,
       SUBSTR(CHAR(SUM(COORD_ACT_ABORTED_TOTAL)),1,11) AS ACTSABORTED,
       SUBSTR(CHAR(MAX(CONCURRENT_ACT_TOP)),1,6) AS ACTSHW,
       CAST(CASE WHEN SUM(COORD_ACT_COMPLETED_TOTAL) = 0 THEN 0
              ELSE SUM(COORD_ACT_COMPLETED_TOTAL * COORD_ACT_LIFETIME_AVG)
              / SUM(COORD_ACT_COMPLETED_TOTAL) END / 1000 AS DECIMAL(9,3))
       AS ACTAVGLIFETIME
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS ('', '', -2)) AS SCSTATS
GROUP BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME
ORDER BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME
```

SUPERCLASS_NAME	SUBCLASS_NAME	ACTSCOMPLETED	ACTSABORTED	ACTSHW	ACTAVGLIFETIME
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 8	0	1	3.750	
BI_APPS	SYSDEFAULTSUBCLASS 4	0	1	14.230	
BATCH	SYSDEFAULTSUBCLASS 1	0	1	25.600	

Nehmen wir an, dass bei früheren Gelegenheiten die Abfrage die folgenden Ergebnisse zurückgegeben hat:

SUPERCLASS_NAME	SUBCLASS_NAME	ACTSCOMPLETED	ACTSABORTED	ACTSHW	ACTAVGLIFETIME
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS 8	0	1	3.750	
BI_APPS	SYSDEFAULTSUBCLASS 4	0	1	5.230	
BATCH	SYSDEFAULTSUBCLASS 1	0	1	25.600	

Die von dieser Abfrage zurückgegebenen Daten können ausreichen, um zu zeigen, dass die Leistungsminderung in der Serviceklasse BI_APPS auftritt, da ihre durchschnittliche Aktivitätslaufzeit signifikant höher als normal ist. Diese Situation könnte darauf hindeuten, dass die verfügbaren Ressourcen für diese Serviceklasse erschöpft sind.

Wenn anhand der Durchschnittswerte für die Serviceklassen für alle Datenbankpartitionen das Problem nicht eingegrenzt werden kann, sollten Sie Durchschnittswerte für jede Datenbankpartition einzeln analysieren. Durch Zusammenfassung der Durchschnittswerte für jede Datenbankpartition zu einem globalen Durchschnitt können große Abweichungen zwischen Datenbankpartitionen verdeckt werden. In dieser Situation wird angenommen, dass jede Datenbankpartition als Koordinatorpartition verwendet wird. Wenn diese Annahme nicht stimmt, beträgt die an Nicht-Koordinatorpartitionen berechnete durchschnittliche Laufzeit Null.

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CAST(COORD_ACT_LIFETIME_AVG / 1000 AS DECIMAL(9,3)) AS AVGLIFETIME
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS(' ', ' ', -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME
```

SUPERCLASS_NAME	SUBCLASS_NAME	PART	AVGLIFETIME
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	0	3.425
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	1	2.752
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	2	8.230
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	3	0.593

In diesem Beispiel empfängt die Datenbankpartition 2 möglicherweise mehr Arbeit als gewöhnlich, da ihre durchschnittliche Aktivitätslaufzeit viel höher ist als die der anderen Datenbankpartitionen.

Viele unterschiedliche Situationen können die Ursache einer Leistungsminderung sein. Verwenden Sie die folgenden Prinzipien, um die von den Workload-Management-Tabellenfunktionen bereitgestellten Informationen bestmöglich zu nutzen:

- Behandeln Sie große Mengen von Sperrkonflikten auf der Ebene der Anwendungslogik und -umgebung (Isolationsstufe usw.).
- Wenn sich die Serviceklasse ihren Schwellenwerten nähert (Anzahl gleichzeitiger Anforderungen usw.), müssen Sie eventuell die Schwellenwerte erhöhen.
- Wenn die einer Serviceklasse zugeordneten Ressourcen nahezu erschöpft sind, kann die Zuordnung zu den Betriebssystemserviceklassen die Ursache des Problems sein (d. h. die der Serviceklasse entsprechende Betriebssystemserviceklasse erhält nicht ausreichend CPU-Ressourcen, E/A-Bandbreite und andere Ressourcen).
- Möglicherweise werden mehr Aktivitäten als erwartet in der Serviceklasse ausgeführt, wodurch mehr Ressourcen als normal verbraucht werden. Prüfen Sie die Anzahl abgeschlossener Aktivitäten, um festzustellen, ob die Menge der ausgeführten Arbeit in der Serviceklasse angemessen ist.
- Aktivitäten verbringen möglicherweise mehr Zeit in Warteschlangen, wenn mehr Aktivitäten als erwartet übergeben werden und Schwellenwerte für gemeinsamen Zugriff definiert sind. Prüfen Sie, ob die durchschnittliche Wartezeit sich um denselben Betrag erhöht hat wie die durchschnittliche Laufzeit. Wenn dies der Fall ist, verhalten sich die Warteschlangen wie erwartet. Wenn jedoch die Laufzeit inakzeptabel ist, sollten Sie der Serviceklasse mehr Ressourcen zuweisen und den Schwellenwert für gemeinsamen Zugriff verringern.

Beispiel: Untersuchen einer workloadbezogenen Leistungsminderung des Systems

Wenn Sie eine Leistungsminderung des Systems bemerken (z. B. dauert die Ausführung einiger Anwendungen viel länger als erwartet) und Sie nicht sicher sind, ob das Problem mit der Konfiguration der Workloads zusammenhängt, können Sie mit Tabellenfunktionsdaten das Problem untersuchen und erforderlichenfalls beheben.

Erstellen Sie zuerst eine Abfrage, die Daten über Serviceklassen und Datenbankpartitionen zusammenfasst, unter Verwendung der Daten aus der Tabellenfunktion WLM_GET_SERVICE_SUBCLASS_STATS. Definieren Sie für das erste und das zweite Argument leere Zeichenfolgen, und stellen Sie das dritte Argument auf -2 (ein Platzhalterzeichen) ein, um anzugeben, dass Daten für alle Serviceklassen auf allen Datenbankpartitionen zusammengestellt werden sollen.

Ihre Abfrage könnte der folgenden ähneln:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(SUM(COORD_ACT_COMPLETED_TOTAL)),1,13) AS ACTSCOMPLETED,
       SUBSTR(CHAR(SUM(COORD_ACT_ABORTED_TOTAL)),1,11) AS ACTSABORTED,
       SUBSTR(CHAR(MAX(CONCURRENT_ACT_TOP)),1,6) AS ACTSHW,
       CAST(CASE WHEN SUM(COORD_ACT_COMPLETED_TOTAL) = 0 THEN 0
              ELSE SUM(COORD_ACT_COMPLETED_TOTAL * COORD_ACT_LIFETIME_AVG)
              / SUM(COORD_ACT_COMPLETED_TOTAL) END / 1000 AS DECIMAL(9,3))
       AS ACTAVGLIFETIME
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS('', '', -2)) AS SCSTATS
GROUP BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME
```

SUPERCLASS_NAME	SUBCLASS_NAME	ACTSCOMPLETED	ACTSABORTED	ACTSHW	ACTAVGLIFETIME
SYSDEFAULTUSERCLASS	SYSDEFAULTSUBCLASS	20	0	1	3.750
SUP1	SUB1	40	0	8	7.223

Im Beispiel oben führt die Serviceunterklasse SUB1 in der Servicesuperklasse SUP1 gleichzeitig mehr Aktivitäten als gewöhnlich aus. Zur weiteren Untersuchung wollen Sie eventuell die Statistikdaten für Workloads prüfen, die dieser Serviceklasse zugeordnet sind. Ihre Abfrage könnte der folgenden ähneln:

```
SELECT SUBSTR(WLSTATS.WORKLOAD_NAME,1,22) AS WL_NAME,
       SUBSTR(CHAR(WLSTATS.DBPARTITIONNUM),1,4) AS PART,
       CONCURRENT_WLO_TOP AS WLO_HIGH_WTRMRK,
       CONCURRENT_WLO_ACT_TOP AS WLO_ACT_HIGH_WTRMRK
FROM TABLE(WLM_GET_WORKLOAD_STATS('', -2)) AS WLSTATS,
     TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES('', '', -2)) AS SCWLOS
WHERE WLSTATS.WORKLOAD_NAME = SCWLOS.WORKLOAD_NAME
AND SCWLOS.SERVICE_SUPERCLASS_NAME = 'SUP1'
AND SCWLOS.SERVICE_SUBCLASS_NAME = 'SUB1'
ORDER BY WL_NAME, PART;
```

WL_NAME	PART	WLO_HIGH_WTRMRK	WLO_ACT_HIGH_WTRMRK
LYNNSALES	0	2	8
LYNNSALES	1	0	0
SYSDEFAULTWORKLOAD	0	1	1
SYSDEFAULTWORKLOAD	1	0	0

Die Ausgabe zeigt, dass eine Anwendung in der Workload LYNNSALES gleichzeitig 8 Aktivitäten übergeben hat. Sie können erwägen, einen Schwellenwert hinzuzufügen, der den gemeinsamen Zugriff von Koordinatoraktivitäten für jedes Workloadvorkommen beschränkt.

Beispiel: Analysieren von Workloads nach Aktivitätstyp

Sie können mit Workload-Management-Tabellenfunktionen die Workloads in Ihrer Umgebung im Hinblick auf die Typen von ausgeführten Aktivitäten untersuchen.

In einigen Situationen sind Sie möglicherweise interessiert am Verhalten eines bestimmten Typs von Aktivitäten, z. B. LOAD-Aktivitäten. Sie können z. B. folgendermaßen beobachten, wieviele LOAD-Aktivitäten gleichzeitig im System aktiv sind:

```
SELECT COUNT(*)
FROM TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(CAST(NULL AS BIGINT), -2)) AS ACTS
WHERE ACTIVITY_TYPE = 'LOAD'
```

Sie können eine Anzahl der Aktivitäten eines bestimmten Typs, die seit der letzten Zurücksetzung der Workload-Management-Statistikdaten übergeben wurden, mit der Tabellenfunktion WLM_GET_WORK_ACTION_SET_STATS ermitteln (siehe folgendes Beispiel). Nehmen wir an, dass die Arbeitsklassen READCLASS und LOADCLASS für Aktivitäten vom Typ READ und vom Typ LOAD vorhanden sind (außerdem muss eine Arbeitsaktion für jede Arbeitsklasse vorhanden sein, da sonst Aktivitäten in der Arbeitsklasse nicht klassifiziert werden). * stellt alle Aktivitäten dar, die nicht in die Arbeitsklasse READCLASS oder LOADCLASS fallen.

```
SELECT SUBSTR(WORK_ACTION_SET_NAME,1,18) AS WORK_ACTION_SET_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(WORK_CLASS_NAME,1,15) AS WORK_CLASS_NAME,
       LAST_RESET,
       SUBSTR(CHAR(ACT_TOTAL),1,14) AS TOTAL_ACTS
FROM TABLE(WLM_GET_WORK_ACTION_SET_STATS('', -2)) AS WASSTATS
ORDER BY WORK_ACTION_SET_NAME, WORK_CLASS_NAME, PART
```

WORK_ACTION_SET_NAME	PART	WORK_CLASS_NAME	LAST_RESET	TOTAL_ACTS
AdminActionSet	0	ReadClass	2005-11-25-18.52.49.343000	8
AdminActionSet	1	ReadClass	2005-11-25-18.52.50.478000	0
AdminActionSet	0	LoadClass	2005-11-25-18.52.49.343000	2
AdminActionSet	1	LoadClass	2005-11-25-18.52.50.478000	0
AdminActionSet	0	*	2005-11-25-18.52.50.478000	0
AdminActionSet	1	*	2005-11-25-18.52.50.478000	0

Sie können die durchschnittliche Laufzeit der LOAD-Aktivitäten anzeigen, indem Sie ein Arbeitsaktionsset für die Zuordnung der LOAD-Aktivitäten zu einer bestimmten Serviceunterklasse erstellen. Nehmen wir z. B. an, dass Sie LOAD-Aktivitäten der Serviceunterklasse LOADSERVICECLASS unter der Servicesuperklasse MYSUPERCLASS zuordnen wollen. Dann können Sie die Tabellenfunktion WLM_GET_SERVICE_CLASS_STATS abfragen:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CAST(COORD_ACT_LIFETIME_AVG / 1000 AS DECIMAL(9,3)) AS AVGLIFETIME
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS('MYSUPERCLASS', 'LOADSERVICECLASS',
-2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART
```

SUPERCLASS_NAME	SUBCLASS_NAME	PART	AVGLIFETIME
SYSDEFAULTUSERCLASS	LOADSERVICECLASS	0	4691.242
SYSDEFAULTUSERCLASS	LOADSERVICECLASS	1	4644.740
SYSDEFAULTUSERCLASS	LOADSERVICECLASS	2	4612.431
SYSDEFAULTUSERCLASS	LOADSERVICECLASS	3	4593.451

Beispiel: Erkennen von blockierten Aktivitäten

Workload-Management-Tabellenfunktionen erleichtern die Ermittlung einer bestimmten Aktivität im Datenserver und gegebenenfalls deren Abbruch, ohne dass dabei die gesamte Anwendung beendet werden muss.

Ermitteln einer blockierten Aktivität

Im Folgenden sehen Sie ein Beispiel für die Ermittlung einer blockierten Abfrage. Nehmen wir an, dass ein Benutzer der Vertriebsabteilung (SALES), der die Anwendung SalesReport ausführt, sich beklagt, dass die Anwendung blockiert.

Ermitteln Sie die Anwendungskennung, und zeigen Sie dann mit der Tabellenfunktion WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES alle Aktivitäten an, die derzeit in dieser Anwendung ausgeführt werden. Wenn die Anwendungskennung z. B. 1 ist, könnte Ihre Abfrage der folgenden ähneln:

```
SELECT SUBSTR(CHAR(COORD_PARTITION_NUM),1,5) AS COORD,
SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
SUBSTR(CHAR(UOW_ID),1,5) AS UOWID,
SUBSTR(CHAR(ACTIVITY_ID),1,5) AS ACTID,
SUBSTR(CHAR(PARENT_UOW_ID),1,8) AS PARUOWID,
SUBSTR(CHAR(PARENT_ACTIVITY_ID),1,8) AS PARACTID,
SUBSTR(CHAR(ACTIVITY_TYPE),1,8) AS ACTTYPE,
SUBSTR(CHAR(NESTING_LEVEL),1,7) AS NESTING
FROM TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(1, -2)) AS WLOACTS
ORDER BY PART, UOWID, ACTID
```

COORD	PART	UOWID	ACTID	PARUOWID	PARACTID	ACTTYPE	NESTING
0	0	2	3	-	-	CALL	0
0	0	2	5	2	3	READ_DML	1

Für die Aktivität wird die UOW-ID 2 und die Aktivitäts-ID 5 ermittelt. Mit der Tabellenfunktion WLM_GET_SERVICE_CLASS_AGENTS können Sie dann erkennen, was die Agenten tun, die an dieser Aktivität arbeiten:

```
SELECT APPLICATION_HANDLE,
UOW_ID,
ACTIVITY_ID,
SUBSTR(REQUEST_TYPE,1,8) AS REQUEST_TYPE,
SUBSTR(EVENT_TYPE,1,8) AS EVENT_TYPE,
SUBSTR(EVENT_OBJECT,1,8) AS EVENT_OBJECT
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS(' ', ' ',
CAST(NULL AS BIGINT),
-2)) AS AGENTS
WHERE APPLICATION_HANDLE = 1
AND UOW_ID = 2
AND ACTIVITY_ID = 5
```

Die Aktivität könnte z. B. in der Warteschlange stehen, ausgeführt werden oder auf Grund einer Sperre warten. Wenn die Aktivität in der Warteschlange steht, sieht das Ergebnis folgendermaßen aus:

APPLICATION_HANDLE	UOW_ID	ACTIVITY_ID	REQUEST_TYPE	EVENT_TYPE	EVENT_OBJECT
1	2	5	OPEN	WAIT	WLM_QUEUE

Wenn die Aktivität ausgeführt wird, sieht das Ergebnis folgendermaßen aus:

APPLICATION_HANDLE	UOW_ID	ACTIVITY_ID	REQUEST_TYPE	EVENT_TYPE	EVENT_OBJECT
1	2	5	OPEN	PROCESS	REQUEST

Wenn die Aktivität auf Grund einer Sperre wartet, sieht das Ergebnis folgendermaßen aus:

```
APPLICATION_HANDLE UOW_ID ACTIVITY_ID REQUEST_TYPE EVENT_TYPE EVENT_OBJECT
-----
                1      2          5          OPEN    ACQUIRE    LOCK
```

Wenn Sie wissen, was die Aktivität gerade tut, können Sie entsprechend fortfahren:

- Wenn die Aktivität in der Warteschlange steht und der Benutzer angibt, dass die Abfrage so lange dauert, dass das Ergebnis nicht mehr von Interesse ist, oder wenn Sie meinen, dass die Abfrage zu viele Ressourcen verbraucht, können Sie sie abbrechen.
- Wenn die Aktivität wichtig ist und sich in der Warteschlange befindet, können Sie erwägen, andere weniger wichtige gerade ausgeführte Arbeit abzubrechen (und dadurch den gemeinsamen Zugriff verringern, so dass Aktivitäten aus der Warteschlange genommen werden). Vielleicht reicht es dem Benutzer auch, zu wissen, dass die Arbeit nicht blockiert ist, sondern nur auf die Gelegenheit wartet, ausgeführt zu werden.
- Wenn die Aktivität auf Grund einer Sperre wartet, können Sie mit dem Überwachungsprogramm für Momentaufnahme untersuchen, auf Grund welcher Sperren die Anwendung wartet.
- Wenn die Aktivität auf Grund einer Sperre wartet, die von einer Aktivität mit niedrigerer Priorität gehalten wird, können Sie erwägen, diese Aktivität abzubrechen.

Vielleicht finden Sie es auch hilfreich, die DML-Anweisung zu kennen, die die Aktivität 5 ausführt. Wenn Sie einen aktiven Aktivitätsereignismonitor haben, können Sie die Prozedur `WLM_CAPTURE_ACTIVITY_IN_PROGRESS` ausführen, um Informationen zur DML-Anweisung sowie andere Informationen zur Aktivität 5 während ihrer Ausführung zu erfassen. Im Gegensatz zum Anweisungsereignismonitor können Sie mit der Prozedur `WLM_CAPTURE_ACTIVITY_IN_PROGRESS` Informationen zu einer bestimmten Abfrage statt zu jeder zu diesem Zeitpunkt ausgeführten Anweisung erfassen. Sie können mit `WLM_GET_ACTIVITY_DETAILS` auch den Anweisungstext, abgeschnitten auf die ersten 1024 Zeichen, erhalten.

Wenn Sie beschließen, dass Sie die Aktivität abbrechen müssen, können Sie mit der Routine `WLM_CANCEL_ACTIVITY` die Aktivität abbrechen, ohne die absetzende Anwendung beenden zu müssen:

```
CALL WLM_CANCEL_ACTIVITY (1, 2, 5)
```

Die Anwendung, die die Aktivität abgesetzt hat, empfängt einen Fehler `SQL4725N`. Jede Anwendung, die mit negativem SQL-Code umgehen kann, kann diesen SQL-Code handhaben.

Ermitteln einer blockierten Aktivität auf Grund eines Sperrkonflikts

Nehmen wir an, dass Sie eine Situation haben, in der sich ein Benutzer über eine blockierte Anwendung beklagt. Nehmen wir weiter an, dass Sie entweder den Anwendungsnamen oder die Berechtigungs-ID der blockierten Anwendung kennen. Mit Hilfe dieser Informationen können Sie mit dem Befehl `LIST APPLICATIONS` die Anwendungskennung ermitteln. Angenommen, die vom Befehl `LIST APPLICATIONS` zurückgegebene Anwendungskennung lautet 2. Mit der Tabellen-

funktion WLM_GET_SERVICE_CLASS_AGENTS können Sie dann ermitteln, welche Agenten an dieser Aktivität arbeiten. Ihre Abfrage könnte der folgenden ähneln:

```
SELECT SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHANDLE,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(AGENT_TID),1,9) AS AGENT_TID,
       SUBSTR(CHAR(AGENT_TYPE),1,11) AS AGENTTYPE,
       SUBSTR(CHAR(EVENT_OBJECT),1,11) AS EVENTOBJECT,
       SUBSTR(CHAR(REQUEST_TYPE),1,7) AS REQTYPE,
       SUBSTR(CHAR(UOW_ID),1,6) AS UOW_ID,
       SUBSTR(CHAR(ACTIVITY_ID),1,6) AS ACT_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS(' ', ' ', 2, -2)) AS SCDETAILS
ORDER BY APPHANDLE, PART, AGENT_TID
```

APPHANDLE	PART	AGENT_TID	AGENTTYPE	EVENTOBJECT	REQTYPE	UOW_ID	ACT_ID
2	0	1	COORDINATOR	REQUEST	OPEN	2	1
2	1	3	SUBAGENT	LOCK	-	2	1

Die Ergebnisse zeigen, dass der Agent 1 auf eine ferne Antwort wartet. Wenn Sie sich den Agenten auf der fernen Partition ansehen, der an derselben Aktivität arbeitet, zeigt das Feld EVENTOBJECT an, dass der Agent darauf wartet, eine Sperre zu erhalten.

Als Nächstes muss ermittelt werden, wer der Eigentümer der Sperre ist. Sie können diese Information durch Aktivieren der Monitorschalter und Verwenden der Tabellenfunktion des Überwachungsprogramms für Momentaufnahme erhalten (siehe folgendes Beispiel):

```
SELECT AGENT_ID AS WAITING_FOR_LOCK,
       SUBSTR(APPL_ID_HOLDING_LK,1,40) AS HOLDING_LOCK,
       CAST(LOCK_MODE_REQUESTED AS SMALLINT) AS WANTED,
       CAST(LOCK_MODE AS SMALLINT) AS HELD
FROM TABLE(SNAPSHOT_LOCKWAIT('SAMPLE',-1)) AS SLW
```

WAITING_FOR_LOCK	HOLDING_LOCK	WANTED	HELD
2	*LOCAL.DB2.060131021547	9	5

Sie können den Sperrereigentümer auch mit der folgenden Befehlsfolge ermitteln:

```
db2pd -db database alias -locks
db2pd -db database alias -transactions
```

Wenn Sie die blockierte Aktivität abbrechen wollen, können Sie die Prozedur WLM_CANCEL_ACTIVITY verwenden. Wenn der erfolgreiche Abschluss der blockierten Anwendung wichtiger ist als der erfolgreiche Abschluss der Anwendung, die Eigentümer der Sperre ist, können Sie die Beendigung dieser Anwendung erzwingen.

Beispiel: Erfassen von Informationen zu einer Aktivität für die spätere Analyse

Mit den Workload-Management-Tools können Sie Informationen zu einer Aktivität zur späteren Analyse erfassen.

Nehmen wir an, Sie haben eine gespeicherte Prozedur namens MYSCHEMA.MYS-LOWSTP, und diese wird langsamer als gewöhnlich ausgeführt. Sie erhalten allmählich Klagen über diese Situation und beschließen, die Ursache der Leistungsminderung zu untersuchen. Wenn die Untersuchung während der Ausführung der

gespeicherten Prozedur nicht praktikabel ist, können Sie Informationen zur Aktivität der gespeicherten Prozedur und aller darin verschachtelter Aktivitäten erfassen.

Nehmen wir an, dass Sie einen aktiven Aktivitätsereignismonitor namens DB2ACTIVITIES haben. Sie können dann eine Arbeitsklasse für CALL-Anweisungen erstellen, die auf das Schema der gespeicherten Prozedur MYSCHEMA.MYSLOWSTP angewendet werden. Dann können Sie eine Arbeitsaktion erstellen, um die CALL-Aktivität und alle verschachtelten Aktivitäten einer Serviceklasse zuzuordnen, für die die Sammlung von Aktivitäten aktiviert ist. Die CALL-Aktivität und alle darin verschachtelten Aktivitäten werden an den Ereignismonitor gesendet. Im Folgenden sehen Sie Beispiele für den DDL-Code, der zum Erstellen der Workload-Management-Objekte erforderlich ist:

```
CREATE SERVICE CLASS SC1;
CREATE WORKLOAD WL1 APPLNAME ('DB2BP') SERVICE CLASS SC1;
CREATE SERVICE CLASS PROBLEMQUERIESSC UNDER SC1 COLLECT ACTIVITY DATA ON COORDINATOR WITH DETAILS;

CREATE WORK CLASS SET PROBLEMQUERIES
(WORK CLASS CALLSTATEMENTS WORK TYPE CALL ROUTINES IN SCHEMA MYSCHEMA);

CREATE WORK ACTION SET DATABASEACTIONS FOR SERVICE CLASS SC1 USING WORK CLASS SET PROBLEMQUERIES
(WORK ACTION CAPTURECALL ON WORK CLASS CALLSTATEMENTS MAP ACTIVITY WITH NESTED TO PROBLEMQUERIESSC);
```

Nachdem die gespeicherte Prozedur MYSCHEMA.MYSLOWSTP ausgeführt wird, können Sie die folgende Abfrage absetzen, um die Anwendungskennung, die UOW-ID und die Aktivitäts-ID für die Aktivität zu ermitteln:

```
SELECT AGENT_ID,
       UOW_ID,
       ACTIVITY_ID
FROM ACTIVITY_DB2ACTIVITIES
WHERE SC_WORK_ACTION_SET_ID = (SELECT ACTIONSETID
                               FROM SYSCAT.WORKACTIONSETS
                               WHERE ACTIONSETNAME = 'DATABASEACTIONS')
AND SC_WORK_CLASS_ID = (SELECT WORKCLASSID
                        FROM SYSCAT.WORKCLASSES
                        WHERE WORKCLASSNAME = 'CALLSTATEMENTS'
                        AND WORKCLASSSETID =
                          (SELECT WORKCLASSSETID FROM SYSCAT.WORKACTIONSETS WHERE ACTIONSETNAME
                           = 'DATABASEACTIONS'));
```

Wenn die erfasste Aktivität eine Anwendungskennung von 1, eine UOW-ID von 2 und eine Aktivitäts-ID von 3 hat, werden die folgenden Ergebnisse generiert:

AGENT_ID	UOW_ID	ACTIVITY_ID
1	2	3

Mit diesen Informationen können Sie die folgende Abfrage auf den Tabellen ACTIVITY_DB2ACTIVITIES und ACTIVITYSTMT_DB2ACTIVITIES absetzen, um zu ermitteln, wo die Aktivität ihre Zeit verbracht hat:

```
WITH RAH (LEVEL, APPL_ID, PARENT_UOW_ID, PARENT_ACTIVITY_ID,
         UOW_ID, ACTIVITY_ID, STMT_TEXT, TIME_CREATED, TIME_COMPLETED) AS
(SELECT 1, ROOT.APPL_ID, ROOT.PARENT_UOW_ID,
     ROOT.PARENT_ACTIVITY_ID, ROOT.UOW_ID, ROOT.ACTIVITY_ID,
     ROOTSTMT.STMT_TEXT, ROOT.TIME_CREATED, ROOT.TIME_COMPLETED
FROM ACTIVITY_DB2ACTIVITIES ROOT, ACTIVITYSTMT_DB2ACTIVITIES ROOTSTMT
WHERE ROOT.APPL_ID = ROOTSTMT.APPL_ID AND ROOT.AGENT_ID = 1
     AND ROOT.UOW_ID = ROOTSTMT.UOW_ID AND ROOT.UOW_ID = 2
     AND ROOT.ACTIVITY_ID = ROOTSTMT.ACTIVITY_ID AND ROOT.ACTIVITY_ID = 3
UNION ALL
SELECT PARENT.LEVEL + 1, CHILD.APPL_ID, CHILD.PARENT_UOW_ID,
     CHILD.PARENT_ACTIVITY_ID, CHILD.UOW_ID,
     CHILD.ACTIVITY_ID, CHILDSTMT.STMT_TEXT, CHILD.TIME_CREATED,
```

```

        CHILD.TIME_COMPLETED
FROM RAH PARENT, ACTIVITY_DB2ACTIVITIES CHILD,
ACTIVITYSTMT_DB2ACTIVITIES CHILDSTMT
WHERE PARENT.APPL_ID = CHILD.APPL_ID AND
CHILD.APPL_ID = CHILDSTMT.APPL_ID AND
PARENT.UOW_ID = CHILD.PARENT_UOW_ID AND
CHILD.UOW_ID = CHILDSTMT.UOW_ID AND
PARENT.ACTIVITY_ID = CHILD.PARENT_ACTIVITY_ID AND
CHILD.ACTIVITY_ID = CHILDSTMT.ACTIVITY_ID AND
PARENT.LEVEL < 64
)
SELECT UOW_ID, ACTIVITY_ID, SUBSTR(STMT_TEXT,1,40),
TIMESTAMPDIFF(2, CHAR(TIME_COMPLETED - TIME_CREATED)) AS
LIFE_TIME
FROM RAH
ORDER BY UOW_ID, ACTIVITY_ID;

```

Die Ergebnisse ähneln der folgenden Ausgabe:

UOW_ID	ACTIVITY_ID	STMT_TEXT	LIFE_TIME
2	3	CALL SLOWPROC	1000
2	4	SELECT COUNT(*) FROM ORG	1
2	5	SELECT * FROM MYHUGETABLE	999

Die Ergebnisse deuten darauf hin, dass die gespeicherte Prozedur ihre meiste Zeit bei der Abfrage der Tabelle MYHUGETABLE verbringt. Als Nächstes müssen Sie untersuchen, welche Änderungen an der Tabelle MYHUGETABLE für die Verlangsamung der darauf ausgeführten Abfragen verantwortlich sein könnten.

Wenn viele gespeicherte Prozeduren gleichzeitig ausgeführt werden, ergibt sich bei der Durchführung der Analyse ein größerer Systemaufwand. Zur Behebung dieses Problems können Sie eine Workload und eine Serviceklasse für die Ausführung einer gespeicherten Prozedur erstellen, die von einer bestimmten Berechtigungs-ID und/oder einer bestimmten Anwendung abgesetzt wird. Dann können Sie mit der oben beschriebenen Methode das Verhalten der gespeicherten Prozedur analysieren.

Beispiel: Untersuchen der Agentenauslastung nach Serviceklasse

Die Workload-Management-Lösung stellt die Tabellenfunktion WLM_GET_SERVICE_CLASS_AGENTS zur Verfügung, mit der Sie die relative Verteilung von Agenten auf die Serviceklassen ermitteln können.

Es können Situationen auftreten, in denen eine Datenserverressource, z. B. ein Agent, von einer Gruppe von Benutzern oder von einer Anwendung zu stark genutzt wird. Nehmen wir z. B. an, dass eine Benutzergruppe fast alle verfügbaren Agenten nutzt und dass ein Benutzer, der nicht zu dieser Gruppe gehört, sich bei Ihnen darüber beklagt.

Als Erstes muss ermittelt werden, wieviele Agenten für jede Serviceklasse arbeiten. Sie können eine Abfrage wie die folgende verwenden:

```

SELECT SUBSTR(AGENTS.SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
SUBSTR(AGENTS.SERVICE_SUBCLASS_NAME,1,19) AS SUBCLASS_NAME,
COUNT(*) AS AGENT_COUNT
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS('', '', CAST(NULL AS BIGINT), -2)) AS AGENTS
WHERE AGENT_STATE = 'ACTIVE'
GROUP BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME
ORDER BY SERVICE_SUPERCLASS_NAME, SERVICE_SUBCLASS_NAME

SUPERCLASS_NAME    SUBCLASS_NAME    AGENT_COUNT

```

```

-----
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS      7
TEST                SYSDEFAULTSUBCLASS      20

```

Wenn Sie zu dem Schluss kommen, dass eine bestimmte Serviceklasse mehr als den ihr zustehenden Anteil von Agenten nutzt, können Sie Maßnahmen ergreifen, um die Anzahl der zulässigen Aktivitäten für eine Workload oder eine Serviceklasse zu beschränken. Alternativ können Sie die Anzahl der Verbindungen für eine Serviceklasse beschränken.

Beispiel: Optimieren einer Workload-Management-Konfiguration, wenn Kapazitätsplanungsinformationen verfügbar sind

Wenn Sie Kapazitätsplanung durchgeführt haben, sollten Sie Informationen zu den Typen von Benutzern und ihren erwarteten Antwortzeiten haben. Mit Hilfe dieser Informationen können Sie Ihre Workload-Management-Konfiguration aufbauen, deren Effektivität bestimmen und sie optimieren.

Nehmen wir an, dass Sie Kapazitätsplanung durchgeführt haben und dass die Daten in der folgenden Tabelle die entsprechenden Ergebnisse für Arbeitstypen und die angestrebten Antwortzeiten darstellen:

Tabelle 41. Ergebnisse der Kapazitätsplanung

Arbeitstyp	Anwendung	Ziel	Wichtigkeit	Erwarteter Durchsatz
Auftragserfassung	orderentryapp.exe	Durchschnittliche Antwortzeit < 1 Sekunde	Hoch	10 000 (sowohl Einfügungen als auch Aktualisierungen) pro Tag
Business-Intelligence-Abfragen	businessobjects.exe	Durchschnittliche Antwortzeit < 10 Sekunden	Hoch	100 Abfragen pro Tag
Stapelverarbeitung	batchapp.exe	Maximierung des Durchsatzes	Niedrig	5000 Aktualisierungen pro Tag
Sonstige	Alle anderen Anwendungen	Beste Leistung	Niedrig	100 Aktivitäten pro Tag

Basierend auf den Daten in der Tabelle oben könnten Sie drei Serviceklassen (ORDER_ENTRY_SC, BI_QUERIES_SC und BATCH_SC) und drei Workloads (ORDER_ENTRY_WL, BI_QUERIES_WL und BATCH_WL) erstellen, um die Arbeit den Serviceklassen zuzuordnen. Nach der Erstellung der Serviceklassen und der Workloads könnten Sie einen Statistikereignismonitor erstellen, um zusammengefasste Aktivitätsinformationen wie das Histogramm der Aktivitätslaufzeit für jede Serviceklasse zu erfassen. Nehmen wir an, dass die Daten in der folgenden Tabelle einen Vergleich der durchschnittlichen täglichen Anzahl von Aktivitäten in jeder Serviceklasse (berechnet aus dem Histogramm der Aktivitätslaufzeit) mit den Volumina darstellen, die in der Kapazitätsplanungsübung vorhergesagt wurden:

Tabelle 42. Aktivitäten pro Tag

Serviceklasse	Vorhergesagte Aktivitäten pro Tag	Tatsächliche Aktivitäten pro Tag
ORDER_ENTRY_SC	10 000	9700
BI_QUERIES_SC	100	115
BATCH_SC	5000	5412
SYSDEFAULTUSERCLASS	100	85

Die beobachteten Daten zeigen, dass die Kapazitätsplanungsschätzungen präzise waren. Die Daten in der folgenden Tabelle vergleichen die durchschnittlichen Aktivitätslaufzeiten (aus dem Histogramm der Aktivitätslaufzeit) mit den bei der Kapazitätsplanung festgelegten angestrebten Antwortzeiten und zeigen, dass die Aktivitäten in der Serviceklasse BI_QUERIES_SC nicht die zugehörigen Antwortzeitziele erreichen.

Tabelle 43. Antwortzeiten

Serviceklasse	Angestrebte Antwortzeit	Tatsächliche durchschnittliche Laufzeit
ORDER_ENTRY_SC	< 1 Sekunde	0,8 Sekunden
BI_QUERIES_SC	< 10 Sekunden	30 Sekunden
BATCH_SC		2 Sekunden
SYSDEFAULTUSERCLASS		10 Minuten

Mit den Workload-Management-Schnittstellen können Sie unterschiedliche Strategien verwenden, wenn Sie sich mit dem Problem befassen, dass Business-Intelligence-Abfragen nicht ihre angestrebten Antwortzeiten erreichen:

- Beschränken des gemeinsamen Zugriffs von Serviceklassen mit niedrigerer Priorität
- Zulassen, dass der Betriebssystem-Workload-Manager weniger wichtigen Serviceklassen weniger CPU-Ressourcen zur Verfügung stellt
- Ändern der Agenten- und E/A-Vorabsezugriffsprioritäten für die Serviceklassen
- Verwenden einer beliebigen Kombination der vorherigen drei Methoden

Nehmen wir an, die CPU ist die Ressource, wegen der die Business-Intelligence-Abfragen ihre Ziele nicht erreichen können. Nehmen wir weiterhin an, dass Sie den Betriebssystem-Workload-Manager verwenden, um der Serviceklasse SYSDEFAULTUSERCLASS weniger CPU-Ressourcen als anderen Serviceklassen zur Verfügung zu stellen. Sie können dann zusammengefasste Aktivitätsinformationen über einen Zeitraum von mehreren Tagen erfassen, um zu beobachten, ob die Änderung der CPU-Zuordnung die erwarteten Ergebnisse bringt. Die Daten in der folgenden Tabelle zeigen einen weiteren Vergleich zwischen den angestrebten Antwortzeiten und den tatsächlichen durchschnittlichen aus den Histogrammen berechneten Laufzeiten, nachdem Sie die Änderungen über den Betriebssystem-Workload-Manager vorgenommen haben. Alle Serviceklassen erreichen jetzt ihre angestrebte Antwortzeiten. Durch die CPU-Neuzuordnung haben sich die Antwortzeiten von Aktivitäten in der Serviceklasse SYSDEFAULTUSERCLASS verdoppelt.

Tabelle 44. Antwortzeiten nach der Rekonfiguration

Serviceklasse	Angestrebte Antwortzeit	Tatsächliche durchschnittliche Laufzeit
ORDER_ENTRY_SC	< 1 Sekunde	0,6 Sekunden
BI_QUERIES_SC	< 10 Sekunden	9,5 Sekunden
BATCH_SC		1,5 Sekunden
SYSDEFAULTUSERCLASS		20 Minuten

Beispiel: Optimieren einer Workload-Management-Konfiguration, wenn keine Kapazitätsplanungsinformationen verfügbar sind

Mit den Workload-Management-Tools können Sie eine Workload-Management-Konfiguration entwerfen, überwachen und optimieren, selbst wenn Sie keine Kapazitätsanalysedaten für die Entwicklung der Konfiguration haben.

Nehmen wir an, dass Sie anfangs nicht wissen, welche Workloads und Serviceklassen erstellt werden müssen, weil Sie entweder die Arbeit auf dem System nicht genau kennen oder noch nicht wissen, welche Workloads für stabile Ausführungsergebnisse erforderlich sind. Nehmen wir weiter an, dass Sie wissen, dass einige Anwendungen Anforderungen bezüglich der Antwortzeit haben, aber Sie noch nicht wissen, wieviele andere Anwendungen mit solchen zeitkritischen Anwendungen um Ressourcen konkurrieren. Sie können die Workload-Management-Überwachungsfunktionen verwenden, um dies zu ermitteln.

Gehen Sie wie folgt vor, um eine Workload-Management-Konfiguration einzurichten, die Überwachungsdaten als Grundlage verwendet:

1. Klassifizieren Sie die Anwendungen, von denen Sie wissen, dass sie wichtig sind. Sie müssen diese Anwendungen eingrenzen und ihnen einen ausreichenden Anteil der Systemressourcen zuteilen.
2. Für die restliche Arbeit erfassen Sie Statistikdaten für die größten Aktivitäten, da diese Aktivitäten die größte Auswirkung pro Aktivität auf das System haben.
3. Analysieren Sie die in Schritt 2 erfassten Aktivitätsinformationen.
4. Wiederholen Sie Schritt 1 bis 3 für den noch nicht klassifizierten Teil der Arbeit. Wiederholen Sie diesen Schritt, bis Sie wissen, dass es sich nicht mehr lohnt, die verbleibende noch nicht klassifizierte Arbeit weiter zu klassifizieren.

Die folgenden Abschnitte enthalten Informationen zur Durchführung dieser Schritte.

Schritt 1. Eingrenzen der als wichtig bekannten Anwendungen und Zuteilen der angemessenen Ressourcenmenge für diese Anwendungen

Nehmen wir an, dass Sie zwei wichtige Business-Intelligence-Anwendungen, BI1 und BI2, haben und dass Sie die Antwortzeiten für diese Anwendungen minimieren müssen. Sie können für diese beiden Anwendungen Workloads erstellen und sie einer Serviceklasse namens MOSTIMPORTANT zuordnen, für die Sie Systemressourcen zuordnen können.

Auf dem Betriebssystem AIX können Sie AIX Workload Manager verwenden, um eine Serviceklasse namens MOSTIMPORTANT zu erstellen und dieser Serviceklasse eine garantierte Menge an Ressourcen zuzuordnen.

Auf dem DB2-Datenserver erstellen Sie die erforderlichen Serviceklassen und Workloads wie folgt:

```
CREATE SERVICE CLASS MOSTIMPORTANT OUTBOUND CORRELATOR 'MOSTIMPORTANT'  
CREATE WORKLOAD BI1WORKLOAD APPLNAME ('BI1') SERVICE CLASS MOSTIMPORTANT  
CREATE WORKLOAD BI2WORKLOAD APPLNAME ('BI2') SERVICE CLASS MOSTIMPORTANT
```

Für die Zwecke dieses Beispiels nehmen wir an, dass noch ein beträchtlicher Teil der Systemauslastung nicht berücksichtigt ist, selbst nachdem Sie die bekannten Anwendungen berücksichtigt haben. Sie müssen daher diese Arbeit besser verstehen und möglicherweise steuern.

Schritt 2. Erfassen von Statistikdaten für die größten Aktivitäten in der restlichen nicht klassifizierten Arbeit

Eine lang laufende Aktivität hat eine größere individuelle Auswirkung auf das System als eine Aktivität mit kurzer Laufzeit, da die lang laufende Aktivität über einen längeren Zeitraum Systemressourcen belegt. Das Erfassen von Informationen zu lang laufenden Aktivitäten erzeugt jedoch keinen höheren Systemaufwand als das Erfassen von Informationen zu Aktivitäten mit kurzer Laufzeit. Die beste Methode zum Erfassen von Informationen zum Großteil der Arbeit ist daher, zuerst Informationen zu den am längsten laufenden Aktivitäten zu erfassen.

Beginnen Sie mit dem Erfassen von Aktivitätsinformationen, indem Sie zuerst eine Aktivitätslaufzeit festlegen, ab der Sie Aktivitätsinformationen erfassen wollen. Sie können dies vereinfachen, indem Sie die Erfassung für einen Teil der nicht klassifizierten Aktivitäten, z. B. 30%, auswählen und dann das Histogramm der Aktivitätslaufzeit für diese Aktivitäten beobachten. Lassen Sie das System arbeiten, so dass die speicherinternen Statistikdaten aktualisiert werden. Führen Sie dann die Prozedur WLM_COLLECT_STATS aus, um die Statistikdaten an einen aktiven Statistikereignismonitor zu senden.

Mit der folgenden Abfrage können Sie das Histogramm der Aktivitätslaufzeit für die Serviceklasse SYSDEFAULTUSERCLASS in Form einer Tabelle erhalten, die den Anteil der Gesamtaktivitäten darstellt, der in den jeweiligen Laufzeitbereich fiel. Bei dieser Abfrage wird davon ausgegangen, dass die Datenbank nicht partitioniert ist.

```
WITH TOTAL AS (
SELECT PARENTSERVICECLASSNAME,
      SERVICECLASSNAME,
      HIST.HISTOGRAM_TYPE,
      SUM(NUMBER_IN_BIN) AS NUMBER_IN_BIN
FROM HISTOGRAMBIN_DB2STATISTICS AS HIST,
      SYSCAT.SERVICECLASSES SC
WHERE
  HIST.SERVICE_CLASS_ID = SC.SERVICECLASSID
  AND HIST.TOP >= 0
  AND SC.PARENTSERVICECLASSNAME = 'SYSDEFAULTUSERCLASS'
  AND SC.SERVICECLASSNAME = 'SYSDEFAULTSUBCLASS'
  AND HIST.HISTOGRAM_TYPE = 'COORDACTLIFETIME'
GROUP BY PARENTSERVICECLASSNAME, SERVICECLASSNAME, HISTOGRAM_TYPE)
SELECT CAST(CAST(TOP AS DOUBLE) / 60000 AS DECIMAL(14,3)) AS TOP_IN_MINUTES,
       CAST(100 * CAST(SUM(HIST.NUMBER_IN_BIN) AS DOUBLE) / TOTAL.NUMBER_IN_BIN AS DECIMAL(4,2))
       AS PERCENT_IN_BIN
FROM HISTOGRAMBIN_DB2STATISTICS AS HIST,
      SYSCAT.SERVICECLASSES SC,
      TOTAL
WHERE HIST.SERVICE_CLASS_ID = SC.SERVICECLASSID
  AND HIST.TOP >= 0
  AND SC.PARENTSERVICECLASSNAME = 'SYSDEFAULTUSERCLASS'
  AND SC.SERVICECLASSNAME = 'SYSDEFAULTSUBCLASS'
  AND HIST.HISTOGRAM_TYPE = 'COORDACTLIFETIME'
  AND TOTAL.PARENTSERVICECLASSNAME = SC.PARENTSERVICECLASSNAME
  AND TOTAL.SERVICECLASSNAME = SC.SERVICECLASSNAME
  AND TOTAL.HISTOGRAM_TYPE = HIST.HISTOGRAM_TYPE
GROUP BY TOP, SC.PARENTSERVICECLASSNAME, SC.SERVICECLASSNAME, HIST.HISTOGRAM_TYPE, TOTAL.NUMBER_IN_BIN;
```

TOP_IN_MINUTES	PERCENT_IN_BIN
0.000	0.00
0.000	0.00
0.000	0.00
0.000	0.00
0.000	0.00
0.000	0.00
0.000	0.00
0.000	0.00
0.000	0.00
0.001	0.00
0.001	0.00
0.002	0.00
0.004	0.00
0.006	0.00
0.009	0.00
0.014	0.00
0.021	0.00
0.033	0.00
0.050	0.00
0.077	0.00
0.118	0.00
0.180	0.00
0.274	0.00
0.419	0.00
0.639	0.00
0.975	0.00
1.488	0.00
2.269	0.00
3.462	0.00
5.280	0.00
8.054	0.00
12.286	0.00
18.740	0.00
28.584	10.00
43.600	15.00
66.505	45.00
101.442	23.00
154.731	5.00
236.015	2.00
360.000	0.00

Die folgende Abbildung zeigt die Ergebnisse der obigen Abfrage in Form eines Diagramms:

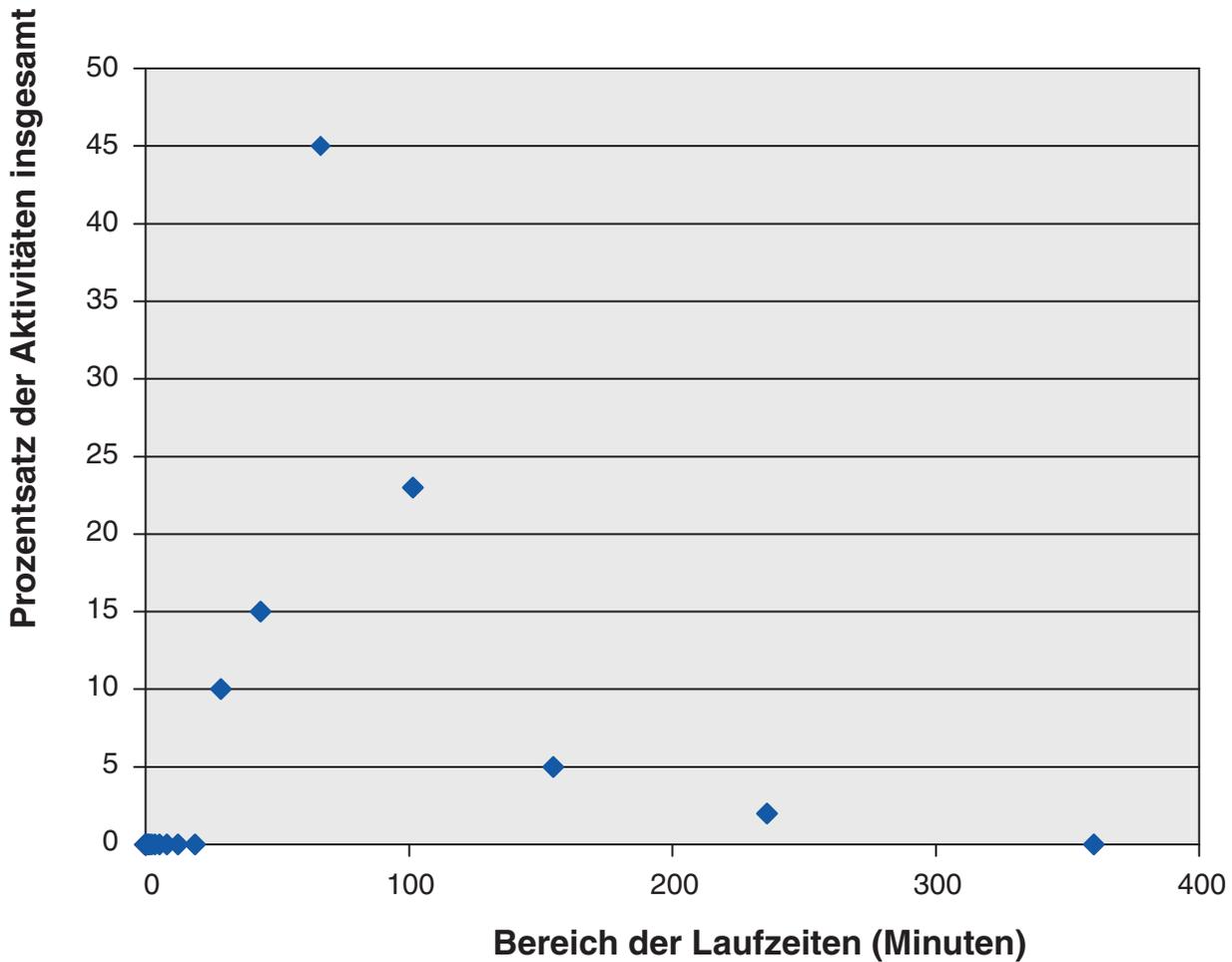


Abbildung 27. Histogramm der Aktivitätslaufzeit nicht klassifizierter Aktivitäten

In diesem Beispiel fallen 30% der Aktivitäten in den Bereich einer Laufzeit von 101 Minuten oder mehr. Zum Erfassen von Informationen zu diesen Aktivitäten erstellen Sie einen Aktivitätslaufzeitschwellenwert von 100 Minuten mit den Optionen CONTINUE und COLLECT ACTIVITY DATA (siehe folgendes Beispiel). Wenn gegen diesen Schwellenwert verstoßen wird, werden Aktivitätsinformationen an einen aktiven Aktivitätsereignismonitor gesendet.

```
CREATE THRESHOLD COLLECTLONGESTRUNNING30PERCENT
FOR SERVICE CLASS SYSDEFAULTSUBCLASS UNDER SYSDEFAULTUSERCLASS
ACTIVITIES ENFORCEMENT DATABASE ENABLE
WHEN ACTIVITYTOTALTIME > 100 MINUTES COLLECT ACTIVITY DATA CONTINUE
```

Lassen Sie das System arbeiten, damit Daten erfasst werden.

Wenn der Systemaufwand für das Erfassen von Informationen zu den 30% am längsten laufenden Aktivitäten akzeptabel ist, dann können Sie erlauben, dass die Datenerfassung einige Stunden oder einige Tage fortgesetzt wird. Mit den erfassten Daten können Sie ermitteln, welche Benutzer und Anwendungen die am längsten laufenden noch nicht klassifizierten 30% der DML-Aktivitäten erzeugen. Unter diesen Aktivitäten könnten einige sein, die zeitkritisch sind. Sie könnten auch einige Überraschungen erleben, z. B. Anwendungen mit niedriger Priorität, die eine beträchtliche Anzahl großer Aktivitäten ausführen. Wenn Sie mit der Erfassung und der Analyse der Daten fertig sind, können Sie den Schwellenwert löschen.

Schritt 3. Analysieren der im vorherigen Schritt erfassten Aktivitätsinformationen

Sie können die Informationen, die Sie im vorherigen Schritt zu Aktivitäten erfasst haben, nach der übergebenden Anwendung analysieren. Sie könnten die folgende Abfrage absetzen:

```
SELECT SUBSTR (APPL_NAME, 1,16) APPLICATION_NAME,
        AVG(TIMESTAMPDIFF(4, CHAR(TIME_COMPLETED - TIME_CREATED))) AS AVG_LIFETIME_MINUTES
        COUNT(*) AS ACTIVITY_COUNT
FROM ACTIVITY_DB2ACTIVITIES
GROUP BY APPL_NAME
ORDER BY APPL_NAME
```

APPLICATION_NAME	AVG_LIFETIME_MINUTES	ACTIVITY_COUNT
MOSTLYSMALL1	120	21
MOSTLYSMALL2	110	15
UNIMPORTANTAPP	150	10213

Eine Analyse der Aktivitäten nach der übergebenden Anwendung zeigt, dass eine große Zahl der am längsten laufenden Aktivitäten von der Anwendung UNIMPOR-TANTAPP, einer relativ unwichtigen Anwendung, übergeben wurden. Sie können eine Workload verwenden, um diese Anwendung von den anderen nicht klassifi-zierten Anwendungen abzugrenzen und sie einer Serviceklasse namens BESTEF-FORT zuordnen, die Ressourcen nur dann erhält, wenn der Ressourcenbedarf aller anderen Aktivitäten erfüllt ist.

Gemäß den obigen Ergebnissen scheinen die verbleibenden Anwendungen in der Standardserviceklasse wenige große Aktivitäten zu übergeben. Vielleicht finden Sie es lohnend, die Erfassung von Aktivitäten zu wiederholen, die in der Standard-serviceklasse ausgeführt werden, ohne die Erfassung auf lang laufende Aktivitäten einzuschränken.

Schritt 4. Wiederholen von Schritt 1 bis 3 für den noch nicht klassifizierten Teil der Arbeit, bis es nicht mehr lohnt, die verbleibende nicht klassifizierte Arbeit weiter zu klassifizieren

Jetzt, da die zwei wichtigen Anwendungen in der Serviceklasse MOSTIMPOR-TANT und die unwichtige Anwendung in der Serviceklasse BESTEFFORT ausgeführt werden, wird viel weniger Arbeit in der Standardbenutzerserviceklasse ausgeführt. In dieser Situation kann es weniger aufwendig sein, Informationen zu jeder Aktivität in dieser Serviceklasse zu erfassen. Alternativ brauchen Sie mögli-cherweise die Arbeit nicht weiter zu unterteilen und können an diesem Punkt auf-hören. Nehmen wir an, Sie wollen Informationen zu den verbleibenden Aktivitäten erfassen, für den Fall, dass die verbleibende Arbeit mit Überraschungen aufwartet. Sie können dies erreichen, indem Sie COLLECT ACTIVITY DATA für die Standard-benutzerserviceklasse angeben und einen Aktivitätsereignismonitor erstellen:

```
ALTER SERVICE CLASS SYSDEFAULTSUBCLASS UNDER SYSDEFAULTUSERCLASS
COLLECT ACTIVITY DATA ON COORDINATOR WITHOUT DETAILS
```

Lassen Sie das System arbeiten, damit Daten erfasst werden. Sie können die Ergeb-nisse wie in Schritt 3 analysieren.

```
SELECT SUBSTR (APPL_NAME,1,16) APPLICATION_NAME,
        AVG(TIMESTAMPDIFF(4, CHAR(TIME_COMPLETED - TIME_CREATED))) AS AVG_LIFETIME_MINUTES
        COUNT(*) AS ACTIVITY_COUNT
FROM ACTIVITY_DB2ACTIVITIES
GROUP BY APPL_NAME
ORDER BY APPL_NAME
```

APPLICATION_NAME	AVG_LIFETIME_MINUTES	ACTIVITY_COUNT
MOSTLYSMALL1	5	1501
MOSTLYSMALL2	7	124
ONLYSMALL	2	10123

Die Ergebnisse zeigen, dass die Anwendung ONLYSMALL die Mehrheit der nicht klassifizierten Aktivitäten erzeugt. Da diese Anwendung nicht in den Ergebnissen enthalten war, als Sie Informationen zu den größten Aktivitäten erfasst haben, können Sie annehmen, dass ONLYSMALL während des Zeitraums der Datenerfassung keine großen Abfragen erzeugte.

Beispiel: Erkennen von Aktivitäten mit niedrigem geschätztem Aufwand und langer Laufzeit

Das folgende Beispiel zeigt, wie Sie mit Arbeitsklassen, Arbeitsaktionssets, Schwellenwerten und der Aktivitätssammlung Aktivitäten ermitteln können, die einen geringen geschätzten Aufwand aber eine hohe Laufzeit haben. Diese Situation könnte darauf hindeuten, dass der geschätzte Aufwand (in Timerons) auf Grund von veralteten Tabellen- und Indexstatistikdaten ungenau ist.

Als Erstes muss ein Arbeitsklassenset mit einer Arbeitsklasse erstellt werden, mit der Aktivitäten mit einem niedrigen geschätzten Aufwand ermittelt werden. Zum Beispiel:

```
CREATE WORK CLASS SET WCS1 (WORK CLASS SMALLDML WORK TYPE DML FOR TIMERONCOST
FROM 0 TO 500)
```

Als Nächstes erstellen Sie dann ein Datenbankarbeitsaktionsset mit einer Arbeitsaktion, die einen Schwellenwert für die Gesamtzeit der Aktivität auf die Arbeitsklasse SMALLDML anwendet. Die Schwellenwertaktion ist CONTINUE, und die Option COLLECT ACTIVITY DATA ist angegeben, so dass eine Aktivität, die einen Schwellenwertverstoß verursacht, bei ihrem Abschluss an den Aktivitätseignismonitor gesendet wird:

```
CREATE WORK ACTION SET WAS1 FOR DATABASE USING WORK CLASS SET WCS1
(WORK ACTION WA1 ON WORK CLASS SMALLDML WHEN ACTIVITYTOTALTIME > 15 MINUTES
COLLECT ACTIVITY DATA WITH DETAILS CONTINUE)
```

Zuletzt erstellen und aktivieren Sie einen Schwellenwertverstoß-Ereignismonitor und einen Aktivitätseignismonitor:

```
CREATE EVENT MONITOR THVIOLATIONS FOR THRESHOLD VIOLATIONS WRITE TO TABLE
SET EVENT MONITOR THVIOLATIONS STATE 1
```

```
CREATE EVENT MONITOR DB2ACTIVITIES FOR ACTIVITIES WRITE TO TABLE
SET EVENT MONITOR DB2ACTIVITIES STATE 1
```

Wenn jetzt eine DML-Aktivität mit einem geschätzten Aufwand von weniger als 500 Timerons länger als 15 Minuten ausgeführt wird, wird ein Schwellenwertverstoßeintrag in den THVIOLATIONS-Ereignismonitor geschrieben (um anzugeben, dass gegen den Gesamtzeitschwellenwert verstoßen wurde), und Details zur DML-Aktivität werden beim Abschluss der Aktivität erfasst und an den DB2ACTIVITIES-Ereignismonitor gesendet. Mit den im DB2ACTIVITIES-Ereignismonitor erfassten Aktivitätsinformationen können Sie die weitere Untersuchung vornehmen. Sie können z. B. die Anweisung EXPLAIN auf der Abfrage ausführen und den Zugriffsplan untersuchen.

Sie sollten auch die Systembelastung und die Warteschlangensteuerung zum Zeitpunkt der Aktivitätssammlung berücksichtigen, da eine lange Laufzeit ein Ergebnis von unzureichenden Systemressourcen oder des Einstellens der Aktivität in die Warteschlange sein kann. Eine lange Laufzeit weist nicht notwendigerweise auf veraltete Statistikdaten hin.

Teil 5. Referenz

Kapitel 8. Prozeduren und Tabellenfunktionen

WLM_CANCEL_ACTIVITY - Abbrechen einer Aktivität

Bei dieser Prozedur wird eine bestimmte Aktivität abgebrochen. Wenn der Abbruch erfolgt, wird eine Fehlermeldung an die Anwendung zurückgegeben, die die abgebrochene Aktivität übergeben hat.

Syntax

```
►► WLM_CANCEL_ACTIVITY(—anwendungskennung—, —uow-id—, —aktivitäts-id—) ◀◀
```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

anwendungskennung

Ein Eingabeargument des Typs BIGINT, das die Anwendungskennung angibt, deren Aktivität abgebrochen werden soll. Wird dieses Argument nicht angegeben, wird keine Aktivität gefunden, und die Fehlermeldung SQL4702N wird mit dem SQLSTATE-Wert 5U035 zurückgegeben.

uow-id

Ein Eingabeargument des Typs INTEGER, das die UOW-ID der Aktivität angibt, die abgebrochen werden soll. Wird dieses Argument nicht angegeben, wird keine Aktivität gefunden, und die Fehlermeldung SQL4702N wird mit dem SQLSTATE-Wert 5U035 zurückgegeben.

aktivitäts-id

Ein Eingabeargument des Typs INTEGER, das die Aktivitäts-ID angibt, die die Aktivität innerhalb der UOW (Unit of Work) eindeutig angibt, die abgebrochen werden soll. Wird dieses Argument nicht angegeben, wird keine Aktivität gefunden, und die Fehlermeldung SQL4702N wird mit dem SQLSTATE-Wert 5U035 zurückgegeben.

Berechtigung

Zugriffsrecht EXECUTE für die Prozedur WLM_CANCEL_ACTIVITY.

Beispiel

Ein Administrator kann die Tabellenfunktion WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES verwenden, um die Anwendungskennung, die UOW-ID und die Aktivitäts-ID einer Aktivität zu finden. Wenn Sie eine Aktivität mit der Anwendungskennung 1, UOW-ID 2 und Aktivitäts-ID 3 abbrechen möchten, müssen Sie Folgendes eingeben:

```
CALL WLM_CANCEL_ACTIVITY(1, 2, 3)
```

Hinweise zur Verwendung

- Wenn keine Aktivität gefunden werden kann, wird die Fehlermeldung SQL4702N mit dem SQLSTATE-Wert 5U035 zurückgegeben.

- Wenn die Aktivität nicht abgebrochen werden kann, weil sie sich nicht im korrekten Status befindet (nicht initialisiert), wird die Fehlermeldung SQL4703N (Ursachencode 1) mit dem SQLSTATE-Wert 5U016 zurückgegeben.
- Wenn die Aktivität erfolgreich abgebrochen wird, wird die Fehlermeldung SQL4725N mit dem SQLSTATE-Wert 57014 an die abgebrochene Anwendung zurückgegeben.
- Wenn der Koordinator zum Zeitpunkt des Abbruchs eine Anforderung für eine andere Aktivität verarbeitet oder inaktiv ist, wird die Aktivität in den Status CANCEL_PENDING versetzt. Sie wird erst abgebrochen, wenn der Koordinator die nächste Anforderung für die Aktivität verarbeitet.

WLM_CAPTURE_ACTIVITY_IN_PROGRESS - Erfassen von Aktivitätsinformationen für Aktivitätsereignismonitor

Diese Prozedur hat zur Folge, dass Informationen für eine bestimmte Aktivität zusammengestellt und in den aktiven Aktivitätsereignismonitor geschrieben werden. Wenn sie auf eine Aktivität angewendet wird, die über untergeordnete Aktivitäten verfügt, generiert diese Prozedur rekursiv bis zur untersten Ebene einen Datensatz für jede untergeordnete Aktivität. Diese Informationen werden erfasst und gesendet, sobald diese Prozedur aufgerufen wird. Sie wartet nicht, bis die Ausführung der Aktivität abgeschlossen ist. Der Datensatz der Aktivität im Ereignismonitor wird als Teildatensatz gekennzeichnet.

Syntax

```

▶▶ WLM_CAPTURE_ACTIVITY_IN_PROGRESS (—anwendungskennung—, —————▶
▶ uow-id—, —aktivitäts-id—) —————▶▶

```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

anwendungskennung

Ein Eingabeargument des Typs BIGINT, das die Anwendungskennung angibt, deren Aktivität erfasst werden soll. Wird dieses Argument nicht angegeben, wird keine Aktivität gefunden, und die Fehlermeldung SQL4702N wird mit dem SQLSTATE-Wert 5U035 zurückgegeben.

uow-id

Ein Eingabeargument des Typs INTEGER, das die UOW-ID der Aktivität angibt, die erfasst werden soll. Wird dieses Argument nicht angegeben, wird keine Aktivität gefunden, und die Fehlermeldung SQL4702N wird mit dem SQLSTATE-Wert 5U035 zurückgegeben.

aktivitäts-id

Ein Eingabeargument des Typs INTEGER, das die Aktivitäts-ID angibt, die die Aktivität innerhalb der UOW (Unit of Work) eindeutig angibt, die erfasst werden soll. Wird dieses Argument nicht angegeben, wird keine Aktivität gefunden, und die Fehlermeldung SQL4702N wird mit dem SQLSTATE-Wert 5U035 zurückgegeben.

Berechtigung

Zugriffsrecht EXECUTE für die Prozedur WLM_CAPTURE_ACTIVITY_IN_PROGRESS.

Beispiel

Eine einzelne Prozedur MYSCHEMA.MYSLOWSTP wird möglicherweise langsamer ausgeführt als normal. Ein Benutzer beklagt sich und der Administrator beschließt, die Ursache der Leistungsminderung zu untersuchen. Die Untersuchung während der Ausführung der gespeicherten Prozedur ist möglicherweise nicht praktikabel. Deshalb kann der Administrator die Aktivität der gespeicherten Prozedur und aller darin verschachtelter Aktivitäten erfassen.

Wenn beispielsweise ein Ereignismonitor für DB2-Aktivitäten mit dem Namen DB2ACTIVITIES vorhanden ist und aktiviert wurde, kann der Administrator die Funktion WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES verwenden, um die Anwendungskennung, die UOW-ID und die Aktivitäts-ID für den Aufruf dieser gespeicherten Prozedur zu verwenden. Wenn die Aktivität mit einer Anwendungskennung von 1, eine UOW-ID von 2 und eine Aktivitäts-ID von 3 angegeben wird, kann der Administrator wie folgt einen Aufruf für WLM_CAPTURE_ACTIVITY_IN_PROGRESS absetzen:

```
CALL WLM_CAPTURE_ACTIVITY_IN_PROGRESS(1,2,3)
```

Sobald die Prozedur abgeschlossen ist, kann der Administrator für einen Aktivitätsereignismonitor mit dem Namen DB2ACTIVITIES die folgende Tabellenfunktion verwenden, um zu ermitteln, an welcher Position die Aktivität die Zeit verbracht hat:

```
CREATE FUNCTION SHOWCAPTUREDACTIVITY(APPHNDL BIGINT,  
                                     UOWID INTEGER,  
                                     ACTIVITYID INTEGER)  
RETURNS TABLE (UOW_ID INTEGER, ACTIVITY_ID INTEGER, STMT_TEXT VARCHAR(40),  
                LIFE_TIME DOUBLE)  
LANGUAGE SQL  
READS SQL DATA  
NO EXTERNAL ACTION  
DETERMINISTIC  
RETURN WITH RAH (LEVEL, APPL_ID, PARENT_UOW_ID, PARENT_ACTIVITY_ID,  
                 UOW_ID, ACTIVITY_ID, STMT_TEXT, ACT_EXEC_TIME) AS  
(SELECT 1, ROOT.APPL_ID, ROOT.PARENT_UOW_ID,  
        ROOT.PARENT_ACTIVITY_ID, ROOT.UOW_ID, ROOT.ACTIVITY_ID,  
        ROOTSTMT.STMT_TEXT, ACT_EXEC_TIME  
 FROM ACTIVITY_DB2ACTIVITIES ROOT, ACTIVITYSTMT_DB2ACTIVITIES ROOTSTMT  
 WHERE ROOT.APPL_ID = ROOTSTMT.APPL_ID AND ROOT.AGENT_ID = APPHNDL  
       AND ROOT.UOW_ID = ROOTSTMT.UOW_ID AND ROOT.UOW_ID = UOWID  
       AND ROOT.ACTIVITY_ID = ROOTSTMT.ACTIVITY_ID AND ROOT.ACTIVITY_ID = ACTIVITYID  
 UNION ALL  
 SELECT PARENT.LEVEL +1, CHILD.APPL_ID, CHILD.PARENT_UOW_ID,  
        CHILD.PARENT_ACTIVITY_ID, CHILD.UOW_ID,  
        CHILD.ACTIVITY_ID, CHILDSTMT.STMT_TEXT, CHILD.ACT_EXEC_TIME  
 FROM RAH PARENT, ACTIVITY_DB2ACTIVITIES CHILD,  
        ACTIVITYSTMT_DB2ACTIVITIES CHILDSTMT  
 WHERE PARENT.APPL_ID = CHILD.APPL_ID AND  
        CHILD.APPL_ID = CHILDSTMT.APPL_ID AND  
        PARENT.UOW_ID = CHILD.PARENT_UOW_ID AND  
        CHILD.UOW_ID = CHILDSTMT.UOW_ID AND  
        PARENT.ACTIVITY_ID = CHILD.PARENT_ACTIVITY_ID AND  
        CHILD.ACTIVITY_ID = CHILDSTMT.ACTIVITY_ID AND  
        PARENT.LEVEL < 64  
 )  
SELECT UOW_ID, ACTIVITY_ID, SUBSTR(STMT_TEXT,1,40),
```

```

ACT_EXEC_TIME AS
LIFE_TIME
FROM RAH
ORDER BY UOW_ID, ACTIVITY_ID;

```

Ein Beispiel für eine Abfrage zum Verwenden der Tabellenfunktion ist:

```

SELECT * FROM TABLE(SHOWCAPTUREDACTIVITY(1, 2, 3))
AS ACTS ORDER BY UOW_ID, ACTIVITY_ID

```

Hinweise

Wenn kein aktiver Aktivitätsereignismonitor vorhanden ist, wird die Fehler-
nachricht SQL1633W mit dem SQLSTATE-Wert 01H53 zurückgegeben.

Wenn Sie diese Prozedur verwenden, um Aktivitätsinformationen für eine Proze-
dur mit INOUT-Parametern zu erfassen, könnten die INOUT-Werte bis zum Zeit-
punkt der Erfassung überschrieben worden sein. Diese Situation tritt nicht auf,
wenn Sie die Workload, die Serviceklasse, die Arbeitsaktion oder den prädiktiven
Schwellenwert, für die/den Sie Aktivitätsdaten erfassen, mit der Angabe von COL-
LECT ACTIVITY WITH DETAILS AND VALUES erstellt haben, oder wenn Sie das
Objekt ändern und das Schlüsselwort COLLECT ACTIVITY DATA mit der Klausel
ON COORDINATOR oder ON ALL sowie mit der Klausel WITH DETAILS AND
VALUES angeben.

WLM_COLLECT_STATS - Erfassen und Zurücksetzen von Workload- Management-Statistiken

Diese Prozedur hat zur Folge, dass Statistikdaten für Serviceklassen, Workloads,
Arbeitsklassen und Schwellenwertwarteschlangen im Statistikereignismonitor
zusammengestellt und geschrieben werden. Die Statistikdaten für Serviceklassen,
Workloads, Arbeitsklassen und Schwellenwertwarteschlangen werden außerdem
zurückgesetzt. Wenn kein aktiver Statistikereignismonitor vorhanden ist, werden
die Statistikdaten nur zurückgesetzt.

Syntax

```

▶▶—WLM_COLLECT_STATS—(—)—————▶▶

```

Das Schema ist SYSPROC.

Berechtigung

Zugriffsrecht EXECUTE für die Prozedur WLM_COLLECT_STATS.

Beispiele

Beispiel 1: WLM_COLLECT_STATS zum Erfassen und Zurücksetzen von Statistiken
aufrufen.

```

CALL WLM_COLLECT_STATS()

```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

```

Rückgabestatus = 0

```

Beispiel 2: WLM_COLLECT_STATS zum Erfassen und Zurücksetzen von Statistiken
aufrufen, während ein anderer Aufruf in Bearbeitung ist.

```
CALL WLM_COLLECT_STATS()
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

```
SQL1632W Die Anforderung zum Erfassen und Zurücksetzen von Statistiken wurde
ignoriert, da eine andere Anforderung zum Erfassen und Zurücksetzen von
Statistiken bereits in Bearbeitung ist.
```

Hinweise zur Verwendung

Die Prozedur `WLM_COLLECT_STATS` wird verwendet, um Statistikdaten manuell zu erfassen. Es werden dieselben Operationen zum Erfassen (Senden der Statistiken an den aktiven Statistikereignismonitor) und Zurücksetzen ausgeführt, die in dem mit dem Datenbankkonfigurationsparameter `WLM_COLLECT_INT` definierten Intervall automatisch auftreten. Wenn die Prozedur zu einem Zeitpunkt aufgerufen wird, an dem eine andere Anforderung zum Erfassen und Zurücksetzen bereits in Bearbeitung ist (z. B. wenn die Prozedur zu einem Zeitpunkt aufgerufen wird, an dem ein anderer Aufruf der Prozedur oder eine automatische Erfassung ausgeführt wird), wird eine Warnung (SQL1632W mit dem `SQLSTATE`-Wert 01H53) zurückgegeben, und die Anforderung wird ignoriert.

Die Prozedur `WLM_COLLECT_STATS` startet nur den Erfassungs- und Zurücksetzungsprozess. Die Prozedur wird möglicherweise an den Aufrufenden zurückgegeben, bevor der Prozess abgeschlossen wurde, d. h. bevor alle Statistiken in den aktiven Statistikereignismonitor geschrieben wurden. Je nach dem, wie schnell die Erfassung und Zurücksetzung der Statistikdaten erfolgt, wird der Aufruf der Prozedur `WLM_COLLECT_STATS` (welche selbst eine Aktivität ist und in der Aktivitätsstatistik protokolliert wird) entweder für das vorherige Erfassungsintervall oder für das gerade gestartete neue Erfassungsintervall protokolliert.

WLM_GET_ACTIVITY_DETAILS - Zurückgeben detaillierter Informationen zu einer bestimmten Aktivität

Diese Funktion gibt detaillierte Informationen zu einer bestimmten Aktivität zurück, die durch ihre Anwendungskennung, UOW-ID und Aktivitäts-ID angegeben wird.

Syntax

```
►►—WLM_GET_ACTIVITY_DETAILS—(—anwendungskennung—,—uow-id—,—
►—aktivitäts-id—,—dbpartitionsnummer—)—►►
```

Das Schema ist `SYSPROC`.

Tabellenfunktionsparameter

anwendungskennung

Ein Eingabeargument des Typs `BIGINT`, das eine gültige Anwendungskennung angibt. Wird das Argument ausgelassen, gibt diese Funktion keine Zeilen zurück. Wird das Argument ausgelassen, wird der Fehler `SQL171N` zurückgegeben.

uow-id

Ein Eingabeargument des Typs `INTEGER`, das eine gültige UOW-ID angibt, die

innerhalb der Anwendung eindeutig ist. Wird das Argument ausgelassen, gibt diese Funktion keine Zeilen zurück. Wird das Argument ausgelassen, wird der Fehler SQL171N zurückgegeben.

aktivitäts-id

Ein Eingabeargument des Typs INTEGER, das eine gültige Aktivitäts-ID angibt, die innerhalb der UOW (Unit of Work) eindeutig ist. Wird das Argument ausgelassen, gibt diese Funktion keine Zeilen zurück. Wird das Argument ausgelassen, wird der Fehler SQL171N zurückgegeben.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe eines Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_ACTIVITY_DETAILS.

Beispiel

Detaillierte Informationen zu einer einzelnen Aktivität können Sie mit Hilfe der Tabellenfunktion WLM_GET_ACTIVITY_DETAILS erhalten. Diese Tabellenfunktion gibt Aktivitätsinformationen wie Name/Wert-Paare für jede Partition zurück. Dieses Beispiel ist so beschränkt, dass nur eine aus elf Elementen bestehende Untergruppe der Name/Wert-Paare für jede Partition für eine Aktivität gezeigt wird, die durch eine Anwendungskennung von 1, eine UOW-ID von 1 und eine Aktivitäts-ID von 5 angegeben wird. Eine vollständige Liste der Name/Wert-Paare finden Sie in Tabelle 46 auf Seite 211 und Tabelle 47 auf Seite 214.

```
SELECT SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(NAME, 1, 20) AS NAME,
       SUBSTR(VALUE, 1, 30) AS VALUE
FROM TABLE(WLM_GET_ACTIVITY_DETAILS(1, 1, 5, -2)) AS ACTDETAIL
WHERE NAME IN ('APPLICATION_HANDLE',
              'COORD_PARTITION_NUM',
              'LOCAL_START_TIME',
              'UOW_ID',
              'ACTIVITY_ID',
              'PARENT_UOW_ID',
              'PARENT_ACTIVITY_ID',
              'ACTIVITY_TYPE',
              'NESTING_LEVEL',
              'INVOCATION_ID',
              'ROUTINE_ID')
ORDER BY PART
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

PART	NAME	VALUE
0	APPLICATION_HANDLE	1
0	COORD_PARTITION_NUM	0
0	LOCAL_START_TIME	2005-11-25-18.52.49.343000
0	UOW_ID	1
0	ACTIVITY_ID	5
0	PARENT_UOW_ID	1
0	PARENT_ACTIVITY_ID	3
0	ACTIVITY_TYPE	READ_DML
0	NESTING_LEVEL	0
0	INVOCATION_ID	1

```

0  ROUTINE_ID          0
1  APPLICATION_HANDLE  1
1  COORD_PARTITION_NUM 0
1  LOCAL_START_TIME   2005-11-25-18.52.49.598000
1  UOW_ID              1
1  ACTIVITY_ID         5
1  PARENT_UOW_ID       1
1  PARENT_ACTIVITY_ID  1
1  ACTIVITY_TYPE       READ_DML
1  NESTING_LEVEL       0
1  INVOCATION_ID       1
1  ROUTINE_ID          0

```

Hinweise

Der ACTIVITY_STATE-Wert QUEUED bedeutet, dass die Koordinatoraktivität versucht, die Schwellenwerttickets über einen Fernprozeduraufruf abzurufen, aber bisher keine Antwort erhalten hat. Wenn dieser Status angezeigt wird, kann dies bedeuten, dass die Aktivität von WLM in die Warteschlange gestellt wurde. Wird er nur kurzzeitig angezeigt, kann dies darauf hindeuten, dass die Aktivität gerade ihre Tickets abrufen. Wenn Sie sich ein genaueres Bild darüber machen wollen, ob die Aktivität tatsächlich in die Warteschlange gestellt wird, können Sie ermitteln, welcher Agent die Aktivität verarbeitet (mit Hilfe der Tabellenfunktion WLM_GET_SERVICE_CLASS_AGENTS) und feststellen, ob das Ereignisobjekt dieses Agenten in der Katalogpartition den Wert WLM_QUEUE aufweist.

Zurückgegebene Informationen

Tabelle 45. Für WLM_GET_ACTIVITY_DETAILS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
NAME	VARCHAR(256)	Elementname. Mögliche Werte finden Sie in Tabelle 46 und Tabelle 47 auf Seite 214.
VALUE	VARCHAR(1024)	Elementwerte. Mögliche Werte finden Sie in Tabelle 46 und Tabelle 47 auf Seite 214.

Tabelle 46. Zurückgegebene Elemente

Elementname	Beschreibung
APPLICATION_HANDLE	Eine systemweit eindeutige Kennung für die Anwendung. In einer Datenbank mit einer Partition besteht diese Kennung aus einem 16-Bit-Zähler. In einer Datenbank mit mehreren Partitionen besteht diese Kennung aus der Nummer der koordinierenden Partition, die mit einem 16-Bit-Zähler verknüpft ist. Außerdem ist diese Kennung auf jeder Partition gleich, auf der die Anwendung unter Umständen eine sekundäre Verbindung herstellt.
COORD_PARTITION_NUM	Die Koordinatorpartition der Aktivität.
UOW_ID	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der diese Aktivität ursprünglich gestartet wurde.
ACTIVITY_ID	Eindeutige Kennung (ID) der Aktivität innerhalb einer Anwendung.

Tabelle 46. Zurückgegebene Elemente (Forts.)

Elementname	Beschreibung
PARENT_UOW_ID	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der die übergeordnete Aktivität dieser Aktivität ursprünglich gestartet wurde. Gibt eine leere Zeichenfolge zurück, wenn die Aktivität keine übergeordnete Aktivität aufweist oder auf einer fernen Partition ausgeführt wird.
PARENT_ACTIVITY_ID	Eindeutige Aktivitätenkennung (ID) innerhalb einer UOW für die übergeordnete Aktivität der Aktivität mit der ID <code>ACTIVITY_ID</code> . Gibt eine leere Zeichenfolge zurück, wenn die Aktivität keine übergeordnete Aktivität aufweist.
ACTIVITY_STATE	Mögliche Werte: <ul style="list-style-type: none"> • <code>CANCEL_PENDING</code> • <code>EXECUTING</code> • <code>IDLE</code> • <code>INITIALIZING</code> • <code>QP_CANCEL_PENDING</code> • <code>QP_QUEUED</code> • <code>QUEUED</code> • <code>TERMINATING</code> • <code>UNKNOWN</code>
ACTIVITY_TYPE	Mögliche Werte: <ul style="list-style-type: none"> • <code>CALL</code> • <code>DDL</code> • <code>LOAD</code> • <code>OTHER</code> • <code>READ_DML</code> • <code>WRITE_DML</code>
NESTING_LEVEL	Die Verschachtelungsebene dieser Aktivität. Die Verschachtelungsebene gibt an, wie tief diese Aktivität in seiner obersten übergeordneten Aktivität verschachtelt ist.
INVOCATION_ID	Unterscheidet einen bestimmten Aufruf dieser Aktivität von anderen Aufrufen dieser Aktivität in derselben Verschachtelungsebene.
ROUTINE_ID	Eindeutige Routinenkennung.
UTILITY_ID	Handelt es sich bei der Aktivität um ein Dienstprogramm, ist dies die Kennung (ID) des Dienstprogramms. Andernfalls hat dieses Feld den Wert '0'.
SERVICE_CLASS_ID	Eindeutige Kennung (ID) der Serviceklasse, zu der diese Aktivität gehört.
DATABASE_WORK_ACTION_SET_ID	Wurde diese Aktivität einem Arbeitsaktionsset zugeordnet, das auf die Datenbank angewendet wurde, enthält diese Spalte die Kennung (ID) des Arbeitsaktionssets. Diese Spalte enthält den Wert '0', wenn die Aktivität nicht einem Arbeitsaktionsset zugeordnet wurde, das auf die Datenbank angewendet wurde.

Tabelle 46. Zurückgegebene Elemente (Forts.)

Elementname	Beschreibung
DATABASE_WORK_CLASS_ID	Wurde diese Aktivität einem Arbeitsaktionsset zugeordnet, das auf die Datenbank angewendet wurde, enthält diese Spalte die Kennung (ID) der Arbeitsklasse dieser Aktivität. Diese Spalte enthält den Wert '0', wenn die Aktivität nicht einem Arbeitsaktionsset zugeordnet wurde, das auf die Datenbank angewendet wurde.
SERVICE_CLASS_WORK_ACTION_SET_ID	Wurde diese Aktivität einem Arbeitsaktionsset zugeordnet, das auf eine Serviceklasse angewendet wurde, enthält diese Spalte die Kennung (ID) des Arbeitsaktionssets. Diese Spalte enthält den Wert '0', wenn die Aktivität nicht einem Arbeitsaktionsset zugeordnet wurde, das auf eine Serviceklasse angewendet wurde.
SERVICE_CLASS_WORK_CLASS_ID	Wurde diese Aktivität einem Arbeitsaktionsset zugeordnet, das auf eine Serviceklasse angewendet wurde, enthält diese Spalte die Kennung (ID) der Arbeitsklasse dieser Aktivität. Diese Spalte enthält den Wert '0', wenn die Aktivität nicht einem Arbeitsaktionsset zugeordnet wurde, das auf eine Serviceklasse angewendet wurde.
ENTRY_TIME	Der Zeitpunkt, an dem diese Aktivität im System eingegangen ist.
LOCAL_START_TIME	Der Zeitpunkt des Starts dieser Aktivität in der Partition. Die Angabe erfolgt in Ortszeit. Das Feld kann leer sein, wenn eine Aktivität im System initiiert wurde, diese Aktivität sich aber in einer Warteschlange befindet und die Ausführung der Aktivität noch nicht gestartet wurde.
LAST_REFERENCE_TIME	Jedes Mal, wenn eine Anforderung in dieser Aktivität auftritt, wird dieses Feld aktualisiert.
PACKAGE_NAME	Wenn die Aktivität eine SQL-Anweisung ist, stellt dies den Namen ihres Pakets dar.
PACKAGE_SCHEMA	Wenn die Aktivität eine SQL-Anweisung ist, stellt dies den Schemanamen ihres Pakets dar.
PACKAGE_VERSION_ID	Wenn die Aktivität eine SQL-Anweisung ist, stellt dies die Version ihres Pakets dar.
SECTION_NUMBER	Wenn die Aktivität eine SQL-Anweisung ist, stellt dies ihre Abschnittsnummer dar.
STMT_PKG_CACHE_ID	Kennung (ID) des Anweisungspaketcaches.
STMT_TEXT	Wenn die Aktivität dynamisches SQL aufweist, oder wenn sie statisches SQL aufweist, für das der Anweisungstext verfügbar ist, enthält dieses Feld die ersten 1024 Zeichen des Anweisungstexts. Anderenfalls enthält es eine leere Zeichenfolge.
EFFECTIVE_ISOLATION	Die effektive Isolationsstufe für diese Aktivität.
EFFECTIVE_LOCK_TIMEOUT	Der effektive Wert für die Überschreitung der Sperrzeit für diese Aktivität.
EFFECTIVE_QUERY_DEGREE	Der effektive Wert des Abfragegrads für diese Aktivität.
QUERY_COST_ESTIMATE	Vom SQL-Compiler geschätzter Aufwand (in Timeron) für eine Abfrage.

Tabelle 46. Zurückgegebene Elemente (Forts.)

Elementname	Beschreibung
ROWS_FETCHED	Dies ist die Anzahl der aus der Tabelle gelesenen Zeilen. Es werden nur die Werte für die Datenbankpartition zurückgemeldet, für die dieser Datensatz aufgezeichnet wird. Bei DPF-Systemen (Database Partitioning Feature) spiegeln diese Werte möglicherweise nicht die korrekten Summen für die gesamte Aktivität wider. Ist der Anweisungsmonitorschalter nicht aktiviert, werden für dieses Element keine Daten erfasst, und es wird stattdessen der Wert -1 angezeigt.
ROWS_MODIFIED	Dies ist die Anzahl der eingefügten, aktualisierten und gelöschten Zeilen. Es werden nur die Werte für die Datenbankpartition zurückgemeldet, für die dieser Datensatz aufgezeichnet wird. Bei DPF-Systemen (Database Partitioning Feature) spiegeln diese Werte möglicherweise nicht die korrekten Summen für die gesamte Aktivität wider. Ist der Anweisungsmonitorschalter nicht aktiviert, werden für dieses Element keine Daten erfasst, und es wird stattdessen der Wert -1 angezeigt.
SYSTEM_CPU_TIME	Die System-CPU-Zeit insgesamt (in Sekunden und Mikrosekunden), die vom Agentenprozess des Datenbankmanagers, der UOW (Unit of Work) oder der Anweisung verwendet wurde. Ist der Anweisungsmonitorschalter oder der Zeitmarkenschalter nicht aktiviert, werden für dieses Element keine Daten erfasst, und es wird stattdessen der Wert -1 angezeigt.
USER_CPU_TIME	Die Benutzer-CPU-Zeit insgesamt (in Sekunden und Mikrosekunden), die vom Agentenprozess des Datenbankmanagers, der UOW (Unit of Work) oder der Anweisung verwendet wurde. Ist der Anweisungsmonitorschalter oder der Zeitmarkenschalter nicht aktiviert, werden für dieses Element keine Daten erfasst, und es wird stattdessen der Wert -1 angezeigt.
QP_QUERY_ID	Die Abfrage-ID, die dieser Aktivität vom Query Patroller zugeordnet wird, wenn es sich bei der Aktivität um eine Abfrage handelt. Eine Abfrage-ID von 0 gibt an, dass Query Patroller dieser Aktivität keine Abfrage-ID zugeordnet hat.

Die folgenden Elemente werden nur zurückgegeben, wenn die entsprechenden Schwellenwerte für die Aktivität gelten.

Tabelle 47. Zurückgegebene Elemente (sofern zutreffend)

Elementname	Beschreibung
CONCURRENTWORKLOADACTIVITIES_THRESHOLD_ID	Die ID des Schwellenwerts.
CONCURRENTWORKLOADACTIVITIES_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
CONCURRENTWORKLOADACTIVITIES_THRESHOLD_VIOLATED	'Yes' gibt an, dass diese Aktivität gegen den Schwellenwert verstoßen hat. 'No' gibt an, dass diese Aktivität nicht gegen den Schwellenwert verstoßen hat.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_ID	Die ID des Schwellenwerts.

Tabelle 47. Zurückgegebene Elemente (sofern zutreffend) (Forts.)

Elementname	Beschreibung
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_QUEUED	Gibt an, ob die Aktivität durch diesen Schwellenwert in eine Warteschlange eingereiht wurde.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_ID	Die ID des Schwellenwerts.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_QUEUED	'Yes' gibt an, dass die Aktivität durch diesen Schwellenwert in eine Warteschlange eingereiht wurde. 'No' gibt an, dass die Aktivität nicht in eine Warteschlange eingereiht wurde.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_ID	Die ID des Schwellenwerts.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_QUEUED	'Yes' gibt an, dass die Aktivität durch diesen Schwellenwert in eine Warteschlange eingereiht wurde. 'No' gibt an, dass die Aktivität nicht in eine Warteschlange eingereiht wurde.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_ID	Die ID des Schwellenwerts.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_QUEUED	Gibt an, ob die Aktivität durch diesen Schwellenwert in eine Warteschlange eingereiht wurde.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
ESTIMATEDSQLCOST_THRESHOLD_ID	Die ID des Schwellenwerts.
ESTIMATEDSQLCOST_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.

Tabelle 47. Zurückgegebene Elemente (sofern zutreffend) (Forts.)

Elementname	Beschreibung
ESTIMATEDSQLCOST_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
SQLTEMPSPACE_THRESHOLD_ID	Die ID des Schwellenwerts.
SQLTEMPSPACE_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
SQLTEMPSPACE_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
SQLROWSRETURNED_THRESHOLD_ID	Die ID des Schwellenwerts.
SQLROWSRETURNED_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
SQLROWSRETURNED_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.
ACTIVITYTOTALTIME_THRESHOLD_ID	Die ID des Schwellenwerts.
ACTIVITYTOTALTIME_THRESHOLD_VALUE	Der Wert, bei dessen Überschreitung der Schwellenwert ausgelöst wird.
ACTIVITYTOTALTIME_THRESHOLD_VIOLATED	'Yes' gibt an, dass gegen den Schwellenwert verstoßen wurde. 'No' gibt an, dass bisher nicht gegen den Schwellenwert verstoßen wurde.

WLM_GET_QUEUE_STATS - Zurückgeben der Schwellenwertwarteschlangenstatistik

Diese Funktion gibt eine Basisstatistik für mindestens eine Schwellenwertwarteschlange zurück.

Diese Funktion gibt eine Statistikzeile für jede Schwellenwertwarteschlange zurück. Statistikdaten werden für Warteschlangen in allen aktiven Partitionen zurückgegeben.

Syntax

```

▶▶—WLM_GET_QUEUE_STATS—(—schwellenwertvergleichselement—,—schwellenwertdomäne—,——————▶
▶—schwellenwertname—,—schwellenwert-id—)—————▶▶▶

```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

schwellenwertvergleichselement

Ein Eingabeargument des Typs VARCHAR(27), das ein gültiges Schwellenwertvergleichselement angibt. Die möglichen Werte sind:

- CONCDDBC: Schwellenwert für gleichzeitig ablaufende Datenbankkoordinatoraktivitäten
- DBCONN: Schwellenwert für die Gesamtzahl der Verbindungen zu Datenbankpartitionen
- SCCONN: Schwellenwert für die Gesamtzahl der Verbindungen zu Serviceklassenpartitionen
- NULL oder eine leere Zeichenfolge: Es werden Daten für alle möglichen Schwellenwertvergleichselemente zurückgegeben. Die Werte für *schwollenwertvergleichselement* entsprechen denen der Spalte THRESHOLDPREDICATE in der Sicht SYSCAT.THRESHOLDS.

schwollenwertdomäne

Ein Eingabeargument des Typs VARCHAR(18), das eine gültige Schwellenwertdomäne angibt. Die möglichen Werte sind:

- DB: Datenbank
- SB: Serviceunterklasse
- SP: Servicesuperklasse
- WA: Arbeitsaktionsset
- NULL oder eine leere Zeichenfolge: Es werden Daten für alle möglichen Schwellenwertdomänen zurückgegeben. Die Werte für *schwollenwertdomäne* entsprechen denen der Spalte DOMAIN in der Sicht SYSCAT.THRESHOLDS.

schwollenwertname

Ein Eingabeargument des Typs VARCHAR(128), das einen gültigen Schwellenwertnamen angibt. Ist das Argument NULL oder eine leere Zeichenfolge, werden Daten für alle Schwellenwerte zurückgegeben, die die anderen Kriterien erfüllen. Die Werte für *schwollenwertname* entsprechen denen der Spalte THRESHOLDNAME in der Sicht SYSCAT.THRESHOLDS.

schwollenwert-id

Ein Eingabeargument des Typs INTEGER(18), das eine gültige Schwellenwert-ID angibt. Ist das Argument NULL oder -1, werden Daten für alle Schwellenwerte zurückgegeben, die die anderen Kriterien erfüllen. Die Werte für *schwollenwert-id* entsprechen denen der Spalte THRESHOLDID in der Sicht SYSCAT.THRESHOLDS.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_QUEUE_STATS.

Beispiel

Zum Anzeigen aller Basisstatistikdaten für alle Warteschlangen auf dem System, über alle Partitionen hinweg:

```
SELECT substr(THRESHOLD_NAME, 1, 6) THRESHNAME,
       THRESHOLD_PREDICATE,
       THRESHOLD_DOMAIN,
       DBPARTITIONNUM PART,
       QUEUE_SIZE_TOP,
       QUEUE_TIME_TOTAL,
       QUEUE_ASSIGNMENTS_TOTAL QUEUE_ASSIGN
FROM table(WLM_GET_QUEUE_STATS('',' ', '-1')) as QSTATS
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

THRESHNAME	THRESHOLD_PREDICATE	THRESHOLD_DOMAIN	...
LIMIT1	CONCDBC	DB	...
LIMIT2	SCCONN	SP	...
LIMIT3	DBCONN	DB	...

Ausgabe zu der Abfrage (Forts.)

...	PART	QUEUE_SIZE_TOP	QUEUE_TIME_TOTAL	QUEUE_ASSIGN
...	0	12	1238540	734
...	0	4	741249	24
...	0	7	412785	128

Hinweise

Es wird keine Spaltenberechnung über mehrere Warteschlangen (in einer Partition) oder über mehrere Partitionen (für eine oder mehrere Warteschlangen) ausgeführt. Dieser Typ der Spaltenberechnung kann jedoch erreicht werden, indem wie im Beispiel oben gezeigt SQL-Abfragen verwendet werden.

Zurückgegebene Informationen

Tabelle 48. Für WLM_GET_QUEUE_STATS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
THRESHOLD_PREDICATE	VARCHAR(27)	<p>Schwellenwertvergleichselement des für diese Warteschlange verantwortlichen Schwellenwerts. Folgende Werte sind möglich:</p> <ul style="list-style-type: none"> • CONCDBC: Schwellenwert für gleichzeitig ablaufende Datenbankkoordinatoraktivitäten • DBCONN: Schwellenwert für die Gesamtzahl der Verbindungen zu Datenbankpartitionen • SCCONN: Schwellenwert für die Gesamtzahl der Verbindungen zu Serviceklassenpartitionen <p>Die Werte für das Schwellenwertvergleichselement entsprechen denen der Spalte THRESHOLDPREDICATE in der Sicht SYSCAT.THRESHOLDS.</p>
THRESHOLD_DOMAIN	VARCHAR(18)	<p>Die Domäne des für diese Warteschlange verantwortlichen Schwellenwerts. Die möglichen Werte sind:</p> <ul style="list-style-type: none"> • DB: Datenbank • SB: Serviceunterklasse • SP: Servicesuperklasse • WA: Arbeitsaktionsset <p>Die Werte für die Schwellenwertdomäne entsprechen denen der Spalte DOMAIN in der Sicht SYSCAT.THRESHOLDS.</p>

Tabelle 48. Für WLM_GET_QUEUE_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
THRESHOLD_NAME	VARCHAR(128)	Der eindeutige Name des für diese Warteschlange verantwortlichen Schwellenwerts. Der Wert für den Schwellenwertnamen entspricht dem Wert der Spalte THRESHOLDNAME in der Sicht SYSCAT.THRESHOLDS.
THRESHOLD_ID	INTEGER	Die eindeutige ID des für diese Warteschlange verantwortlichen Schwellenwerts. Der Wert für die Schwellenwert-ID entspricht dem Wert der Spalte THRESHOLDID in der Sicht SYSCAT.THRESHOLDS.
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
SERVICE_SUPERCLASS_NAME	VARCHAR(128)	Name der Servicesuperklasse, die die Domäne für den Schwellenwert darstellt, der für diese Warteschlange verantwortlich ist. Leer, wenn die Schwellenwertdomäne keine Servicesuperklasse oder Serviceunterklasse ist.
SERVICE_SUBCLASS_NAME	VARCHAR(128)	Name der Serviceunterklasse, die die Domäne für den Schwellenwert darstellt, der für diese Warteschlange verantwortlich ist. Leer, wenn die Schwellenwertdomäne keine Serviceunterklasse ist.
WORK_ACTION_SET_NAME	VARCHAR(128)	Name des Arbeitsaktionssets, das die Domäne für den Schwellenwert darstellt, der für diese Warteschlange verantwortlich ist. Leer, wenn die Schwellenwertdomäne kein Arbeitsaktionsset ist.
WORK_CLASS_NAME	VARCHAR(128)	Name der Arbeitsklasse, deren Arbeitsaktion zu einem Arbeitsaktionsset gehört, das die Domäne für den Schwellenwert darstellt, der für diese Warteschlange verantwortlich ist. Leer, wenn die Schwellenwertdomäne kein Arbeitsaktionsset ist.
WORKLOAD_NAME	VARCHAR(128)	Name der Workload, die die Domäne für den Schwellenwert darstellt, der für diese Warteschlange verantwortlich ist. Leer, wenn die Schwellenwertdomäne keine Workload ist.

Tabelle 48. Für WLM_GET_QUEUE_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
LAST_RESET	TIMESTAMP	<p>Zeitpunkt der letzten Zurücksetzung der Statistikdaten. Es gibt vier Ereignisse, deren Auftreten das Zurücksetzen der Statistikdaten auslöst, wodurch diese Zeitmarke aktualisiert wird:</p> <ul style="list-style-type: none"> • Die Prozedur WLM_COLLECT_STATS wird aufgerufen. • Der Prozess der regelmäßigen Erfassung und Zurücksetzung, der vom Datenbankkonfigurationsparameter WLM_COLLECT_INT gesteuert wird, veranlasst eine Erfassung und Zurücksetzung. • Die Datenbank wird reaktiviert. • Der Schwellenwert, für den Warteschlangenstatistikdaten dokumentiert werden, wurde modifiziert, und die Änderung wurde festgeschrieben. <p>Die Zeitmarke LAST_RESET wird in Ortszeit angegeben.</p>
QUEUE_SIZE_TOP	INTEGER	Maximale Anzahl der Verbindungen bzw. Aktivitäten in der Warteschlange, die seit dem letzten Zurücksetzen erreicht wurde.
QUEUE_TIME_TOTAL	BIGINT	Die Zeit, die alle Verbindungen bzw. Aktivitäten, die seit dem letzten Zurücksetzen in die Warteschlange gestellt wurden, insgesamt in der Warteschlange verbracht haben. Die Zeit wird in Millisekunden gemessen.
QUEUE_ASSIGNMENTS_TOTAL	BIGINT	Anzahl der Verbindungen bzw. Aktivitäten, die der Warteschlange seit dem letzten Zurücksetzen zugeordnet wurden.
QUEUE_SIZE_CURRENT	INTEGER	Anzahl der Verbindungen bzw. Aktivitäten in der Warteschlange.
QUEUE_TIME_LATEST	BIGINT	Zeit, die die letzte Verbindung bzw. Aktivität, die die Warteschlange verlassen hat, in der Warteschlange verbracht hat. Die Zeit wird in Millisekunden angegeben.
QUEUE_EXIT_TIME_LATEST	TIMESTAMP	Zeitpunkt, an dem die letzte Verbindung oder Aktivität die Warteschlange verlassen hat.
THRESHOLD_CURRENT_CONCURRENCY	INTEGER	Anzahl der Verbindungen bzw. Aktivitäten, die derzeit entsprechend des Schwellenwerts ausgeführt werden.

Tabelle 48. Für WLM_GET_QUEUE_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
THRESHOLD_MAX_CONCURRENCY	INTEGER	Maximale Anzahl der Verbindungen bzw. Aktivitäten, die gemäß des Schwellenwerts gleichzeitig ausgeführt werden können.

WLM_GET_SERVICE_CLASS_AGENTS - Auflisten von Agenten, die in einer Serviceklasse ausgeführt werden

Mit dieser Funktion wird die Liste der Agenten, der Prozesse im abgeschirmten Modus (db2fmpls) und der Systementitäten in der angegebenen Partition aufgelistet, die in der angegebenen Serviceklasse oder für die angegebene Anwendung ausgeführt werden. Die Systementitäten sind Threads und Prozesse, die keine Agenten sind, wie Seitenlöschfunktionen und Vorabselektfunktionen.

Syntax

```

▶▶—WLM_GET_SERVICE_CLASS_AGENTS—(—servicesuperklassenname—, —————▶
▶—serviceunterklassenname—, —anwendungskennung—, —dbpartitionsnummer—)————▶

```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

servicesuperklassenname

Ein Eingabeargument des Typs VARCHAR(128), das beim Aufruf dieser Funktion einen gültigen Servicesuperklassenamen in der Datenbank angibt, zu der zurzeit eine Verbindung besteht. Ist das Argument NULL oder eine leere Zeichenfolge, werden Daten für alle Superklassen in der Datenbank abgerufen, für die die übrigen Parameter übereinstimmen.

serviceunterklassenname

Ein Eingabeargument des Typs VARCHAR(128), das auf eine bestimmte Unterklasse in einer Superklasse verweist. Ist das Argument NULL oder eine leere Zeichenfolge, werden Daten für alle Unterklassen in der Datenbank abgerufen, für die die übrigen Parameter übereinstimmen.

anwendungskennung

Ein Eingabeargument des Typs BIGINT, das die Anwendungskennung angibt, für die Agenteninformationen zurückgegeben werden sollten. Ist das Argument NULL, werden Daten für alle Anwendungen in der Datenbank abgerufen, für die die übrigen Parameter übereinstimmen. Mit der Anwendungskennung 0 werden nur die Systementitäten zurückgegeben.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe eines Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_SERVICE_CLASS_AGENTS.

Beispiel

Zurückgeben einer Liste der mit der Anwendungskennung 1 verknüpfte Agenten für alle Datenbankpartitionen. Die Anwendungskennung könnte mit dem Befehl LIST APPLICATIONS oder der Tabellenfunktion WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES ermittelt worden sein.

```
SELECT SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHANDLE,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(AGENT_TID),1,9) AS AGENT_TID,
       SUBSTR(AGENT_TYPE,1,11) AS AGENTTYPE,
       SUBSTR(AGENT_STATE,1,10) AS AGENTSTATE,
       SUBSTR(REQUEST_TYPE,1,12) AS REQTYPE,
       SUBSTR(CHAR(UOW_ID),1,6) AS UOW_ID,
       SUBSTR(CHAR(ACTIVITY_ID),1,6) AS ACT_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS(CAST(NULL AS VARCHAR(128)),
     CAST(NULL AS VARCHAR(128)), 1, -2)) AS SCDETAILS
ORDER BY APPHANDLE, PART, AGENT_TID"
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

APPHANDLE	PART	AGENT_TID	AGENTTYPE	AGENTSTATE	REQTYPE	UOW_ID	ACT_ID
1	0	3	COORDINATOR	ACTIVE	FETCH	1	5
1	0	4	SUBAGENT	ACTIVE	SUBSECTION:1	1	5
1	1	2	SUBAGENT	ACTIVE	SUBSECTION:2	1	5

Die Ausgabe zeigt einen Koordinatoragenten und einen Subagenten auf der Partition 0 sowie einen Subagenten auf der Partition 1, der für eine Aktivität mit einer UOW-ID von 1 und einer Aktivitäts-ID von 5 aktiv ist. Der Koordinatoragent gibt an, dass es sich bei der Anforderung um eine Abrufanforderung handelt.

Hinweise

Die Parameter werden über AND verknüpft. Wenn Sie also Datensätze angeben, die miteinander in Konflikt stehen, beispielsweise eine Servicesuperklasse SUP_A und eine Unterklasse SUB_B, wobei SUB_B keine Unterklasse von SUP_A ist, werden keine Zeilen zurückgegeben.

Zurückgegebene Informationen

Tabelle 49. Von WLM_GET_SERVICE_CLASS_AGENTS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
SERVICE_SUPERCLASS_NAME	VARCHAR(128)	Name der Servicesuperklasse, aus der dieser Datensatz erfasst wurde.
SERVICE_SUBCLASS_NAME	VARCHAR(128)	Name der Serviceunterklasse, aus der dieser Datensatz erfasst wurde.
APPLICATION_HANDLE	BIGINT	Eine systemweit eindeutige Kennung für die Anwendung. In einer Datenbank mit einer Partition besteht diese Kennung aus einem 16-Bit-Zähler. In einer Datenbank mit mehreren Partitionen besteht diese Kennung aus der Nummer der koordinierenden Partition, die mit einem 16-Bit-Zähler verknüpft ist. Außerdem ist diese Kennung auf jeder Partition gleich, auf der die Anwendung unter Umständen eine sekundäre Verbindung herstellt.

Tabelle 49. Von WLM_GET_SERVICE_CLASS_AGENTS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
ENTITY	VARCHAR(32)	Wenn der Typ der Entität in dieser Zeile ein Agent ist, wird in diesem Feld "db2agent" angezeigt. Wenn der Typ der Entität in dieser Zeile ein Prozess im abgeschirmten Modus ist, wird in diesem Feld "db2fmp (pid)" angezeigt, wobei pid die Prozess-ID des Prozesses im abgeschirmten Modus ist. Andernfalls wird der Name der Systementität angezeigt.
WORKLOAD_NAME	VARCHAR(128)	Name der Workload, aus der dieser Datensatz erfasst wurde.
WORKLOAD_OCCURRENCE_ID	INTEGER	Die Kennung (ID) des Workloadvorkommens. Dies ist erst in Kombination mit der Nummer der Datenbankkoordinatorpartition und dem Workloadnamen eine eindeutige Angabe des Workloadvorkommens. Alternativ kann anstelle der Nummer der Datenbankkoordinatorpartition auch die Anwendungskennung verwendet werden.
UOW_ID	INTEGER	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der diese Aktivität ursprünglich gestartet wurde.
ACTIVITY_ID	INTEGER	Eindeutige Aktivitäts-ID innerhalb einer UOW (Unit of Work).
PARENT_UOW_ID	INTEGER	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der die übergeordnete Aktivität dieser Aktivität ursprünglich gestartet wurde. Gibt ein leeres Feld zurück, wenn die Aktivität keine übergeordnete Aktivität aufweist.
PARENT_ACTIVITY_ID	INTEGER	Eindeutige Aktivitäts-ID innerhalb einer UOW (Unit of Work) für die übergeordnete Aktivität mit der ID der Aktivitäts-ID (activity_id). Gibt ein leeres Feld zurück, wenn die Aktivität keine übergeordnete Aktivität aufweist.
AGENT_TID	BIGINT	Thread-ID des Agenten oder der Systementität. Wenn diese ID nicht verfügbar ist, hat dieses Feld den Wert Null.
AGENT_TYPE	VARCHAR(32)	Koordinator oder Subagent. Bei einem Koordinator kann sich die Agenten-ID in Konzentratoren umgebungen ändern. Die Agententypen werden folgendermaßen dargestellt: <ul style="list-style-type: none"> • COORDINATOR • OTHER • PDBSUBAGENT • SMP SUBAGENT
SMP_COORDINATOR	INTEGER	Gibt an, ob der Agent ein SMP-Koordinator ist: 1 für 'Ja' und 0 für 'Nein'.
AGENT_SUBTYPE	VARCHAR(32)	Zu den möglichen Untertypen gehören: <ul style="list-style-type: none"> • DSS • OTHER • RPC • SMP
AGENT_STATE	VARCHAR(32)	Gibt an, ob ein Agent zugeordnet oder aktiv ist. Folgende Werte sind möglich: <ul style="list-style-type: none"> • ACTIVE • ASSOCIATED

Tabelle 49. Von WLM_GET_SERVICE_CLASS_AGENTS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
EVENT_TYPE	VARCHAR(32)	Der Typ des Ereignisses, das von diesem Agenten zuletzt verarbeitet wurde. Folgende Werte sind möglich: <ul style="list-style-type: none"> • ACQUIRE • PROCESS • WAIT
EVENT_OBJECT	VARCHAR(32)	Das Objekt des Ereignisses, das von diesem Agenten zuletzt verarbeitet wurde. Folgende Werte sind möglich: <ul style="list-style-type: none"> • COMPRESSION_DICTIONARY_BUILD • IMPLICIT_REBIND • INDEX_RECREATE • LOCK (Sperr) • LOCK_ESCALATION • QP_QUEUE • REMOTE_REQUEST • REQUEST • ROUTINE • WLM_QUEUE
EVENT_STATE	VARCHAR(32)	Der Status des Ereignisses, das von diesem Agenten zuletzt verarbeitet wurde. Folgende Werte sind möglich: <ul style="list-style-type: none"> • EXECUTING • IDLE
REQUEST_ID	VARCHAR(64)	Eindeutig nur in Kombination mit der Anwendungskennung (application_handle). Kann zur Unterscheidung zwischen einer langandauernden Anforderung und Mehrfachanforderungen verwendet werden. Beispielsweise zur Unterscheidung zwischen mehrfachen Abrufen und einem langandauernden Abruf.

Tabelle 49. Von WLM_GET_SERVICE_CLASS_AGENTS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
REQUEST_TYPE	VARCHAR(32)	<p>Der Typ der Anforderung. Folgende Werte sind möglich:</p> <ul style="list-style-type: none"> • Für Koordinatoragenten: <ul style="list-style-type: none"> – CLOSE – COMMIT – COMPILE – DESCRIBE – EXCSQLSET – EXECIMMD – EXECUTE – FETCH – INTERNAL <zahl> – OPEN – PREPARE – REBIND – REDISTRIBUTE – REORG – ROLLBACK – RUNSTATS • Für Subagenten (DSS und SMP): <ul style="list-style-type: none"> – Zeigt die Anzahl der Unterabschnitte im Format "SUBSECTION:<unterabschnittsanzahl>" an, wenn die Anzahl der Unterabschnitte ungleich Null ist. Andernfalls wird NULL zurückgegeben.

Tabelle 49. Von WLM_GET_SERVICE_CLASS_AGENTS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
REQUEST_TYPE (Fortsetzung)	VARCHAR(32)	<ul style="list-style-type: none"> • Für Subagenten (RPC): <ul style="list-style-type: none"> - ABP - CATALOG - INTERNAL - REORG - RUNSTATS - WLM • Für Subagenten (OTHER): <ul style="list-style-type: none"> - ABP - APP_RBSVPT - APP_RELSVPT - BACKUP - CLOSE - EXTERNAL_RBSVPT - EVMON - FORCE - FORCE_ALL - INTERNAL <zahl> - INTERRUPT - NOOP: wenn es keine Anforderung gibt - QP - REDISTRIBUTE - STMT_RBSVPT - STOP_USING - UPDATE_DBM_CFG - WLM <p>Wenn es sich bei dem Anforderungstyp um einen der internen Typen handelt, wird der Wert als 'INTERNAL' gefolgt vom tatsächlichen Wert der internen Konstante angezeigt.</p>
NESTING_LEVEL	INTEGER	Stellt die Verschachtelungsebene der Aktivität dar, deren ID die Aktivitäts-ID (activity_id) ist. Die Verschachtelungsebene gibt an, wie tief diese Aktivität in seiner obersten übergeordneten Aktivität verschachtelt ist.
INVOCATION_ID	INTEGER	Unterscheidet einen bestimmten Aufruf einer Aktivität von anderen Aufrufen dieser Aktivität auf derselben Verschachtelungsebene.
ROUTINE_ID	INTEGER	Eindeutige Routinenkennung. Null, falls nicht Teil einer Routine.

WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Liste der Workloadvorkommen

Mit dieser Funktion wird die Liste aller Workloadvorkommen zurückgegeben, die in einer angegebenen Serviceklasse in einer bestimmten Partition ausgeführt werden. Ein Workloadvorkommen ist eine bestimmte Datenbankverbindung, deren Attribute mit der Definition einer Workload übereinstimmen und die somit mit der Workload verknüpft bzw. dieser zugeordnet ist.

Syntax

```
►►—WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES—(—servicesuperklassenname—,——►►  
►—serviceunterklassenname—,—dbpartitionsnummer—)——►►
```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

servicesuperklassenname

Ein Eingabeargument des Typs VARCHAR(128), das einen gültigen Service-superklassenamen in der Datenbank angibt, zu der zurzeit eine Verbindung besteht. Ist das Argument NULL oder eine leere Zeichenfolge, werden die Daten für alle Superklassen in der Datenbank abgerufen, für die die übrigen Parameter übereinstimmen.

serviceunterklassenname

Ein Eingabeargument des Typs VARCHAR(128), das einen gültigen Service-unterklassenamen in der Datenbank angibt, zu der zurzeit eine Verbindung besteht. Ist das Argument NULL oder eine leere Zeichenfolge, werden die Daten für alle Unterklassen in der Datenbank abgerufen, für die die übrigen Parameter übereinstimmen.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe des Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES.

Beispiel

Wenn ein Administrator anzeigen möchten, welche Workloadvorkommen auf dem System als Ganzes aktiv sind, kann die Funktion WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES mit einem Nullwert oder einer leeren Zeichenfolge für *servicesuperklassenname* und *serviceunterklassenname* sowie mit -2 für *dbpartitionsnummer* aufgerufen werden.

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,  
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,  
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,  
       SUBSTR(CHAR(COORD_PARTITION_NUM),1,9) AS COORDPART,  
       SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHNDL,
```

```

SUBSTR(WORKLOAD_NAME,1,18) AS WORKLOAD_NAME,
SUBSTR(CHAR(WORKLOAD_OCCURRENCE_ID),1,6) AS WLO_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES
(CAST(NULL AS VARCHAR(128)), CAST(NULL AS VARCHAR(128)), -2))
AS SCINFO
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART, APPHNDL,
WORKLOADDEF_NAME, WLO_ID

```

Unter der Annahme, dass das System über vier Datenbankpartitionen verfügt und zu diesem Zeitpunkt zwei Workloads ausführt, würde die oben aufgeführte Abfrage das folgende Ergebnis erzielen:

```

SUPERCLASS_NAME  SUBCLASS_NAME  PART  COORDPART  ...
-----
SYSDEFAULTMAINTENAN  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTSYSTEMCLA  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  1    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  1    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  2    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  2    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  3    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  3    0    ...

```

Ausgabe zu der Abfrage (Forts.)

```

... APPHNDL  WORKLOAD_NAME  WLO_ID
... -----
... -        -        -
... -        -        -
... 1        SYSDEFAULTWORKLOAD  1
... 2        SYSDEFAULTWORKLOAD  2

```

Hinweise

Die Parameter werden über AND verknüpft. Wenn Sie also Datensätze angeben, die miteinander in Konflikt stehen, beispielsweise eine Servicesuperklasse SUP_A und eine Unterklasse SUB_B, wobei SUB_B keine Unterklasse von SUP_A ist, werden keine Zeilen zurückgegeben.

Anmerkung: Die für das Workloadvorkommen erfassten Statistikdaten (z. B. coord_act_completed_total) werden zu Beginn jeder UOW (Unit of Work) zurückgesetzt, wenn sie mit den entsprechenden Workloadstatistikdaten kombiniert werden.

Zurückgegebene Informationen

Tabelle 50. Für WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
SERVICE_SUPERCLASS_NAME	VARCHAR(128)	Name der Servicesuperklasse, aus der dieser Datensatz erfasst wurde.
SERVICE_SUBCLASS_NAME	VARCHAR(128)	Name der Serviceunterklasse, aus der dieser Datensatz erfasst wurde.
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.

Tabelle 50. Für WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
COORD_PARTITION_NUM	SMALLINT	Die Partitionsnummer der Koordinatorpartition des angegebenen Workloadvorkommens.
APPLICATION_HANDLE	BIGINT	Eine systemweit eindeutige Kennung für die Anwendung. In einer Datenbank mit einer Partition besteht diese Kennung aus einem 16-Bit-Zähler. In einer Datenbank mit mehreren Partitionen besteht diese Kennung aus der Nummer der koordinierenden Partition, die mit einem 16-Bit-Zähler verknüpft ist. Außerdem ist diese Kennung auf jeder Partition gleich, auf der die Anwendung unter Umständen eine sekundäre Verbindung herstellt.
WORKLOAD_NAME	VARCHAR(128)	Name der Workload, aus der dieser Datensatz erfasst wurde.
WORKLOAD_OCCURRENCE_ID	INTEGER	Die Kennung (ID) des Workloadvorkommens. Dies ist erst in Kombination mit der Nummer der Datenbankkoordinatorpartition und dem Workloadnamen eine eindeutige Angabe des Workloadvorkommens. Alternativ kann anstelle der Nummer der Datenbankkoordinatorpartition auch die Anwendungskennung verwendet werden.
WORKLOAD_OCCURRENCE_STATE	VARCHAR(32)	Mögliche Werte: <ul style="list-style-type: none"> • DECOUPLED - Dem Workloadvorkommen wurde kein Koordinatoragent zugewiesen (Konzentratorfall). • DISCONNECTPEND - Die Verbindung des Workloadvorkommens zur Datenbank ist getrennt. • FORCED - Das Workloadvorkommen wurde erzwungen. • QUEUED - Der Koordinatoragent des Workloadvorkommens wurde auf Grund eines Warteschlangengrenzwerts des Workload-Managements in die Warteschlange gestellt. • TRANSIENT - Das Workloadvorkommen wurde noch keiner Servicesuperklasse zugeordnet. • UOWEXEC - Das Workloadvorkommen verarbeitet zurzeit eine Anforderung. • UOWWAIT - Das Workloadvorkommen wartet auf eine Anforderung vom Client.
UOW_ID	INTEGER	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der dieses Workloadvorkommen ursprünglich gestartet wurde.
SYSTEM_AUTH_ID	VARCHAR(128)	Systemberechtigungs-ID, unter der das Workloadvorkommen in das System eingefügt wurde.

Tabelle 50. Für WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
SESSION_AUTH_ID	VARCHAR(128)	Sitzungsberechtigungs-ID, unter der das Workloadvorkommen in das System eingefügt wurde.
APPLICATION_NAME	VARCHAR(128)	Der Name der Anwendung, die dieses Workloadvorkommen erstellt hat.
CLIENT_WRKSTNNAME	VARCHAR(255)	Der aktuelle Wert des Sonderregisters CLIENT_WRKSTNNAME für dieses Workloadvorkommen.
CLIENT_ACCTNG	VARCHAR(255)	Der aktuelle Wert des Sonderregisters CLIENT_ACCTNG für dieses Workloadvorkommen.
CLIENT_USER	VARCHAR(255)	Der aktuelle Wert des Sonderregisters CLIENT_USER für dieses Workloadvorkommen.
CLIENT_APPLNAME	VARCHAR(255)	Der aktuelle Wert des Sonderregisters CLIENT_APPLNAME für dieses Workloadvorkommen.
COORD_ACT_COMPLETED_TOTAL	INTEGER	Die Anzahl der Koordinatoraktivitäten auf allen Verschachtelungsebenen, die in der aktuellen UOW dieses Workloadvorkommens bis jetzt beendet wurden. Diese Statistik wird bei jeder Beendigung einer Aktivität in diesem Workloadvorkommen aktualisiert und zu Beginn jeder UOW zurückgesetzt.
COORD_ACT_ABORTED_TOTAL	INTEGER	Die Anzahl der Koordinatoraktivitäten, die in der aktuellen UOW dieses Workloadvorkommens bis jetzt abgebrochen wurden. Diese Statistik wird bei jedem Abbruch einer Aktivität in diesem Workloadvorkommen aktualisiert und zu Beginn jeder UOW zurückgesetzt.
COORD_ACT_REJECTED_TOTAL	INTEGER	Die Anzahl der Koordinatoraktivitäten, die in der aktuellen UOW dieses Workloadvorkommens bis jetzt zurückgewiesen wurden. Aktivitäten werden als 'zurückgewiesen' gewertet, wenn ihre Ausführung durch eine Arbeitsaktion zum Verhindern der Ausführung oder durch einen prädiktiven Schwellenwert verhindert wird. Diese Statistik wird bei jeder Zurückweisung einer Aktivität in diesem Workloadvorkommen aktualisiert und zu Beginn jeder UOW zurückgesetzt.
CONCURRENT_ACT_TOP	INTEGER	Die höchste Anzahl gleichzeitig ablaufender Aktivitäten auf allen Verschachtelungsebenen, entweder im Ausführungsstatus (dazu zählen der Status 'Inaktiv' und der Wartestatus) oder im Warteschlangenstatus, die für dieses Workloadvorkommen in der aktuellen UOW erreicht wurde. Diese Statistik wird zu Beginn jeder UOW zurückgesetzt.

WLM_GET_SERVICE_SUBCLASS_STATS - Zurückgeben der Statistikdaten von Serviceunterklassen

Mit dieser Funktion werden Basisstatistiken mindestens einer Serviceunterklasse zurückgegeben.

Syntax

```
►—WLM_GET_SERVICE_SUBCLASS_STATS—(—servicesuperklassenname—, —————►  
►—serviceunterklassenname—, —dbpartitionsnummer—)—————►◄
```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

servicesuperklassenname

Ein Eingabeargument des Typs VARCHAR(128), das beim Aufruf dieser Funktion einen gültigen Servicesuperklassenamen in der Datenbank angibt, zu der zurzeit eine Verbindung besteht. Ist das Argument NULL oder eine leere Zeichenfolge, werden die Daten für alle Superklassen in der Datenbank abgerufen.

serviceunterklassenname

Ein Eingabeargument des Typs VARCHAR(128), das beim Aufruf dieser Funktion einen gültigen Serviceunterklassenamen in der Datenbank angibt, zu der zurzeit eine Verbindung besteht. Ist das Argument NULL oder eine leere Zeichenfolge, werden die Daten für alle Unterklassen in der Datenbank abgerufen.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe des Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_SERVICE_SUBCLASS_STATS.

Beispiele

Beispiel 1: Da jede Aktivität vor ihrer Ausführung einer DB2-Serviceklasse zugeordnet werden muss, kann der globale Status des Systems regelmäßig überwacht werden, indem die Tabellenfunktionen für die Serviceklassenstatistiken verwendet werden und alle Serviceklassen in allen Partitionen abfragt werden. (Beachten Sie, dass das Übergeben eines Nullwerts für ein Argument bedeutet, dass das Ergebnis durch dieses Argument nicht eingeschränkt werden soll, mit Ausnahme des letzten Arguments, dbpartitionnum, bei dem -2 bedeutet, dass Daten von allen Datenbankpartitionen zurückgegeben werden sollen). Mit der folgenden Anweisung werden Serviceklassenstatistiken, wie durchschnittliche Laufzeit einer Aktivität und Standardabweichung in Sekunden, zurückgegeben:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,  
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,  
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
```

```

CAST(COORD_ACT_LIFETIME_AVG / 1000 AS DECIMAL(9,3)) AS AVGLIFETIME,
CAST(COORD_ACT_LIFETIME_STDDEV / 1000 AS DECIMAL(9,3)) AS STDDEVLIFETIME,
SUBSTR(CAST(LAST_RESET AS VARCHAR(30)),1,16) AS LAST_RESET
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS(CAST(NULL AS VARCHAR(128)),
CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART

```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

```

SUPERCLASS_NAME  SUBCLASS_NAME  PART ...
-----
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 0 ...
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 1 ...
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 2 ...
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 3 ...

```

Ausgabe zu der Abfrage (Forts.)

```

... AVGLIFETIME STDDEVLIFETIME LAST_RESET
... -----
...      691.242          34.322 2006-07-24-11.44
...      644.740          22.124 2006-07-24-11.44
...      612.431          43.347 2006-07-24-11.44
...      593.451          28.329 2006-07-24-11.44

```

Beispiel 2: Mit derselben Tabellenfunktion kann auch der höchste Wert für den durchschnittlichen gemeinsamen Zugriff von Koordinatoraktivitäten zurückgegeben werden, die in der Serviceklasse in den einzelnen Partitionen aktiv sind.

```

SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
CONCURRENT_ACT_TOP AS ACTTOP,
CONCURRENT_WLO_TOP AS CONNTOP
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS(CAST(NULL AS VARCHAR(128)),
CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART

```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

```

SUPERCLASS_NAME  SUBCLASS_NAME  PART ACTTOP  CONNTOP
-----
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 0          10        7
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 1           0         0
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 2           0         0
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 3           0         0

```

Die Ausgabe dieser Tabellenfunktion gibt dem Administrator einen guten Überblick auf hoher Ebene über die "Belastung" in jeder Partition für eine bestimmte Datenbank, indem er die durchschnittlichen Ausführungszeiten und die Anzahl der Aktivitäten überprüft. Erhebliche Abweichungen der Wertangaben auf hoher Ebene, die von diesen Tabellenfunktionen zurückgegeben werden, geben möglicherweise eine Änderung der Belastung auf dem System an.

Hinweise

Einige Statistikdaten werden nur zurückgegeben, wenn die Einstellungen für COLLECT AGGREGATE ACTIVITY DATA und COLLECT AGGREGATE REQUEST DATA für die entsprechende Serviceunterklasse auf einen anderen Wert als NONE gesetzt werden.

Die Tabellenfunktion WLM_GET_SERVICE_SUBCLASS_STATS gibt eine Datenzeile pro Serviceunterklasse und pro Partition zurück. Es wird keine Spaltenberechnung über mehrere Serviceklassen (in einer Partition) oder über mehrere Partitionen (für

eine oder mehrere Serviceklassen) ausgeführt. Die Spaltenberechnung kann jedoch über SQL-Abfragen erreicht werden, wie in den oben aufgeführten Beispielen gezeigt.

Die Parameter werden über AND verknüpft. Wenn Sie also Datensätze angeben, die miteinander in Konflikt stehen, beispielsweise einen Superklassennamen SUPA und einen Unterklassennamen SUBB, wobei SUBB keine Unterklasse von SUPA ist, werden keine Zeilen zurückgegeben.

Zurückgegebene Informationen

Tabelle 51. Für WLM_GET_SERVICE_SUBCLASS_STATS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
SERVICE_SUPERCLASS_NAME	VARCHAR(128)	Name der Servicesuperklasse, aus der dieser Datensatz erfasst wurde.
SERVICE_SUBCLASS_NAME	VARCHAR(128)	Name der Serviceunterklasse, aus der dieser Datensatz erfasst wurde.
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
LAST_RESET	TIMESTAMP	Zeitpunkt der letzten Zurücksetzung der Statistikdaten. Es gibt vier Ereignisse, deren Auftreten das Zurücksetzen der Statistikdaten auslöst, wodurch diese Zeitmarke aktualisiert wird: <ul style="list-style-type: none"> • Die Prozedur WLM_COLLECT_STATS wird aufgerufen. • Der Prozess der regelmäßigen Erfassung und Zurücksetzung, der vom Datenbankkonfigurationsparameter WLM_COLLECT_INT gesteuert wird, veranlasst eine Erfassung und Zurücksetzung. • Die Datenbank wird reaktiviert. • Die Serviceunterklasse, für die Statistikdaten erfasst werden, wurde modifiziert, und die Änderung wurde festgeschrieben. Die Zeitmarke LAST_RESET wird in Ortszeit angegeben.
COORD_ACT_COMPLETED_TOTAL	BIGINT	Die Gesamtzahl von Koordinatoraktivitäten, die Benutzer seit dem letzten Zurücksetzen übergeben haben und die erfolgreich beendet wurden.
COORD_ACT_ABORTED_TOTAL	BIGINT	Die Gesamtzahl von Koordinatoraktivitäten, die Benutzer seit dem letzten Zurücksetzen übergeben haben und die mit Fehlern beendet wurden.
COORD_ACT_REJECTED_TOTAL	BIGINT	Die Gesamtzahl von Koordinatoraktivitäten, die Benutzer seit dem letzten Zurücksetzen übergeben haben und die vor der Ausführung zurückgewiesen anstatt für die Ausführung zugelassen wurden. Aktivitäten werden als 'zurückgewiesen' gewertet, wenn ihre Ausführung durch eine Arbeitsaktion zum Verhindern der Ausführung oder durch einen prädiktiven Schwellenwert verhindert wird.

Tabelle 51. Für WLM_GET_SERVICE_SUBCLASS_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
CONCURRENT_ACT_TOP	INTEGER	Die höchste Anzahl gleichzeitig ablaufender Aktivitäten auf allen Verschachtelungsebenen, entweder im Ausführungsstatus (dazu zählen der Status 'Inaktiv' und der Wartestatus) oder im Warteschlangenstatus, die für diese Serviceunterklasse erreicht wurde.
COORD_ACT_LIFETIME_TOP	BIGINT	Höchstwert für die Laufzeit von Koordinatoraktivitäten. Der Wert wird für alle Verschachtelungsebenen berechnet. Null, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen.
COORD_ACT_LIFETIME_AVG	DOUBLE	Das arithmetische Mittel der Laufzeit von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Null, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen.
COORD_ACT_LIFETIME_STDDEV	DOUBLE	Die Standardabweichung der Laufzeit von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Null, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen. Diese Standardabweichung wird anhand des Histogramms für die Laufzeit der Koordinatoraktivität berechnet und ist möglicherweise ungenau, wenn die für die Daten erforderliche Größe des Histogramms nicht korrekt angegeben wurde. Ist einer der Werte im Histogramm-Bin enthalten, wird -1 zurückgegeben.
COORD_ACT_EXEC_TIME_AVG	DOUBLE	Das arithmetische Mittel der Ausführungszeiten von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Null, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen.

Tabelle 51. Für WLM_GET_SERVICE_SUBCLASS_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
COORD_ACT_EXEC_TIME_STDDEV	DOUBLE	Die Standardabweichung der Ausführungszeiten von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Die Zeit wird in Millisekunden gemessen. Diese Standardabweichung wird anhand des Histogramms für die Ausführungszeit der Koordinatoraktivität berechnet und ist möglicherweise ungenau, wenn die für die Daten erforderliche Größe des Histogramms nicht korrekt angegeben wurde. Ist einer der Werte im Histogramm-Bin enthalten, wird -1 zurückgegeben.
COORD_ACT_QUEUE_TIME_AVG	DOUBLE	Das arithmetische Mittel der Zeit in der Warteschlange von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezzeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Null, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen.
COORD_ACT_QUEUE_TIME_STDDEV	DOUBLE	Die Standardabweichung der Zeit in der Warteschlange von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Null, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen. Diese Standardabweichung wird anhand des Histogramms für die Zeit in der Warteschlange der Koordinatoraktivität berechnet und ist möglicherweise ungenau, wenn die für die Daten erforderliche Größe des Histogramms nicht korrekt angegeben wurde. Ist einer der Werte im Histogramm-Bin enthalten, wird -1 zurückgegeben.
NUM_REQUESTS_ACTIVE	BIGINT	Die Anzahl der Anforderungen, die zu der Zeit der Ausführung dieser Tabellenfunktion in der Serviceunterklasse ausgeführt werden.
NUM_REQUESTS_TOTAL	BIGINT	Die Anzahl der Anforderungen, deren Ausführung in dieser Serviceunterklasse seit dem letzten Zurücksetzen begonnen wurde. Dies gilt für alle Anforderungen unabhängig von ihrer Zugehörigkeit zu einer Aktivität. Wurde COLLECT AGGREGATE REQUEST DATA für diese Serviceunterklasse auf NONE gesetzt, hat diese Spalte den Wert NULL.

Tabelle 51. Für WLM_GET_SERVICE_SUBCLASS_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
REQUEST_EXEC_TIME_AVG	DOUBLE	Das arithmetische Mittel der Ausführungszeiten von Anforderungen, die dieser Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Die Zeit wird in Millisekunden gemessen. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Wurde COLLECT AGGREGATE REQUEST DATA für diese Serviceklasse auf NONE gesetzt, hat diese Spalte den Wert NULL.
REQUEST_EXEC_TIME_STDDEV	DOUBLE	Die Standardabweichung der Ausführungszeiten von Anforderungen, die dieser Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Die Zeit wird in Millisekunden gemessen. Wurde COLLECT AGGREGATE REQUEST DATA für diese Serviceklasse auf NONE gesetzt, hat diese Spalte den Wert NULL. Diese Standardabweichung wird anhand des Histogramms für die Ausführungszeit der Anforderung berechnet und ist möglicherweise ungenau, wenn die für die Daten erforderliche Größe des Histogramms nicht korrekt angegeben wurde. Ist einer der Werte im Histogramm-Bin enthalten, wird -1 zurückgegeben.
REQUEST_EXEC_TIME__TOTAL	BIGINT	Die Summe der Ausführungszeiten von Anforderungen, die dieser Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Die Zeit wird in Millisekunden gemessen. Wurde COLLECT AGGREGATE REQUEST DATA für diese Serviceklasse auf NONE gesetzt, hat diese Spalte den Wert NULL.

WLM_GET_SERVICE_SUPERCLASS_STATS - Zurückgeben der Statistikdaten von Servicesuperklassen

Mit dieser Funktion werden Basisstatistiken mindestens einer Servicesuperklasse zurückgegeben.

Syntax

```

▶—WLM_GET_SERVICE_SUPERCLASS_STATS—(—servicesuperklassenname—, —————▶
▶—dbpartitionsnummer—)—————▶▶
    
```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

servicesuperklassenname

Ein Eingabeargument des Typs VARCHAR(128), das beim Aufruf dieser Funk-

tion einen gültigen Servicesuperklassennamen in der Datenbank angibt, zu der zurzeit eine Verbindung besteht. Ist das Argument NULL oder eine leere Zeichenfolge, werden die Daten für alle Superklassen in der Datenbank abgerufen.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe des Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_SERVICE_SUPERCLASS_STATS.

Beispiel

Beispiel einer Abfrage der Basisstatistik für alle Servicesuperklassen auf dem System, in allen Datenbankpartitionen:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME, 1, 26) SERVICE_SUPERCLASS_NAME,
       DBPARTITIONNUM,
       LAST_RESET,
       CONCURRENT_CONNECTION_TOP CONCURRENT_CONN_TOP
FROM TABLE(WLM_GET_SERVICE_SUPERCLASS_STATS('', -2)) as SCSTATS
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

```
SERVICE_SUPERCLASS_NAME  DBPARTITIONNUM ...
-----
SYSDEFAULTSYSTEMCLASS    0 ...
SYSDEFAULTMAINTENANCECLASS 0 ...
SYSDEFAULTUSERCLASS      0 ...
```

Ausgabe zu der Abfrage (Forts.)

```
... LAST_RESET          CONCURRENT_CONN_TOP
... -----
... 2006-09-05-09.38.44.396788      0
... 2006-09-05-09.38.44.396795      0
... 2006-09-05-09.38.44.396796      1
```

Hinweise

Die Tabellenfunktion WLM_GET_SERVICE_SUPERCLASS_STATS gibt eine Datenzeile pro Servicesuperklasse und pro Partition zurück. Es wird keine Spaltenberechnung über mehrere Servicesuperklassen (in einer Partition) oder über mehrere Partitionen (für eine oder mehrere Servicesuperklassen) ausgeführt. Die Spaltenberechnung kann jedoch über SQL-Abfragen erreicht werden, wie im oben aufgeführten Beispiel gezeigt.

Zurückgegebene Informationen

Tabelle 52. Für WLM_GET_SERVICE_SUPERCLASS_STATS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
SERVICE_SUPERCLASS_NAME	VARCHAR(128)	Name der Servicesuperklasse, aus der dieser Datensatz erfasst wurde.

Table 52. Für WLM_GET_SERVICE_SUPERCLASS_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
LAST_RESET	TIMESTAMP	Zeitpunkt der letzten Zurücksetzung der Statistikdaten. Es gibt vier Ereignisse, deren Auftreten das Zurücksetzen der Statistikdaten auslöst, wodurch diese Zeitmarke aktualisiert wird: <ul style="list-style-type: none"> • Die Prozedur WLM_COLLECT_STATS wird aufgerufen. • Der Prozess der regelmäßigen Erfassung und Zurücksetzung, der vom Datenbankkonfigurationsparameter WLM_COLLECT_INT gesteuert wird, veranlasst eine Erfassung und Zurücksetzung. • Die Datenbank wird reaktiviert. • Die Servicesuperklasse, für die Statistikdaten erfasst werden, wurde modifiziert, und die Änderung wurde festgeschrieben. Die Zeitmarke LAST_RESET wird in Ortszeit angegeben.
CONCURRENT_CONNECTION_TOP	INTEGER	Die höchste Anzahl gleichzeitiger Koordinatorverbindungen, die seit dem letzten Zurücksetzen in dieser Klasse erreicht wurde.

WLM_GET_WORK_ACTION_SET_STATS - Zurückgeben von Statistikdaten zu einem Arbeitsaktionsset

Mit dieser Funktion werden die Statistikdaten für ein Arbeitsaktionsset zurückgegeben.

Syntax

```

WLM_GET_WORK_ACTION_SET_STATS(—arbeitsaktionsset-name—, —————→
—dbpartitionsnummer—)—————→

```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

arbeitsaktionsset-name

Ein Eingabeargument des Typs VARCHAR(128), das das Arbeitsaktionsset angibt, für das Statistikdaten zurückgegeben werden sollen. Ist das Argument NULL oder eine leere Zeichenfolge, werden die Statistikdaten für alle Arbeitsaktionssets zurückgegeben.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe des Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_WORK_ACTION_SET_STATS.

Beispiel

Angenommen, es gibt drei Arbeitsklassen, ReadClass, WriteClass und LoadClass. Es gibt eine Arbeitsaktion, die mit ReadClass verknüpft ist, sowie eine Arbeitsaktion, die mit LoadClass verknüpft ist, jedoch keine Arbeitsaktion, die mit WriteClass verknüpft ist. In Partition 0 gibt es 8 Aktivitäten, die derzeit in der ReadClass ausgeführt werden (oder in der Warteschlange stehen), 4 Aktivitäten, die derzeit in der WriteClass ausgeführt werden (oder in der Warteschlange stehen), 2 Aktivitäten, die derzeit in der LoadClass ausgeführt werden (oder in der Warteschlange stehen) und 3 Aktivitäten, die derzeit ausgeführt werden (oder in der Warteschlange stehen) und keiner Arbeitsklasse zugeordnet wurden. Da mit der Arbeitsklasse 'WriteClass' keine Arbeitsaktion verknüpft ist, werden die 4 Aktivitäten dieser Arbeitsklasse in der künstlichen Klasse '*' zusammen mit den 3 Aktivitäten zahlenmäßig erfasst, die keiner Arbeitsklasse zugeordnet wurden.

```
SELECT SUBSTR(WORK_ACTION_SET_NAME,1,18) AS WORK_ACTION_SET_NAME,  
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,  
       SUBSTR(WORK_CLASS_NAME,1,15) AS WORK_CLASS_NAME,  
       LAST_RESET,  
       SUBSTR(CHAR(WLO_ACT_TOTAL),1,14) AS TOTAL_WLO_ACTS  
FROM TABLE(WLM_GET_WORK_ACTION_SET_STATS  
            (CAST(NULL AS VARCHAR(128)), -2)) AS WASSTATS  
ORDER BY WORK_ACTION_SET_NAME, WORK_CLASS_NAME, PART
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

WORK_ACTION_SET_NAME	PART	WORK_CLASS_NAME	LAST_RESET	TOTAL_WLO_ACTS
AdminActionSet	0	ReadClass	2005-11-25-18.52.49.343000	8
AdminActionSet	1	ReadClass	2005-11-25-18.52.50.478000	0
AdminActionSet	0	LoadClass	2005-11-25-18.52.49.343000	2
AdminActionSet	1	LoadClass	2005-11-25-18.52.50.478000	0
AdminActionSet	0	*	2005-11-25-18.52.49.343000	7
AdminActionSet	1	*	2005-11-25-18.52.50.478000	0

Zurückgegebene Informationen

Tabelle 53. Für WLM_GET_WORK_ACTION_SET_STATS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
WORK_ACTION_SET_NAME	VARCHAR(128)	Der Name des Arbeitsaktionssets.
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.

Tabelle 53. Für WLM_GET_WORK_ACTION_SET_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
LAST_RESET	TIMESTAMP	<p>Zeitpunkt der letzten Zurücksetzung der Statistikdaten. Es gibt vier Ereignisse, deren Auftreten das Zurücksetzen der Statistikdaten auslöst, wodurch diese Zeitmarke aktualisiert wird:</p> <ul style="list-style-type: none"> • Die Prozedur WLM_COLLECT_STATS wird aufgerufen. • Der Prozess der regelmäßigen Erfassung und Zurücksetzung, der vom Datenbankkonfigurationsparameter WLM_COLLECT_INT gesteuert wird, veranlasst eine Erfassung und Zurücksetzung. • Die Datenbank wird reaktiviert. • Das Arbeitsaktionsset, für das Statistikdaten erfasst werden, wurde modifiziert, und die Änderung wurde festgeschrieben. <p>Die Zeitmarke LAST_RESET wird in Ortszeit angegeben.</p>
WORK_CLASS_NAME	VARCHAR(128)	<p>Der Name der Arbeitsklasse, die zu dem angegebenen Arbeitsaktionsset gehört. Mit dieser Arbeitsklasse muss eine Arbeitsaktion verknüpft sein, damit sie in dieser Tabelle angezeigt wird. '*' stellt eine künstliche Arbeitsklasse dar, die erstellt wird, um alle Aktivitäten zahlenmäßig zu erfassen, die nicht zu den übrigen Arbeitsklassen gehören, für die der Benutzer mindestens eine Arbeitsaktion zugeordnet hat.</p>
ACT_TOTAL	BIGINT	<p>Die Anzahl der Aktivitäten auf allen Verschachtelungsebenen, die der mit WORK_CLASS_NAME angegebenen Arbeitsklasse zugeordnet wurden.</p>

WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Zurückgeben einer Liste von Aktivitäten

Mit dieser Funktion wird die Liste aller Aktivitäten zurückgegeben, die über die angegebene Anwendung in der angegebenen Partition übergeben und noch nicht beendet wurden.

Syntax

```

▶▶ WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES (—anwendungskennung—, —————▶
▶—dbpartitionsnummer—) —————▶▶

```

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

anwendungskennung

Ein Eingabeargument des Typs BIGINT, das eine Anwendungskennung angibt, für die eine Liste von Aktivitäten zurückgegeben werden soll. Ist das Argument NULL, werden die Daten für alle Anwendungen in der Datenbank abgerufen, für die die übrigen Parameter übereinstimmen.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe des Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES.

Beispiel

Nachdem eine Anwendungskennung angegeben wurde, können alle Aktivitäten, die derzeit in dieser Anwendung aktiv sind, nachgesehen werden. Angenommen, ein Administrator möchte z. B. die Aktivitäten einer Anwendung auflisten, deren Anwendungskennung mit Hilfe des Befehls zum Auflisten von Anwendungen mit 1 ermittelt wurde:

```
SELECT SUBSTR(CHAR(COORD_PARTITION_NUM),1,5) AS COORD,  
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,  
       SUBSTR(CHAR(UOW_ID),1,5) AS UOWID,  
       SUBSTR(CHAR(ACTIVITY_ID),1,5) AS ACTID,  
       SUBSTR(CHAR(PARENT_UOW_ID),1,8) AS PARUOWID,  
       SUBSTR(CHAR(PARENT_ACTIVITY_ID),1,8) AS PARACTID,  
       ACTIVITY_TYPE AS ACTTYPE,  
       SUBSTR(CHAR(NESTING_LEVEL),1,7) AS NESTING  
FROM TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(1, -2)) AS WLOACTS  
ORDER BY PART, UOWID, ACTID
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

COORD	PART	UOWID	ACTID	PARUOWID	PARACTID	ACTTYPE	NESTING
0	0	2	3	-	-	CALL	0
0	0	2	5	2	3	READ_DML	1

Hinweise

Der ACTIVITY_STATE-Wert QUEUED bedeutet, dass die Koordinatoraktivität versucht, die Schwellenwerttickets über einen Fernprozeduraufruf abzurufen, aber bisher keine Antwort erhalten hat. Wenn dieser Status angezeigt wird, kann dies bedeuten, dass die Aktivität von WLM in die Warteschlange gestellt wurde. Wird er nur kurzzeitig angezeigt, kann dies darauf hindeuten, dass die Aktivität gerade ihre Tickets abrufen. Wenn Sie sich ein genaueres Bild darüber machen wollen, ob die Aktivität tatsächlich in die Warteschlange gestellt wird, können Sie ermitteln, welcher Agent die Aktivität verarbeitet und feststellen, ob das Ereignisobjekt dieses Agenten in der Katalogpartition den Wert WLM_QUEUE aufweist.

Zurückgegebene Informationen

Tabelle 54. Von WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
APPLICATION_HANDLE	BIGINT	Eine systemweit eindeutige Kennung für die Anwendung. In einer Datenbank mit einer Partition besteht diese Kennung aus einem 16-Bit-Zähler. In einer Datenbank mit mehreren Partitionen besteht diese Kennung aus der Nummer der koordinierenden Partition, die mit einem 16-Bit-Zähler verknüpft ist. Außerdem ist diese Kennung auf jeder Partition gleich, auf der die Anwendung unter Umständen eine sekundäre Verbindung herstellt.
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
COORD_PARTITION_NUM	SMALLINT	Die Koordinatorpartition der Aktivität.
LOCAL_START_TIME	TIMESTAMP	Der Zeitpunkt des Starts dieser Aktivität in der Partition. Die Angabe erfolgt in Ortszeit. Dieses Feld kann den Wert NULL haben, wenn eine Aktivität in das System eingegangen ist, diese Aktivität sich aber in einer Warteschlange befindet und die Ausführung der Aktivität noch nicht gestartet wurde.
UOW_ID	INTEGER	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der die Aktivität ursprünglich gestartet wurde.
ACTIVITY_ID	INTEGER	Eindeutige Aktivitäts-ID innerhalb einer UOW (Unit of Work).
PARENT_UOW_ID	INTEGER	Eindeutige Kennung der UOW (Unit of Work) innerhalb einer Anwendung. Verweist auf die UOW, in der die übergeordnete Aktivität dieser Aktivität ursprünglich gestartet wurde. Gibt ein leeres Feld zurück, wenn die Aktivität keine übergeordnete Aktivität aufweist oder auf einer fernen Partition ausgeführt wird.
PARENT_ACTIVITY_ID	INTEGER	Eindeutige Aktivitäts-ID innerhalb einer UOW für die übergeordnete Aktivität der Aktivität mit der ID ACTIVITY_ID. Gibt ein leeres Feld zurück, wenn die Aktivität keine übergeordnete Aktivität aufweist oder auf einer fernen Partition ausgeführt wird.

Tabelle 54. Von WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
ACTIVITY_STATE	VARCHAR(32)	<p>Die folgenden Werte sind möglich:</p> <ul style="list-style-type: none"> • CANCEL_PENDING - Die Aktivität wurde abgebrochen, als kein Agent aktiv an einer Anforderung für die Aktivität gearbeitet hat. Wenn das nächste Mal eine Anforderung als Teil der Aktivität übergeben wird, wird die Aktivität abgebrochen und der Benutzer, der die Aktivität übergeben hat, erhält einen SQL4725N-Fehler. • EXECUTING - Agenten arbeiten aktiv an einer Anforderung für die Aktivität. • IDLE - Es gibt keinen Agenten, der eine Anforderung für die Aktivität aktiv verarbeitet. • INITIALIZING - Die Aktivität wurde übergeben, ihre Ausführung wurde jedoch noch nicht gestartet. Während des Initialisierungsstatus werden prädiktive Schwellenwerte auf die Aktivität angewendet, um festzustellen, ob die Ausführung der Aktivität zulässig ist oder nicht. • QP_CANCEL_PENDING - Ist mit dem Status CANCEL_PENDING identisch, allerdings wurde die Aktivität von Query Patroller abgebrochen und nicht von der Prozedur WLM_CANCEL_ACTIVITY. • QP_QUEUED - Die Aktivität wird von Query Patroller in die Warteschlange gestellt. • QUEUED - Die Koordinatoraktivität hat über einen Fernprozeduraufruf (RPC) an die Katalogpartition versucht, Schwellenwerttickets abzurufen, und hat noch keine Antwort erhalten. • TERMINATING - Die Ausführung der Aktivität ist beendet, und die Aktivität wird aus dem System entfernt.
ACTIVITY_TYPE	VARCHAR(32)	<p>Die folgenden Werte sind möglich:</p> <ul style="list-style-type: none"> • CALL • DDL • LOAD • OTHER • READ_DML • WRITE_DML <p>Eine Beschreibung der verschiedenen Typen von SQL-Anweisungen, die den einzelnen Aktivitätstypen zugeordnet sind, finden Sie unter „Arbeitstypen und SQL-Anweisungen von Arbeitsklassen“ in <i>Workload-Manager Handbuch und Referenz</i>.</p>

Table 54. Von WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
NESTING_LEVEL	INTEGER	Die Verschachtelungsebene dieser Aktivität. Die Verschachtelungsebene gibt an, wie tief diese Aktivität in seiner obersten übergeordneten Aktivität verschachtelt ist.
INVOCATION_ID	INTEGER	Unterscheidet einen bestimmten Aufruf dieser Aktivität von anderen Aufrufen dieser Aktivität in derselben Verschachtelungsebene.
ROUTINE_ID	INTEGER	Eindeutige Routinenkennung.
UTILITY_ID	INTEGER	Handelt es sich bei der Aktivität um ein Dienstprogramm, ist dies die Kennung (ID) des Dienstprogramms. Andernfalls hat dieses Feld den Wert NULL.
SERVICE_CLASS_ID	INTEGER	Eindeutige Kennung (ID) der Serviceklasse, zu der diese Aktivität gehört.
DATABASE_WORK_ACTION_SET_ID	INTEGER	Wenn diese Aktivität innerhalb einer Arbeitsklasse im Datenbankbereich kategorisiert wurde, enthält diese Spalte die ID des Arbeitsklassensets, zu der diese Arbeitsklasse gehört. Diese Spalte enthält '0', wenn die Aktivität nicht in eine Arbeitsklasse der Datenbank kategorisiert wurde.
DATABASE_WORK_CLASS_ID	INTEGER	Wenn diese Aktivität innerhalb einer Arbeitsklasse im Datenbankbereich kategorisiert wurde, enthält diese Spalte die ID der Arbeitsklasse. Diese Spalte enthält '0', wenn die Aktivität nicht in eine Arbeitsklasse der Datenbank kategorisiert wurde.
SERVICE_CLASS_WORK_ACTION_SET_ID	INTEGER	Wenn diese Aktivität innerhalb einer Arbeitsklasse im Serviceklassenbereich kategorisiert wurde, enthält diese Spalte die ID des Arbeitsaktionssets, die dem Arbeitsklassenset zugeordnet ist, zu dem die Arbeitsklasse gehört. Diese Spalte enthält '0', wenn die Aktivität nicht in eine Arbeitsklasse der Serviceklasse kategorisiert wurde.
SERVICE_CLASS_WORK_CLASS_ID	INTEGER	Wenn diese Aktivität innerhalb einer Arbeitsklasse im Serviceklassenbereich kategorisiert wurde, enthält diese Spalte die ID der Arbeitsklasse, die dieser Aktivität zugeordnet ist. Diese Spalte enthält '0', wenn die Aktivität nicht in eine Arbeitsklasse der Serviceklasse kategorisiert wurde.

WLM_GET_WORKLOAD_STATS - Zurückgeben von Workloadstatistikdaten

Mit dieser Funktion werden Workloadstatistikdaten für jede Kombination von Workloadname und Datenbankpartitionsnummer zurückgegeben.

Syntax

►—WLM_GET_WORKLOAD_STATS—(—workloadname—,—dbpartitionsnummer—)————►

Das Schema ist SYSPROC.

Tabellenfunktionsparameter

workloadname

Ein Eingabeargument des Typs VARCHAR(128), das eine bestimmte Workload angibt, für die die Statistikdaten zurückgegeben werden sollen. Ist das Argument NULL oder eine leere Zeichenfolge, werden Statistikdaten für alle Workloads zurückgegeben.

dbpartitionsnummer

Ein Eingabeargument des Typs INTEGER, das beim Aufruf dieser Funktion eine gültige Partitionsnummer in derselben Instanz wie die Datenbank angibt, zu der zurzeit eine Verbindung besteht. Geben Sie -1 für die aktuelle Datenbankpartition oder -2 für alle Datenbankpartitionen an. Bei Angabe eines Nullwerts wird -1 implizit gesetzt.

Berechtigung

Zugriffsrecht EXECUTE für die Funktion WLM_GET_WORKLOAD_STATS.

Beispiel

Ein Administrator möchte die Statistikdaten für Workloads anzeigen. Dies ist mit der folgenden Abfrage möglich:

```
SELECT SUBSTR(WORKLOAD_NAME,1,22) AS WL_DEF_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CONCURRENT_WLO_TOP AS WLO_TOP,
       CONCURRENT_WLO_ACT_TOP AS WLO_ACT_TOP
FROM TABLE(WLM_GET_WORKLOAD_STATS(CAST(NULL AS VARCHAR(128)), -2))
AS WLSTATS
ORDER BY WL_DEF_NAME, PART
```

Das folgende Beispiel zeigt die Ausgabe dieser Abfrage.

WL_DEF_NAME	PART	WLO_TOP	WLO_ACT_TOP
MYUSERWORKLOAD	0	2	8
MYUSERWORKLOAD	1	0	0
SYSDEFAULTUSERWORKLOAD	0	1	1
SYSDEFAULTUSERWORKLOAD	1	0	0

Dieses Beispiel zeigt, dass in Partition 0 die höchste Anzahl gleichzeitiger Vorkommen der Workload MYUSERWORKLOAD 2 lautet und die höchste Anzahl gleichzeitiger Aktivitäten in einer dieser Workloadvorkommen 8 beträgt.

Hinweise

Mit dieser Funktion wird eine Zeile für jede Kombination von Workloadname und Datenbankpartitionsnummer zurückgegeben. Es wird keine Spaltenberechnung über mehrere Workloads, Partitionen oder Serviceklassen hinweg ausgeführt. Die Spaltenberechnung kann jedoch über SQL-Abfragen erreicht werden.

Zurückgegebene Informationen

Tabelle 55. Von WLM_GET_WORKLOAD_STATS zurückgegebene Informationen

Spaltenname	Datentyp	Beschreibung
WORKLOAD_NAME	BIGINT	Name der Workload, aus der dieser Datensatz erfasst wurde.
DBPARTITIONNUM	SMALLINT	Nummer der Partition, aus der dieser Datensatz erfasst wurde.
LAST_RESET	TIMESTAMP	<p>Zeitpunkt der letzten Zurücksetzung der Statistikdaten. Es gibt vier Ereignisse, deren Auftreten das Zurücksetzen der Statistikdaten auslöst, wodurch diese Zeitmarke aktualisiert wird:</p> <ul style="list-style-type: none"> • Die Prozedur WLM_COLLECT_STATS wird aufgerufen. • Der Prozess der regelmäßigen Erfassung und Zurücksetzung, der vom Datenbankkonfigurationsparameter WLM_COLLECT_INT gesteuert wird, veranlasst eine Erfassung und Zurücksetzung. • Die Datenbank wird reaktiviert. • Die Workload, für die Statistikdaten erfasst werden, wurde modifiziert, und die Änderung wurde festgeschrieben. <p>Die Zeitmarke LAST_RESET wird in Ortszeit angegeben.</p>
CONCURRENT_WLO_TOP	INTEGER	Die höchste Anzahl gleichzeitiger Vorkommen der angegebenen Workload in dieser Partition seit dem letzten Zurücksetzen.
CONCURRENT_WLO_ACT_TOP	INTEGER	Die höchste Anzahl gleichzeitig ablaufender Aktivitäten (einschließlich Koordinatoraktivitäten und verschachtelter Aktivitäten), entweder im Ausführungsstatus (dazu zählen der Status 'Inaktiv' und der Wartestatus) oder im Warteschlangenstatus, die seit dem letzten Zurücksetzen in einem Vorkommen dieser Workload erreicht wurde. Wird von jedem Workloadvorkommen am Ende der entsprechenden UOW aktualisiert.
COORD_ACT_COMPLETED_TOTAL	BIGINT	Die Gesamtzahl von Koordinatoraktivitäten auf jeder Verschachtelungsebene, die einem Vorkommen dieser Workload zugeordnet sind und seit dem letzten Zurücksetzen beendet wurden. Wird von jedem Workloadvorkommen am Ende der entsprechenden UOW aktualisiert.
COORD_ACT_ABORTED_TOTAL	BIGINT	Die Gesamtzahl von Koordinatoraktivitäten auf jeder Verschachtelungsebene, die einem Vorkommen dieser Workload zugeordnet sind und seit dem letzten Zurücksetzen vor der Beendigung abgebrochen wurden. Wird von jedem Workloadvorkommen am Ende der entsprechenden UOW aktualisiert.

Table 55. Von WLM_GET_WORKLOAD_STATS zurückgegebene Informationen (Forts.)

Spaltenname	Datentyp	Beschreibung
COORD_ACT_REJECTED_TOTAL	BIGINT	Die Gesamtzahl von Koordinatoraktivitäten auf jeder Verschachtelungsebene, die einem Vorkommen dieser Workload zugeordnet sind und seit dem letzten Zurücksetzen vor der Ausführung zurückgewiesen wurden. Wird von jedem Workloadvorkommen am Ende der entsprechenden UOW aktualisiert. Aktivitäten werden als 'zurückgewiesen' gewertet, wenn ihre Ausführung durch eine Arbeitsaktion zum Verhindern der Ausführung oder durch einen prädiktiven Schwellenwert verhindert wird. Beachten Sie, dass im Gegensatz zur Spalte mit demselben Namen in der Funktion WLM_GET_SERVICE_SUBCLASS_STATS hier auch die Zurückweisungen gezählt werden, die auftreten, bevor eine Aktivität einer Serviceklasse zugeordnet werden kann. Eine Zurückweisung dieser Art tritt beispielsweise auf, wenn eine Aktivität gegen den Schwellenwert ConcurrentWorkloadOccurrences verstößt.
WLO_COMPLETED_TOTAL	BIGINT	Die Anzahl von abgeschlossenen Verarbeitungsprozessen, die seit dem letzten Zurücksetzen ausgeführt wurden.

Kapitel 9. Monitorelemente

Monitorelemente für Workload-Management

Die folgenden Monitorelemente stellen Informationen zu Aktivitäten, Schwellenwertverstößen und Statistiken für das Workload-Management bereit.

activate_timestamp - Zeitmarke für Aktivierung (Monitorelement)

Der Zeitpunkt, zu dem ein Ereignismonitor aktiviert wurde.

Elementkennung

activate_timestamp

Elementtyp

Zeitmarke

Tabelle 56. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivität	event_activity	-
Aktivität	event_activitystmt	-
Aktivität	event_activityvals	-
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Mit Hilfe dieses Elements können Informationen korreliert werden, die von den vorstehenden Ereignistypen zurückgegeben werden.

activity_collected - Erfassen von Aktivitäten (Monitorelement)

Dieses Element gibt an, ob für einen Schwellenwert, gegen den verstoßen wurde, Datensätze des Aktivitätsereignismonitors erfasst werden sollen.

Elementkennung

activity_collected

Elementtyp

Information

Tabelle 57. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Mit diesem Element lässt sich ermitteln, ob für eine Aktivität, die gegen den Schwellenwert verstoßen hat, ein an den Aktivitätsereignismonitor geschriebenes Aktivitätsereignis erwartet werden muss.

Wird eine Aktivität beendet oder abgebrochen und ist der Aktivitätsereignismonitor zu diesem Zeitpunkt aktiv, wird die Aktivität, die gegen den betreffenden Schwellenwert verstoßen hat, erfasst, sofern der Wert dieses Monitorelements 'Y' ist. Ist der Wert dieses Monitorelements 'N', wird die Aktivität nicht erfasst.

activity_id - Aktivitäts-ID (Monitorelement)

Zähler, der eine Aktivität für eine Anwendung innerhalb einer UOW eindeutig identifiziert. In Verbindung mit **appl_id** und **uow_id** in einem Ereignismonitordatensatz für Aktivitäten wird eine erfasste Aktivität von diesem Monitorelement eindeutig identifiziert. In Verbindung mit **appl_id** und **uow_id** in einem Ereignismonitordatensatz für Schwellenwertverstöße wird eine Aktivität, die gegen einen Schwellenwert verstoßen hat, von diesem Monitorelement eindeutig identifiziert.

Elementkennung

activity_id

Elementtyp

Information

Tabelle 58. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-
Aktivitäten	event_activitystmt	-
Aktivitäten	event_activityvals	-
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse des Verhaltens einer Aktivität.

Sie können dieses Element auch in Verbindung mit den Monitorelementen **uow_id** und **agent_id** verwenden, um eine Aktivität eindeutig zu identifizieren.

activity_secondary_id - Sekundäre Aktivitäts-ID (Monitorelement)

Der Wert für dieses Element wird schrittweise erhöht, sobald ein weiterer Aktivitätsdatensatz für eine bestimmte Aktivität geschrieben wird. Wird z. B. für eine Aktivität ein Aktivitätsdatensatz geschrieben, weil die Prozedur **WLM_CAPTURE_ACTIVITY_IN_PROGRESS** aufgerufen wurde, und ein weiterer Datensatz beim Beenden der Aktivität, weist das Element für den ersten Datensatz den Wert 0 und für den zweiten Datensatz den Wert 1 auf.

Elementkennung

activity_secondary_id

Elementtyp

Information

Tabelle 59. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-
Aktivitäten	event_activitystmt	-

Tabelle 59. Informationen zur Ereignisüberwachung (Forts.)

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activityvals	-

Verwendung

In Verbindung mit den Monitorelementen **activity_id**, **uow_id** und **appl_id** können Sie mit diesem Element Aktivitätsdatensätze eindeutig identifizieren, wenn Informationen zu einer Aktivität mehrmals in den Aktivitätsereignismonitor geschrieben wurden.

Beispiel: Informationen zu einer Aktivität werden in den folgenden Fällen zweimal an den Aktivitätsereignismonitor gesendet:

- Unter Verwendung der gespeicherten Prozedur `WLM_CAPTURE_ACTIVITY_IN_PROGRESS` wurden während der Ausführung der Aktivität Informationen zur Aktivität erfasst.
- Informationen zur Aktivität wurden nach Ausführung der Aktivität erfasst, da die Klausel `COLLECT ACTIVITY DATA` für die Serviceklasse angegeben wurde, der die Aktivität zugeordnet ist.

activity_type - Aktivitätstyp (Monitorelement)

Der Aktivitätstyp, auf den sich der Aktivitätsdatensatz bezieht.

Elementkennung

activity_type

Elementtyp

Information

Tabelle 60. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Die möglichen Werte sind:

- LOAD
- READ_DML
- WRITE_DML
- DDL
- CALL
- OTHER

Für ferne Partitionen ist der Wert dieses Monitorelements stets OTHER.

act_exec_time - Ausführungszeit für Aktivitäten (Monitorelement)

Die aufgewendete Ausführungszeit für die Partition (in Mikrosekunden). Bei Cursorn ist die Ausführungszeit die kombinierte Zeit für das Öffnen, das Abrufen und das Schließen. Die Zeit, in der ein Cursor inaktiv ist, zählt nicht zur Ausführungszeit. Für Routinen umfasst die Ausführungszeit die Zeit vom Start bis zum Ende

des Routinenaufrufs. Die Laufzeit von Cursors, die von einer Routine nach deren Beendigung offen gelassen werden (zur Rückgabe einer Ergebnismenge), zählt nicht zur Ausführungszeit der Routine. Bei allen anderen Aktivitäten ist die Ausführungszeit die Differenz zwischen Start- und Stoppzeit. Für alle diese Fälle gilt: Die Ausführungszeit umfasst weder die Zeit für die Initialisierung noch die in einer Warteschlange verbrachte Zeit.

Elementkennung

act_exec_time

Elementtyp

Zeit

Tabelle 61. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element gibt für sich allein Aufschluss darüber, wie viel Zeit von DB2 für die Ausführung einer Aktivität auf den einzelnen Partitionen verbraucht wurde. In Verbindung mit den Monitorelementen **time_started** und **time_completed** kann dieses Element auf der Koordinatorpartition verwendet werden, um die Leerlaufzeit für Cursoraktivitäten zu berechnen. Hierfür wird folgende Formel verwendet:

$$\text{Cursor id+le time} = (\text{time_completed} - \text{time_started}) - \text{act_exec_time}$$

act_total - Gesamtzahl der Aktivitäten (Monitorelement)

Die Gesamtzahl der Aktivitäten auf allen Verschachtelungsebenen, auf die seit dem letzten Zurücksetzen Arbeitsaktionen angewendet wurden, die der angegebenen Arbeitsklasse entsprechen.

Elementkennung

act_total

Elementtyp

Zähler

Tabelle 62. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wcstats	-

Verwendung

Jedes Mal, wenn auf eine Aktivität mindestens eine Arbeitsaktion angewendet wird, die einer Arbeitsklasse zugeordnet ist, wird ein Zähler für die Arbeitsklasse aktualisiert. Dieser Zähler wird mit Hilfe des Monitorelements **act_total** dargestellt. Mit diesem Zähler kann die Effektivität des Arbeitsaktionssets bewertet werden (z. B. auf wie viele Aktivitäten Aktionen angewendet wurden). Darüber hinaus gibt dieser Zähler Aufschluss über die verschiedenen Aktivitätstypen auf dem System.

arm_correlator - Korrelator zur Messung der Anwendungsantwortzeit (Monitorelement)

Transaktionskennung gemäß dem ARM-Standard (Application Response Measurement).

Elementkennung
arm_correlator

Elementtyp
Information

Tabelle 63. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element kann dazu verwendet werden, eine vom Ereignismonitor für Aktivitäten erfasste Aktivität mit den Anwendungen zu verknüpfen, die der Aktivität zugeordnet sind. Voraussetzung ist jedoch, dass die betreffenden Anwendungen den ARM-Standard (Application Response Measurement) unterstützen.

bin_id - ID des Histogramm-Bins (Monitorelement)

Die ID eines Histogramm-Bins. Der Wert für **bin_id** ist innerhalb des Histogramms eindeutig.

Elementkennung
bin_id

Elementtyp
Information

Tabelle 64. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-

Verwendung

Dieses Element ermöglicht die Unterscheidung von Bins innerhalb eines Histogramms.

bottom - Untere Grenze von Histogramm-Bin (Monitorelement)

Die Untergrenze (nicht Teil des Bereichs) des für ein Histogramm-Bin festgelegten Bereichs. Der Wert dieses Monitorelements stellt gleichzeitig die Obergrenze (Teil des Bereichs) des Bereichs des vorangehenden Histogramm-Bins dar, soweit vorhanden.

Elementkennung
bottom

Elementtyp
Information

Tabelle 65. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-

Verwendung

Mit Hilfe dieses Elements kann in Verbindung mit dem entsprechenden Element **top** der Bereich für ein Bin in einem Histogramm bestimmt werden.

concurrent_act_top - Maximaler Wert für gleichzeitige Aktivitäten (Monitorelement)

Der Höchstwert für die gleichzeitig ablaufenden Aktivitäten (auf allen Verschachtelungsebenen) in einer Serviceunterklasse nach dem letzten Zurücksetzen.

Elementkennung

concurrent_act_top

Elementtyp

Grenzwert

Tabelle 66. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-

Verwendung

Dieses Element gibt Aufschluss über den Höchstwert für die Gleichzeitigkeit von Aktivitäten (inklusive verschachtelter Aktivitäten), der auf einer Partition innerhalb des erfassten Zeitintervalls für eine Serviceunterklasse erreicht wurde.

concurrent_connection_top - Maximaler Wert für gleichzeitige Verbindungen (Monitorelement)

Der Höchstwert für gleichzeitige Koordinatorverbindungen der Serviceklasse, der nach dem letzten Zurücksetzen erreicht wurde. Dieses Feld weist in den einzelnen Unterklassen einer Superklasse denselben Wert auf.

Elementkennung

concurrent_connection_top

Elementtyp

Grenzwert

Tabelle 67. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-

Verwendung

Dieses Element gibt durch die Angabe des aktuellen Höchstwerts Hinweise darauf, an welcher Stelle Schwellenwerte für die Gleichzeitigkeit von Verbindungen festgelegt werden sollten. Darüber hinaus kann mit diesem Element überprüft werden, ob ein Schwellenwert korrekt definiert ist und seinen Zweck erfüllt.

concurrent_wlo_act_top - Maximaler Wert für gleichzeitige Workloadaktivitäten (Monitorelement)

Der Höchstwert für gleichzeitig ablaufende Aktivitäten (auf allen Verschachtelungsebenen) für beliebige Vorkommen einer bestimmten Workload seit dem letzten Zurücksetzen.

Elementkennung

concurrent_wlo_act_top

Elementtyp

Grenzwert

Tabelle 68. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wlstats	-

Verwendung

Dieses Element gibt Aufschluss über die höchste Anzahl gleichzeitiger Aktivitäten, die auf einer Partition innerhalb des erfassten Zeitintervalls für beliebige Vorkommen der Workload erreicht wurde.

concurrent_wlo_top - Maximaler Wert für gleichzeitige Workloadvorkommen (Monitorelement)

Der Höchstwert für die gleichzeitige Ausführung von Verarbeitungsprozessen seit dem letzten Zurücksetzen.

Elementkennung

concurrent_wlo_top

Elementtyp

Grenzwert

Tabelle 69. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wlstats	-

Verwendung

Dieses Element gibt Aufschluss über die höchste Anzahl gleichzeitiger Workloadvorkommen, die auf einer Partition innerhalb des erfassten Zeitintervalls für eine Workload erreicht wurde.

coord_act_aborted_total - Gesamtzahl abgebrochener Koordinatoraktivitäten (Monitorelement)

Die Gesamtzahl von Koordinatoraktivitäten auf allen Verschachtelungsebenen, die seit dem letzten Zurücksetzen mit Fehler beendet wurden. Bei Serviceklassen wird dieser Wert beim Beenden von Aktivitäten aktualisiert. Bei Workloads wird der Wert bei Abschluss der zugehörigen UOW mit den einzelnen Workloadvorkommen aktualisiert.

Elementkennung

coord_act_aborted_total

Elementtyp
Zähler

Tabelle 70. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wlstats	-

Verwendung

Dieses Element gibt Aufschluss darüber, ob Aktivitäten auf dem System erfolgreich abgeschlossen werden. Aktivitäten können storniert oder aufgrund von Fehlern oder reaktiven Schwellenwerten vorzeitig abgebrochen werden.

coord_act_completed_total - Gesamtzahl abgeschlossener Koordinatoraktivitäten (Monitorelement)

Die Gesamtzahl von Koordinatoraktivitäten auf allen Verschachtelungsebenen, die seit dem letzten Zurücksetzen erfolgreich abgeschlossen wurden. Bei Serviceklassen wird dieser Wert beim Beenden von Aktivitäten aktualisiert. Bei Workloads wird der Wert bei Abschluss der zugehörigen UOW mit den einzelnen Workloadvorkommen aktualisiert.

Elementkennung
coord_act_completed_total

Elementtyp
Zähler

Tabelle 71. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wlstats	-
Statistiken	event_scstats	-

Verwendung

Dieses Element gibt Aufschluss über den Durchsatz an Aktivitäten im System und ist bei der Berechnung der durchschnittlichen Laufzeit von Aktivitäten über mehrere Partitionen hinweg hilfreich.

coord_act_lifetime_top - Maximaler Wert für die Laufzeit von Koordinatoraktivitäten (Monitorelement)

Höchstwert für die Laufzeit von Koordinatoraktivitäten. Der Wert wird für alle Verschachtelungsebenen berechnet. Die Zeit wird in Millisekunden gemessen. Bei Serviceklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Serviceklasse mit NONE definiert ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Arbeitsklasse nicht definiert ist.

Elementkennung
coord_act_lifetime_top

Elementtyp
Grenzwert

Tabelle 72. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wcstats	-
Statistiken	event_scstats	-

Verwendung

Dieses Element ist beim Beurteilen der Effektivität von Schwellenwerten für die Laufzeit von Aktivitäten hilfreich und somit auch eine wertvolle Entscheidungshilfe für die Konfiguration derartiger Schwellenwerte.

coord_act_rejected_total - Gesamtzahl zurückgewiesener Koordinatoraktivitäten (Monitorelement)

Die Gesamtzahl von Koordinatoraktivitäten auf allen Verschachtelungsebenen, die seit dem letzten Zurücksetzen nicht ausgeführt, sondern zurückgewiesen wurden. Dieser Zähler wird aktualisiert, wenn eine Aktivität durch einen vorbeugenden Schwellenwert oder die Arbeitsaktion zum Verhindern der Ausführung an der Ausführung gehindert wird. Bei Serviceklassen wird dieser Wert beim Beenden von Aktivitäten aktualisiert. Bei Workloads wird der Wert bei Abschluss der zugehörigen UOW mit den einzelnen Workloadvorkommen aktualisiert.

Elementkennung
coord_act_rejected_total

Elementtyp
Zähler

Tabelle 73. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wlstats	-

Verwendung

Mit diesem Element kann ermittelt werden, ob vorbeugende Schwellenwerte und Arbeitsaktionen, die die Ausführung verhindern, effektiv eingesetzt werden oder zu einschränkend wirken.

coord_partition_num - Nummer der Koordinatorpartition (Monitorelement)

Die Partitionsnummer der Koordinatorpartition der Aktivität.

Elementkennung
coord_partition_num

Elementtyp
Information

Tabelle 74. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Dieses Element ermöglicht es, bei Aktivitäten, die über Datensätze auf anderen Partitionen als der Koordinatorpartition verfügen, die Koordinatorpartition zu ermitteln.

cost_estimate_top - Maximaler Wert für Kostenschätzung (Monitorelement)

Der Höchstwert für die geschätzten Kosten von DML-Aktivitäten auf allen Verschachtelungsebenen innerhalb einer Serviceunterklasse oder Arbeitsklasse. Bei Serviceunterklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Serviceunterklasse mit NONE definiert ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Arbeitsklasse nicht definiert ist.

Elementkennung

cost_estimate_top

Elementtyp

Grenzwert

Tabelle 75. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Dieses Element gibt Aufschluss über den Höchstwert für die geschätzten Kosten von DML-Aktivitäten, der auf einer Partition innerhalb des erfassten Zeitintervalls für eine Serviceklasse oder Arbeitsklasse erreicht wurde.

coord_act_lifetime_avg - Durchschnittswert für die Laufzeit von Koordinatoraktivitäten (Monitorelement)

Das arithmetische Mittel der Laufzeit von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse oder Arbeitsklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Bei Serviceunterklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die Serviceunterklasse auf NONE gesetzt ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die Arbeitsklasse nicht definiert ist. Die Zeit wird in Millisekunden gemessen.

Elementkennung

coord_act_lifetime_avg

Elementtyp
Information

Tabelle 76. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Mit dieser Statistik können Sie das arithmetische Mittel der Laufzeit von Koordinatoraktivitäten ermitteln, die einer abgeschlossenen oder abgebrochenen Serviceunterklasse oder Arbeitsklasse zugeordnet wurden.

Ferner kann mit Hilfe dieser Statistik festgestellt werden, ob die Histogrammschablone für das Histogramm der Laufzeit von Aktivitäten geeignet ist. Mit dem Histogramm der Laufzeit von Aktivitäten kann die durchschnittliche Laufzeit von Aktivitäten berechnet werden. Vergleichen Sie den berechneten Durchschnittswert mit diesem Monitorelement. Wenn der berechnete Durchschnittswert vom tatsächlichen Durchschnittswert abweicht, der von diesem Monitorelement zurückgemeldet wird, sollten Sie eine Änderung der Histogrammschablone für das Histogramm der Laufzeit von Aktivitäten in Erwägung ziehen. Verwenden Sie dazu eine Gruppe von Binwerten, die für Ihre Daten besser geeignet sind.

coord_act_queue_time_avg - Durchschnittliche Warteschlangenzeit für Koordinatoraktivitäten (Monitorelement)

Das arithmetische Mittel der Warteschlangenzeit für Koordinatoraktivitäten auf der Verschachtelungsebene 0, die der Serviceunterklasse oder Arbeitsklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Bei Serviceunterklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die Serviceunterklasse auf NONE gesetzt ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die Arbeitsklasse nicht definiert ist. Die Zeit wird in Millisekunden gemessen. Der Wert 0 gibt an, dass die Serviceunterklasse seit dem letzten Zurücksetzen abgeschlossen oder abgebrochen wurde. Gibt -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA der Serviceklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen.

Elementkennung
coord_act_queue_time_avg

Elementtyp
Information

Tabelle 77. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Mit dieser Statistik können Sie das arithmetische Mittel der Warteschlangenzeit für Koordinatoraktivitäten ermitteln, die einer abgeschlossenen oder abgebrochenen Serviceunterklasse oder Arbeitsklasse zugeordnet wurden. Ferner kann mit Hilfe dieser Statistik festgestellt werden, ob die Histogrammschablone für das Histogramm der Warteschlangenzeit von Aktivitäten geeignet ist. Mit dem Histogramm der Warteschlangenzeit von Aktivitäten kann die durchschnittliche Warteschlangenzeit von Aktivitäten berechnet werden. Vergleichen Sie den berechneten Durchschnittswert mit diesem Monitorelement. Wenn der berechnete Durchschnittswert vom tatsächlichen Durchschnittswert abweicht, der von diesem Monitorelement zurückgemeldet wird, sollten Sie eine Änderung der Histogrammschablone für das Histogramm der Warteschlangenzeit von Aktivitäten in Erwägung ziehen. Verwenden Sie dazu eine Gruppe von Binwerten, die für Ihre Daten besser geeignet sind.

coord_act_exec_time_avg - Durchschnittswert für die Ausführungszeit von Koordinatoraktivitäten (Monitorelement)

Das arithmetische Mittel der Ausführungszeiten von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die dieser Serviceunterklasse oder Arbeitsklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Bei Serviceunterklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die Serviceunterklasse auf NONE gesetzt ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die Arbeitsklasse nicht definiert ist. Die Zeit wird in Millisekunden gemessen.

Elementkennung

coord_act_exec_time_avg

Elementtyp

Information

Tabelle 78. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Mit dieser Statistik können Sie das arithmetische Mittel der Ausführungszeit von Koordinatoraktivitäten ermitteln, die einer abgeschlossenen oder abgebrochenen Serviceunterklasse oder Arbeitsklasse zugeordnet wurden.

Ferner kann mit Hilfe dieser Statistik festgestellt werden, ob die Histogrammschablone für das Histogramm der Ausführungszeit von Aktivitäten geeignet ist. Mit dem Histogramm der Ausführungszeit von Aktivitäten kann die durchschnittliche Ausführungszeit von Aktivitäten berechnet werden. Vergleichen Sie den berechneten Durchschnittswert mit diesem Monitorelement. Wenn der berechnete Durchschnittswert vom tatsächlichen Durchschnittswert abweicht, der von diesem Monitorelement zurückgemeldet wird, sollten Sie eine Änderung der Histogrammschablone für das Histogramm der Ausführungszeit von Aktivitäten in Erwägung ziehen. Verwenden Sie dazu eine Gruppe von Binwerten, die für Ihre Daten besser geeignet sind.

request_exec_time_avg - Durchschnittswert für die Ausführungszeit von Anforderungen (Monitorelement)

Das arithmetische Mittel der Ausführungszeiten von Anforderungen, die dieser Serviceunterklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Dieses Monitorelement gibt -1 zurück, wenn COLLECT AGGREGATE REQUEST DATA für die Serviceunterklasse auf NONE gesetzt ist. Die Zeit wird in Millisekunden gemessen.

Elementkennung

request_exec_time_avg

Elementtyp

Information

Tabelle 79. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-

Verwendung

Mit dieser Statistik lässt sich schnell die durchschnittliche Zeit feststellen, die für die Verarbeitung der einzelnen Anforderungen auf einer Datenbankpartition in dieser Serviceunterklasse benötigt wird.

Ferner kann mit Hilfe dieses Durchschnittswerts ermittelt werden, ob die Histogrammschablone für das Histogramm der Ausführungszeit von Anforderungen geeignet ist. Mit dem Histogramm der Ausführungszeit von Anforderungen kann die durchschnittliche Ausführungszeit von Anforderungen berechnet werden. Vergleichen Sie den berechneten Durchschnittswert mit diesem Monitorelement. Wenn der berechnete Durchschnittswert vom tatsächlichen Durchschnittswert abweicht, der von diesem Monitorelement zurückgemeldet wird, sollten Sie eine Änderung der Histogrammschablone für das Histogramm der Ausführungszeit von Anforderungen in Erwägung ziehen. Verwenden Sie dazu eine Gruppe von Binwerten, die für Ihre Daten besser geeignet sind.

coord_act_est_cost_avg - Durchschnittswert für den geschätzten Aufwand für Koordinatoraktivitäten (Monitorelement)

Das arithmetische Mittel des geschätzten Aufwands für DML-Koordinatoraktivitäten auf der Verschachtelungsebene 0, die dieser Serviceunterklasse oder Arbeitsklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Bei Serviceunterklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die Serviceunterklasse auf NONE oder BASE gesetzt ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die Arbeitsklasse nicht definiert ist. Die Zeit wird in Millisekunden gemessen.

Elementkennung

coord_act_est_cost_avg

Elementtyp

Information

Tabelle 80. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Mit dieser Statistik können Sie das arithmetische Mittel des geschätzten Aufwands für DML-Koordinatoraktivitäten auf der Verschachtelungsebene 0 ermitteln, die einer abgeschlossenen oder abgebrochenen Serviceunterklasse oder Arbeitsklasse seit dem letzten Zurücksetzen zugeordnet wurden.

Ferner kann mit Hilfe dieses Durchschnittswerts festgestellt werden, ob die Histogrammschablone für das Histogramm des geschätzten Aufwands für Aktivitäten geeignet ist. Mit dem Histogramm des geschätzten Aufwands für Aktivitäten kann der Durchschnittswert für den geschätzten Aufwand für Aktivitäten berechnet werden. Vergleichen Sie den berechneten Durchschnittswert mit diesem Monitorelement. Wenn der berechnete Durchschnittswert vom tatsächlichen Durchschnittswert abweicht, der von diesem Monitorelement zurückgemeldet wird, sollten Sie eine Änderung der Histogrammschablone für das Histogramm des geschätzten Aufwands für Aktivitäten in Erwägung ziehen. Verwenden Sie dazu eine Gruppe von Binwerten, die für Ihre Daten besser geeignet sind.

coord_act_interarrival_time_avg - Durchschnittswert für die Ankunftszeit von Koordinatoraktivitäten (Monitorelement)

Das arithmetische Mittel der Zeit zwischen der Ankunft von Koordinatoraktivitäten auf der Verschachtelungsebene 0, die dieser Serviceunterklasse oder Arbeitsklasse seit dem letzten Zurücksetzen zugeordnet wurden. Wenn für den intern aufgezeichneten Durchschnittswert ein Überlauf stattgefunden hat, wird der Wert -2 zurückgegeben. Bei Serviceunterklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die Serviceunterklasse auf NONE oder BASE gesetzt ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die Arbeitsklasse nicht definiert ist. Die Zeit wird in Millisekunden gemessen.

Elementkennung

coord_act_interarrival_time_avg

Elementtyp

Information

Tabelle 81. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Mit dieser Statistik können Sie das arithmetische Mittel der Zeit zwischen der Ankunft von Koordinatoraktivitäten auf der Verschachtelungsebene 0 ermitteln, die der Serviceunterklasse oder Arbeitsklasse zugeordnet wurden.

Mit der Zeit zwischen der Ankunft von Aktivitäten kann die Ankunftsrate ermittelt werden, die den Kehrwert der Zeit zwischen der Ankunft von Aktivitäten darstellt. Ferner kann mit Hilfe dieses Durchschnittswerts festgestellt werden, ob die Histogrammschablone für das Histogramm der Zeit zwischen der Ankunft von Aktivitäten geeignet ist. Mit dem Histogramm der Zeit zwischen der Ankunft von Aktivitäten kann die durchschnittliche Zeit zwischen der Ankunft von Aktivitäten berechnet werden. Vergleichen Sie den berechneten Durchschnittswert mit diesem Monitorelement. Wenn der berechnete Durchschnittswert vom tatsächlichen Durchschnittswert abweicht, der von diesem Monitorelement zurückgegeben wird, sollten Sie eine Änderung der Histogrammschablone für das Histogramm der Zeit zwischen der Ankunft von Aktivitäten in Erwägung ziehen. Verwenden Sie dazu eine Gruppe von Binwerten, die für Ihre Daten besser geeignet sind.

db_work_action_set_id - Set-ID für Datenbankarbeitsaktionen (Monitorelement)

Wurde die Aktivität innerhalb einer Arbeitsklasse im Datenbankbereich kategorisiert, zeigt dieses Monitorelement die ID des Arbeitsaktionssets an, die dem Arbeitsklassenset, zu dem die Arbeitsklasse gehört, zugeordnet ist. Andernfalls zeigt dieses Monitorelement 0 an.

Elementkennung

db_work_action_set_id

Elementtyp

Information

Tabelle 82. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element kann in Verbindung mit dem Element **db_work_class_id** verwendet werden, um die Datenbankarbeitsklasse der Aktivität, soweit vorhanden, eindeutig zu identifizieren.

db_work_class_id - Datenbankarbeitsklassen-ID (Monitorelement)

Wurde die Aktivität innerhalb einer Arbeitsklasse im Datenbankbereich kategorisiert, zeigt dieses Monitorelement die ID der Arbeitsklasse an. Andernfalls zeigt dieses Monitorelement 0 an.

Elementkennung

db_work_class_id

Elementtyp

Information

Tabelle 83. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element kann in Verbindung mit dem Element **db_work_action_set_id** verwendet werden, um die Datenbankarbeitsklasse der Aktivität, soweit vorhanden, eindeutig zu identifizieren.

histogram_type - Histogrammtyp (Monitorelement)

Der Typ des Histogramms im Zeichenfolgeformat.

Es gibt sechs verschiedene Typen von Histogrammen.

CoordActQueueTime

Ein Histogramm der Zeit, die nicht verschachtelte Aktivitäten in einer Warteschlange (z. B. in einer Schwellenwertwarteschlange) verbringen, gemessen auf der Koordinatorpartition.

CoordActExecTime

Ein Histogramm der Ausführungszeit nicht verschachtelter Aktivitäten auf der Koordinatorpartition. Die Ausführungszeit umfasst nicht die für die Initialisierung benötigte bzw. die in Warteschlangen verbrachte Zeit. Bei Cursors umfasst die Ausführungszeit nur die Zeit für das Öffnen, Abrufen und Schließen von Anforderungen.

CoordActLifetime

Ein Histogramm der abgelaufenen Laufzeit nicht verschachtelter Aktivitäten, gemessen auf der Koordinatorpartition von dem Zeitpunkt an, zu dem die Aktivität im System ankommt, bis zu dem Zeitpunkt, zu dem die Ausführung der Aktivität abgeschlossen wird. Zur Laufzeit gehört auch die Zeit, die eine Aktivität für die Initialisierung und die Ausführung benötigt, sowie die in Warteschlangen verbrachte Zeit.

CoordActInterArrivalTime

Ein Histogramm des Zeitintervalls zwischen der Ankunft nicht verschachtelter Koordinatoraktivitäten.

CoordActEstCost

Ein Histogramm der geschätzten Kosten nicht verschachtelter DML-Aktivitäten.

ReqExecTime

Ein Histogramm der Ausführungszeiten von Anforderungen. Es enthält alle Anforderungen auf Koordinator- und Nicht-Koordinatorpartitionen, einschließlich der Anforderungen, die keiner Aktivität zugeordnet sind.

Elementkennung

histogram_type

Elementtyp

Information

Tabelle 84. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-

Verwendung

Mit Hilfe dieses Elements kann der Typ eines Histogramms bestimmt werden. Zu einem Statistikdatensatz können mehrere Histogramme gehören, jedoch jeweils nur ein Histogramm eines bestimmten Typs.

last_wlm_reset - Zeitpunkt des letzten Zurücksetzens (Monitorelement)

Dieses Element zeigt in Form einer lokalen Zeitmarke den Zeitpunkt an, zu dem der letzte Ereignisdatensatz für Statistikdaten dieses Typs erstellt wurde.

Elementkennung

last_wlm_reset

Elementtyp

Information

Tabelle 85. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wlstats	-
Statistiken	event_wcstats	-
Statistiken	event_qstats	-

Verwendung

Mit den Monitorelementen **wlm_last_reset** und **statistics_timestamp** lässt sich der Zeitraum ermitteln, in dem die Statistikdaten im Statistikdatensatz eines Ereignismonitors erfasst wurden. Das Erfassungsintervall beginnt mit der Zeit von **wlm_last_reset** und endet mit **statistics_timestamp**.

num_threshold_violations - Anzahl der Schwellenwertverstöße (Monitorelement)

Die Anzahl der Schwellenwertverstöße, die in der Datenbank seit der letzten Aktivierung aufgetreten sind.

Elementkennung

num_threshold_violations

Elementtyp

Zähler

Tabelle 86. Informationen zur Momentaufnahmeüberwachung

Ebene der Momentaufnahme	Logische Datengruppierung	Monitorschalter
Datenbank	dbase	Einfach

Bei der Momentaufnahmeüberwachung kann dieser Zähler zurückgesetzt werden.

Tabelle 87. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Datenbank	event_db	-

Verwendung

Mit diesem Element lässt sich feststellen, ob Schwellenwerte für eine bestimmte Anwendung sinnvoll definiert sind und ob zu viele Schwellenwertverstöße auftreten.

number_in_bin - Anzahl in Bin (Monitorelement)

Dieses Element enthält den Zähler für die Anzahl der Aktivitäten bzw. Anforderungen, die in den von dem Histogramm-Bin erfassten Zeitraum fallen.

Elementkennung

number_in_bin

Elementtyp

Information

Tabelle 88. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-

Verwendung

Dieses Element gibt Aufschluss über die Höhe eines Bins in dem Histogramm.

parent_activity_id - ID der übergeordneten Aktivität (Monitorelement)

Die eindeutige ID innerhalb der UOW der übergeordneten Aktivität, die der der Aktivität übergeordneten Aktivität zugeordnet ist. Ist keine übergeordnete Aktivität vorhanden, ist der Wert dieses Monitorelements 0.

Elementkennung

parent_activity_id

Elementtyp

Information

Tabelle 89. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Mit diesem Element lässt sich in Verbindung mit den Elementen **parent_uow_id** und **appl_id** die übergeordnete Aktivität der im Aktivitätsdatensatz beschriebenen Aktivität eindeutig identifizieren.

parent_uow_id - UOW-ID der übergeordneten Aktivität (Monitorelement)

Die innerhalb einer Anwendungskennung eindeutige Kennung der UOW (Unit of Work). Die ID der UOW, in der die übergeordnete Aktivität der Aktivität ihren Ursprung hat. Ist keine übergeordnete Aktivität vorhanden, ist der Wert 0.

Elementkennung

parent_uow_id

Elementtyp
Information

Tabelle 90. Informationen zur Ereignisüberwachung

Reignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Mit diesem Element lässt sich in Verbindung mit den Elementen **parent_activity_id** und **appl_id** die übergeordnete Aktivität der im Aktivitätsdatensatz beschriebenen Aktivität eindeutig identifizieren.

prep_time - Vorbereitungszeit (Monitorelement)

Die für die Vorbereitung einer SQL-Anweisung erforderliche Zeit (in Millisekunden), wenn es sich bei der Aktivität um eine SQL-Anweisung handelt.

Elementkennung
prep_time

Elementtyp
Zeit

Tabelle 91. Informationen zur Ereignisüberwachung

Reignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Mit diesem Element kann bei SQL-Aktivitäten ermittelt werden, wie viel Zeit der Gesamtlaufzeit einer Aktivität für die Vorbereitung der SQL-Anwendung benötigt wurde.

queue_assignments_total - Gesamtzahl der Warteschlangenzuordnungen (Monitorelement)

Die Anzahl der Verbindungen bzw. Aktivitäten, die der Schwellenwertwarteschlange seit dem letzten Zurücksetzen zugeordnet wurden.

Elementkennung
queue_assignments_total

Elementtyp
Zähler

Tabelle 92. Informationen zur Ereignisüberwachung

Reignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-

Verwendung

Dieses Element gibt Aufschluss über die Anzahl der Aktivitäten bzw. Verbindungen, die innerhalb eines bestimmten Zeitraums, der durch das Intervall für die Erfassung von Statistikdaten vorgegeben wird, in diese Warteschlange eingereicht

wurden. Dieses Element ist hilfreich bei der Beurteilung der Effizienz von Schwellenwerten für die Warteschlangensteuerung.

queue_size_top - Maximaler Wert für Warteschlangenlänge (Monitorelement)

Der Höchstwert für die Warteschlangenlänge, der seit dem letzten Zurücksetzen erreicht wurde.

Elementkennung

queue_size_top

Elementtyp

Grenzwert

Tabelle 93. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-

Verwendung

Dieses Element gibt Aufschluss über die Effizienz von Schwellenwerten für die Warteschlangensteuerung und ermöglicht es zu erkennen, ob Warteschlangen zu umfangreich werden.

queue_time_total - Gesamtwarteschlangenzeit (Monitorelement)

Die Zeit, die alle Verbindungen bzw. Aktivitäten, die seit dem letzten Zurücksetzen in die Warteschlange gestellt wurden, insgesamt in der Warteschlange verbraucht haben. Die Zeit wird in Millisekunden gemessen.

Elementkennung

queue_time_total

Elementtyp

Zähler

Tabelle 94. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-

Verwendung

Dieses Element gibt Aufschluss über die Effizienz von Schwellenwerten für die Warteschlangensteuerung und ermöglicht es zu erkennen, ob Warteschlangen zu umfangreich werden.

rows_fetched - Abgerufene Zeilen (Monitorelement)

Die Anzahl der aus der Tabelle gelesenen Zeilen.

Dieses Monitorelement ist ein Alias des Monitorelements **rows_read**.

Anmerkung: Dieses Monitorelement meldet nur die Werte für die Datenbankpartition, für die diese Angaben aufgezeichnet werden. Bei DPF-Systemen (Database Partitioning Feature) spiegeln diese Werte möglicherweise nicht die korrekten Summen für die gesamte Aktivität wider.

Elementkennung

rows_fetched

Elementtyp

Zähler

Tabelle 95. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	Anweisung

Verwendung

Nähere Informationen können Sie dem Abschnitt zum Monitorelement **rows_read** entnehmen.

rows_modified - Modifizierte Zeilen (Monitorelement)

Die Anzahl der eingefügten, aktualisierten und gelöschten Zeilen.

Dieses Monitorelement ist ein Alias des Monitorelements **rows_written**.

Anmerkung: Dieses Monitorelement meldet nur die Werte für die Datenbankpartition, für die dieser Datensatz aufgezeichnet wird. Bei DPF-Systemen (Database Partitioning Feature) spiegeln diese Werte möglicherweise nicht die korrekten Summen für die gesamte Aktivität wider.

Elementkennung

rows_modified

Elementtyp

Zähler

Tabelle 96. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	Anweisung

Verwendung

Nähere Informationen können Sie dem Abschnitt zum Monitorelement **rows_written** entnehmen.

rows_returned - Zurückgegebene Zeilen (Monitorelement)

Die Anzahl der Zeilen, die ausgewählt und an die Anwendung zurückgegeben wurden. Dieses Element weist einen Wert 0 für Aktivitätsteildatensätze auf (z. B. wenn eine Aktivität erfasst wird, während sie noch ausgeführt wird, oder wenn ein vollständiger Aktivitätsdatensatz auf Grund von Speicherbeschränkungen nicht in den Ereignismonitor geschrieben werden konnte).

Dieses Monitorelement ist ein Aliasname des Monitorelements **fetch_count**.

Elementkennung
rows_returned

Elementtyp
Zähler

Tabelle 97. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Sie können dieses Element verwenden, um Schwellenwerte für die an die Anwendung zurückgegebenen Zeilen zu ermitteln oder um zu überprüfen, ob ein solcher Schwellenwert korrekt definiert ist und seinen Zweck erfüllt.

rows_returned_top - Maximaler Wert für zurückgegebene Zeilen (Monitorelement)

Der Höchstwert für die für DML-Aktivitäten auf allen Verschachtelungsebenen innerhalb einer Service- oder Arbeitsklasse zurückgegebenen Zeilen. Bei Serviceklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Serviceklasse mit NONE definiert ist. Bei Arbeitsklassen gibt dieses Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Arbeitsklasse nicht definiert ist.

Elementkennung
rows_returned_top

Elementtyp
Grenzwert

Tabelle 98. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Dieses Element gibt Aufschluss über den Höchstwert für zurückgegebene Zeilen bei DML-Aktivitäten, der auf einer Partition innerhalb des erfassten Zeitintervalls für eine Serviceklasse oder Arbeitsklasse erreicht wurde.

sc_work_action_set_id - Set-ID für Serviceklassenarbeitsaktionen (Monitorelement)

Wurde die Aktivität innerhalb einer Arbeitsklasse im Serviceklassenbereich kategorisiert, zeigt dieses Monitorelement die ID des Arbeitsaktionssets an, die dem Arbeitsklassenset, zu dem die Arbeitsklasse gehört, zugeordnet ist. Andernfalls zeigt dieses Monitorelement 0 an.

Elementkennung
sc_work_action_set_id

Elementtyp
Information

Tabelle 99. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element kann in Verbindung mit dem Element **sc_work_class_id** verwendet werden, um die Serviceklassenarbeitsklasse der Aktivität, soweit vorhanden, eindeutig zu identifizieren.

sc_work_class_id - Arbeitsklassen-ID für Serviceklassen (Monitorelement)

Wurde die Aktivität innerhalb einer Arbeitsklasse im Serviceklassenbereich kategorisiert, zeigt dieses Monitorelement die ID der Arbeitsklasse an, die dieser Aktivität zugeordnet ist. Andernfalls zeigt dieses Monitorelement 0 an.

Elementkennung
sc_work_class_id

Elementtyp
Information

Tabelle 100. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element kann in Verbindung mit dem Element **sc_work_action_set_id** verwendet werden, um die Serviceklassenarbeitsklasse der Aktivität, soweit vorhanden, eindeutig zu identifizieren.

section_env - Abschnittsumgebung (Monitorelement)

Eine interne Kennung, die Abschnittsdetails zu einer Aktivität angibt.

Elementkennung
section_env

Elementtyp
Information

Tabelle 101. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activitystmt	-

Verwendung

Dieses Element ist für zukünftige IBM Tools vorgesehen, die Abschnittsinformationen zu der in diesem Datensatz beschriebenen Aktivität extrahieren.

service_class_id - Serviceklassen-ID (Monitorelement)

Eindeutige ID der Serviceklasse. Kann für Joins mit der Histogramm-Bin-Tabelle verwendet werden.

Elementkennung

service_class_id

Elementtyp

Information

Tabelle 102. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-
Statistiken	event_scstats	-

Verwendung

Mit diesem Element können in Verbindung mit den Monitorelementen **statistics_timestamp** und **partition_number** Datensätze mit Histogramm-Bins mit Statistikdatensätzen für Serviceklassen verknüpft werden.

service_subclass_name - Name der Serviceunterklasse (Monitorelement)

Der Name der Serviceunterklasse, auf die sich dieser Aktivitätsdatensatz oder Statistikdatensatz bezieht.

Elementkennung

service_subclass_name

Elementtyp

Information

Tabelle 103. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-
Statistiken	event_scstats	-
Statistiken	event_qstats	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Aktivitätselementen eine Analyse des Verhaltens einer Aktivität und in Verbindung mit anderen Statistik-elementen eine Analyse einer Serviceklasse oder Schwellenwertwarteschlange.

service_superclass_name - Name der Servicesuperklasse (Monitorelement)

Der Name der Servicesuperklasse, auf die sich dieser Aktivitätsdatensatz oder Statistikdatensatz bezieht.

Elementkennung

service_superclass_name

Elementtyp

Information

Tabelle 104. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-
Statistiken	event_scstats	-
Statistiken	event_qstats	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Aktivitätselementen eine Analyse des Verhaltens einer Aktivität und in Verbindung mit anderen Statistik-elementen eine Analyse einer Serviceklasse oder Schwellenwertwarteschlange.

statistics_timestamp - Zeitmarke für Statistikdatensatz (Monitorelement)

Der Zeitpunkt, zu dem der Statistikdatensatz generiert wurde.

Elementkennung

statistics_timestamp

Elementtyp

Information

Tabelle 105. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wlstats	-
Statistiken	event_wcstats	-
Statistiken	event_qstats	-
Statistiken	event_histogrambin	-

Verwendung

Mit Hilfe dieses Elements kann ermittelt werden, wann der Statistikdatensatz erstellt wurde.

Verwenden Sie dieses Element in Verbindung mit dem Element **last_wlm_reset**, um das Zeitintervall zu bestimmen, in dem die Statistikdaten in diesem Statistikdatensatz generiert wurden.

Mit Hilfe dieses Monitorelements können darüber hinaus auch alle Statistikdatensätze zusammengefasst werden, die in demselben Erfassungsintervall generiert wurden.

temp_tablespace_top - Maximaler Wert für Tabellenbereiche für temporäre Tabellen (Monitorelement)

Der Höchstwert für die Belegung des Speicherplatzes für Tabellenbereiche für temporäre Tabellen durch DML-Aktivitäten auf allen Verschachtelungsebenen innerhalb einer Service- oder Arbeitsklasse. Bei Serviceklassen gibt dieses Monitorelement -1 zurück, wenn COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Serviceklasse mit NONE definiert ist. Bei Arbeitsklassen gibt dieses

Monitorelement -1 zurück, wenn die Arbeitsaktion COLLECT AGGREGATE ACTIVITY DATA für die jeweilige Arbeitsklasse nicht definiert ist.

Elementkennung

temp_tablespace_top

Elementtyp

Grenzwert

Tabelle 106. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_scstats	-
Statistiken	event_wcstats	-

Verwendung

Dieses Element gibt Aufschluss über den Höchstwert für die Speicherplatzbelegung durch Systemtabellenbereiche für temporäre Tabellen für DML-Aktivitäten, der auf einer Partition innerhalb des erfassten Zeitintervalls für eine Service- oder Arbeitsklasse erreicht wurde.

Dieses Element wird nur von Aktivitäten aktualisiert, auf die ein Schwellenwert für den Speicherplatz von Tabellenbereichen für temporäre Tabellen angewendet wird. Wird kein Schwellenwert für den Speicherplatz von Tabellenbereichen für temporäre Tabellen auf eine Aktivität angewendet, wird der Wert 0 zurückgegeben. Ist für die Service- oder Arbeitsklasse keine kumulative Datenerfassung für Aktivitäten aktiviert, wird der Wert -1 zurückgegeben.

threshold_action - Schwellenwertaktion (Monitorelement)

Die für den Schwellenwert vorgesehene Aktion, auf den sich dieser Datensatz für Schwellenwertverstöße bezieht. Mögliche Aktionen sind Stoppen und Fortfahren.

Elementkennung

threshold_action

Elementtyp

Information

Tabelle 107. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Mit diesem Element kann festgestellt werden, ob eine Aktivität, die gegen den Schwellenwert verstoßen hat, bei Auftreten des Verstoßes gestoppt wurde, oder ob die Ausführung fortgesetzt werden konnte. Wenn die Aktivität gestoppt wurde, erhält die Anwendung, die die Aktivität übergeben hat, einen SQL-Fehler (SQL4712N).

threshold_domain - Schwellenwertdomäne (Monitorelement)

Die Domäne des für diese Warteschlange verantwortlichen Schwellenwerts.

Mögliche Werte:

- Datenbank
- Arbeitsaktionsset
- Servicesuperklasse
- Serviceunterklasse
- Auslastung

Elementkennung

threshold_domain

Elementtyp

Information

Tabelle 108. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-

Verwendung

Mit diesem Element lassen sich die Warteschlangenstatistiken für Schwellenwerte voneinander abgrenzen, die dasselbe Vergleichselement, aber unterschiedliche Domänen aufweisen.

threshold_maxvalue - Maximaler Wert für Schwellenwert (Monitorelement)

Bei Schwellenwerten, die sich nicht auf Warteschlangen beziehen, stellt dieses Monitorelement den Wert dar, dessen Überschreitung den Schwellenwertverstoß verursacht hat. Bei Schwellenwerten für die Warteschlangensteuerung gibt dieses Monitorelement die Ebene des gemeinsamen Zugriffs an, die den Schwellenwertverstoß verursacht hat. Die Ebene des gemeinsamen Zugriffs, die den Verstoß gegen den Schwellenwert für die Warteschlangensteuerung verursacht hat, ergibt sich aus der Summe der Monitorelemente **threshold_maxvalue** und **threshold_queuesize**.

Elementkennung

threshold_maxvalue

Elementtyp

Information

Tabelle 109. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Bei Aktivitätsschwellenwerten liefert dieses Element einen Protokolldatensatz, der den maximalen Wert für den Schwellenwert zum Zeitpunkt des Schwellenwertverstoßes angibt. Dies ist hilfreich, wenn sich der maximale Wert für den Schwellenwert seit dem Schwellenwertverstoß geändert hat und der alte Wert in der Sicht SYSCAT.THRESHOLDS nicht mehr angezeigt wird.

threshold_name - Name des Schwellenwerts (Monitorelement)

Der eindeutige Name des für diese Warteschlange verantwortlichen Schwellenwerts.

Elementkennung
threshold_name

Elementtyp
Information

Tabelle 110. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-

Verwendung

Mit diesem Element lässt sich der Schwellenwert für Warteschlangensteuerung, auf den sich die Statistikdaten dieses Datensatzes beziehen, eindeutig ermitteln.

threshold_predicate - Vergleichselement für Schwellenwert (Monitorelement)

Gibt den Typ des Schwellenwerts an, gegen den verstoßen bzw. für den Statistikdaten erfasst wurden.

Elementkennung
threshold_predicate

Elementtyp
Information

Tabelle 111. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-
Statistiken	event_qstats	-

Verwendung

Dieses Monitorelement ermöglicht in Verbindung mit anderen Monitorelementen für Statistiken oder Schwellenwertverstöße eine Analyse des Schwellenwertverstoßes.

threshold_queuesize - Größe der Schwellenwertwarteschlange (Monitorelement)

Die Größe der Warteschlange für einen Schwellenwert für die Warteschlangensteuerung. Ein Versuch, diese Größe zu überschreiten, führt zu einem Schwellenwertverstoß. Für Schwellenwerte, die nicht auf Warteschlangen bezogen sind, hat dieses Monitorelement den Wert 0.

Elementkennung
threshold_queuesize

Elementtyp
Information

Tabelle 112. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Mit diesem Element lässt sich die Anzahl der Aktivitäten bzw. Verbindungen in der Warteschlange für diesen Schwellenwert bestimmen, die sich zum Zeitpunkt des Schwellenwertverstoßes in der Warteschlange befanden.

thresholdid - Schwellenwert-ID (Monitorelement)

Gibt den Schwellenwert an, auf den sich ein Datensatz für Schwellenwertverstöße bezieht oder für den Warteschlangenstatistikdaten erfasst wurden.

Elementkennung

thresholdid

Elementtyp

Information

Tabelle 113. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-
Statistiken	event_qstats	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse einer Warteschlange mit Schwellenwertsteuerung bzw. der Aktivität, die gegen einen Schwellenwert verstoßen hat.

time_completed - Abschlusszeitpunkt (Monitorelement)

Der Zeitpunkt, zu dem die Ausführung der von dem Aktivitätsdatensatz beschriebenen Aktivität abgeschlossen wurde. Bei diesem Element handelt es sich um eine lokale Zeitmarke.

Elementkennung

time_completed

Elementtyp

Information

Tabelle 114. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse des Verhaltens einer Aktivität.

time_created - Erstellungszeitpunkt (Monitorelement)

Der Zeitpunkt, zu dem die von dem Aktivitätsdatensatz beschriebene Aktivität von einem Benutzer übergeben wurde. Bei diesem Element handelt es sich um eine lokale Zeitmarke.

Elementkennung
time_created

Elementtyp
Information

Tabelle 115. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse des Verhaltens einer Aktivität.

time_of_violation - Zeitpunkt des Verstoßes (Monitorelement)

Der Zeitpunkt, zu dem der in diesem Datensatz beschriebene Schwellenwertverstoß aufgetreten ist. Bei diesem Element handelt es sich um eine lokale Zeitmarke.

Elementkennung
time_of_violation

Elementtyp
Information

Tabelle 116. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Monitorelementen für Schwellenwertverstöße die Analyse eines Schwellenwertverstoßes.

time_started - Startzeitpunkt (Monitorelement)

Der Zeitpunkt, zu dem die Ausführung der von dem Aktivitätsdatensatz beschriebenen Aktivität gestartet wurde. Bei diesem Element handelt es sich um eine lokale Zeitmarke.

Elementkennung
time_started

Elementtyp
Information

Tabelle 117. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse des Verhaltens einer Aktivität.

bottom - Obere Grenze von Histogramm-Bin (Monitorelement)

Die Obergrenze (Teil des Bereichs) des für ein Histogramm-Bin festgelegten Bereichs. Der Wert dieses Monitorelements stellt gleichzeitig die Untergrenze (nicht Teil des Bereichs) des Bereichs des nachfolgenden Histogramm-Bins dar, soweit vorhanden.

Elementkennung

top

Elementtyp

Information

Tabelle 118. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-

Verwendung

Mit diesem Element kann in Verbindung mit dem entsprechenden Element **bottom** der Bereich für ein Bin in einem Histogramm bestimmt werden.

uow_id - UOW-ID (Monitorelement)

Die UOW-ID (Unit of Work), auf die sich der Aktivitätsdatensatz bezieht. Die UOW-ID ist innerhalb einer Anwendungskennung eindeutig.

Elementkennung

uow_id

Elementtyp

Information

Tabelle 119. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-
Aktivitäten	event_activitystmt	-
Aktivitäten	event_activityvals	-
Schwellenwertverstöße	event_thresholdviolations	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse des Verhaltens einer Aktivität.

Sie können dieses Element auch in Verbindung mit den Monitorelementen **activity_id** und **appl_id** verwenden, um eine Aktivität eindeutig zu identifizieren.

wlo_completed_total - Gesamtzahl abgeschlossener Workloadvorkommen (Monitorelement)

Die Anzahl von abgeschlossenen Verarbeitungsprozessen, die seit dem letzten Zurücksetzen ausgeführt wurden.

Elementkennung

wlo_completed_total

Elementtyp

Zähler

Tabelle 120. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wlstats	-

Verwendung

Mit diesem Element kann festgestellt werden, wie viele Ausführungen eines bestimmten Verarbeitungsprozesses zur Auslastung des Systems beitragen.

work_action_set_id - ID des Arbeitsaktionssets (Monitorelement)

Die ID des Arbeitsaktionssets, auf das sich dieser Statistikdatensatz bezieht.

Elementkennung

work_action_set_id

Elementtyp

Information

Tabelle 121. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_histogrambin	-
Statistiken	event_wcstats	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Elementen für die Protokollierung von Aktivitäten eine Analyse des Verhaltens einer Aktivität und in Verbindung mit anderen Statistikelementen eine Analyse einer Arbeitsklasse.

work_action_set_name - Name des Arbeitsaktionssets (Monitorelement)

Der Name des Arbeitsaktionssets, auf den sich die als Teil dieses Ereignisses angezeigten Statistikdaten beziehen.

Elementkennung

work_action_set_name

Elementtyp

Information

Tabelle 122. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-
Statistiken	event_wcstats	-

Verwendung

Mit diesem Element lässt sich in Verbindung mit dem Element **work_class_name** die Arbeitsklasse eindeutig identifizieren, deren Statistikdaten in diesem Datensatz angezeigt werden. Ferner lässt sich die Arbeitsklasse eindeutig identifizieren, die Domäne der Schwellenwertwarteschlange ist, deren Statistikdaten in diesem Datensatz angezeigt werden.

work_class_id - Arbeitsklassen-ID (Monitorelement)

Die ID der Arbeitsklasse, auf die sich der Statistikdatensatz bezieht.

Elementkennung

work_class_id

Elementtyp

Information

Tabelle 123. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wcstats	-
Statistiken	event_histogrambin	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Statistikelementen eine Analyse einer Arbeitsklasse.

work_class_name - Name der Arbeitsklasse (Monitorelement)

Der Name der Arbeitsklasse, auf die sich die als Teil dieses Ereignisses angezeigten Statistikdaten beziehen.

Elementkennung

work_class_name

Elementtyp

Information

Tabelle 124. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_qstats	-
Statistiken	event_wcstats	-

Verwendung

Mit diesem Element lässt sich in Verbindung mit dem Element **work_action_set_name** die Arbeitsklasse eindeutig identifizieren, deren Statistikdaten in diesem Datensatz angezeigt werden.

Ferner lässt sich die Arbeitsklasse eindeutig identifizieren, die Domäne der Schwellenwertwarteschlange ist, deren Statistikdaten in diesem Datensatz angezeigt werden.

workload_id - Workload-ID (Monitorelement)

Die ID für die Workload, zu der dieser Statistikdatensatz für Aktivitäten, Anwendungen oder Workloads gehört.

Elementkennung

workload_id

Elementtyp

Information

Tabelle 125. Informationen zur Momentaufnahmeüberwachung

Ebene der Momentaufnahme	Logische Datengruppierung	Monitorschalter
Anwendung	appl_info	Einfach

Tabelle 126. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wlstats	-
Aktivitäten	event_activity	-

Verwendung

Mit dieser ID lässt sich die Workload, zu der dieser Statistikdatensatz für Aktivitäten, Anwendungen oder Workloads gehört, eindeutig identifizieren.

workload_name - Name der Workload (Monitorelement)

Der Name der Workload, auf den sich der Statistikdatensatz bezieht.

Elementkennung

workload_name

Elementtyp

Information

Tabelle 127. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Statistiken	event_wlstats	-

Verwendung

Dieses Element ermöglicht in Verbindung mit anderen Statistikelementen eine Analyse einer Workload.

workload_occurrence_id - ID des Workloadvorkommens (Monitorelement)

Die ID der Workloadausführung, zu der diese Aktivität gehört.

Elementkennung

workload_occurrence_id

Elementtyp

Tabelle 128. Informationen zur Ereignisüberwachung

Ereignistyp	Logische Datengruppierung	Monitorschalter
Aktivitäten	event_activity	-

Verwendung

Mit diesem Element kann das Workloadvorkommen identifiziert werden, das die Aktivität übergeben hat.

Kapitel 10. Befehle

SET WORKLOAD (Befehl)

Gibt die Workload an, der die Datenbankverbindung zugeordnet werden soll. Dieser Befehl kann vor dem Herstellen einer Verbindung zu einer Datenbank ausgegeben werden, oder er kann verwendet werden, um die aktuelle Verbindung erneut zuzuordnen, sobald die Verbindung hergestellt worden ist. Wenn die Verbindung hergestellt wurde, wird die erneute Workloadzuordnung am Anfang der nächsten UOW ausgeführt.

Berechtigung

Keine

Erforderliche Verbindung

Keine

Befehlssyntax

```
➤➤—SET WORKLOAD TO AUTOMATIC  
SYSDEFAULTADMWORKLOAD—➤➤
```

Befehlsparameter

AUTOMATIC

Gibt an, dass die Datenbankverbindung einer Workload zugeordnet wird, die durch die vom Server automatisch ausgeführten Workloadbewertung ausgewählt wird.

SYSDEFAULTADMWORKLOAD

Gibt an, dass die Datenbankverbindung der Workload SYSDEFAULTADMWORKLOAD zugeordnet wird. Hierdurch können Benutzer mit der Berechtigung DBADM oder SYSADM die normale Workloadbewertung umgehen.

Beispiele

Zum Zuordnen der Verbindung zur Workload SYSDEFAULTADMWORKLOAD:

```
SET WORKLOAD TO SYSDEFAULTADMWORKLOAD
```

Zum Zurücksetzen der Workloadzuordnung, so dass die Workload verwendet wird, die durch die vom Server ausgeführte Workloadbewertung ausgewählt wird:

```
SET WORKLOAD TO AUTOMATIC
```

Hinweise zur Verwendung

Wenn die Sitzungsberechtigungs-ID der Datenbankverbindung nicht über die Berechtigung SYSADM oder DBADM verfügt, kann die Verbindung der Workload SYSDEFAULTADMWORKLOAD nicht zugeordnet werden, und es wird ein Fehler zurückgegeben. Wenn der Befehl SET WORKLOAD TO SYSDEFAULTADMWORKLOAD vor dem Herstellen der Verbindung zu einer Datenbank abgesetzt wird, wird der Fehler nach dem Herstellen der Datenbankverbindung am Anfang der

ersten UOW (Unit of Work) zurückgegeben. Wenn der Befehl nach dem Herstellen der Datenbankverbindung abgesetzt wird, wird der Fehler am Anfang der nächsten UOW zurückgegeben - dem Zeitpunkt, an dem die erneute Workloadzuordnung erfolgen soll.

Kapitel 11. Konfigurationsparameter

wlm_collect_int - Workload-Management-Erfassungsintervall (Konfigurationsparameter)

Dieser Parameter gibt ein Erfassungs- und Zurücksetzungsintervall (in Minuten) für die Workload-Management-Statistik an.

In einem regelmäßigen zeitlichen Abstand in Minuten (x *wlm_collect_int*, wobei x der Wert des Parameters *wlm_collect_int* ist) werden alle Workload-Management-Statistikdaten gesammelt und an sämtliche aktiven Ereignismonitore für Statistiken gesendet; anschließend wird die Statistik zurückgesetzt. Wenn ein aktiver Ereignismonitor vorhanden ist, wird die Statistik in Abhängigkeit davon, wie er erstellt wurde, entweder in die Datei oder in eine Tabelle geschrieben. Wenn kein aktiver Ereignismotor vorhanden ist, wird die Statistik nur zurückgesetzt, aber nicht erfasst.

Der Erfassungs- und Zurücksetzungsprozess wird von der Katalogpartition eingeleitet. Der Parameter *wlm_collect_int* muss in der Katalogpartition angegeben werden. Er wird in keiner anderen Partition verwendet.

Konfigurationstyp

Datenbank

Parametertyp

Online konfigurierbar

Standardwert [Bereich]

0 [0 (keine Erfassung), 5 - 32 767]

Die vom Ereignismonitor für Statistiken erfasste Workload-Management-Statistik kann zum Überwachen von Kurz- und Langzeitsystemverhalten verwendet werden. Ein kleines Intervall kann zum Erreichen von Kurz- und Langzeitsystemverhalten verwendet werden, da die Ergebnisse zusammengefasst werden können und so ein Lanzeitverhalten erreicht werden kann. Ein manuelles Zusammenfassen der Ergebnisse aus unterschiedlichen Intervallen verkompliziert allerdings die Analyse. Ein kleines Intervall erhöht unnötigerweise den Systemaufwand, wenn es nicht erforderlich ist. Aus diesem Grund sollten Sie das Intervall reduzieren, um das Kurzzeitverhalten zu erfassen, und das Intervall erhöhen, um den Systemaufwand zu reduzieren, wenn lediglich die Analyse des Langzeitverhaltens genügt.

Das Intervall muss für jede Datenbank angepasst werden, und nicht für die einzelnen SQL-Anforderungen, Befehlsaufrufe oder Anwendungen. Andere Konfigurationsparameter müssen nicht berücksichtigt werden.

Anmerkung: Alle WLM-Statistiktabellenfunktionen geben Statistikdaten zurück, die seit der letzten Zurücksetzung der Statistik gesammelt wurden. Die Statistik wird regelmäßig auf der Basis des durch diesen Konfigurationsparameter angegebenen Intervalls zurückgesetzt.

Kapitel 12. Katalogsichten

SYSCAT.HISTOGRAMTEMPLATEBINS

Jede Zeile stellt einen Bin in der Histogrammschablone dar.

Tabelle 129. Katalogsicht SYSCAT.HISTOGRAMTEMPLATEBINS

Spaltenname	Datentyp	Nullwert?	Beschreibung
TEMPLATENAME	VARCHAR(128)	Y	Name der Histogrammschablone.
TEMPLATEID	INTEGER		Kennung (ID) für die Histogrammschablone.
BINID	INTEGER		Kennung (ID) für den Histogrammschablonenbin.
BINUPPERVALUE	BIGINT		Die Obergrenze für einen einzelnen Bin in der Histogrammschablone.

SYSCAT.HISTOGRAMTEMPLATES

Jede Zeile stellt eine Histogrammschablone dar.

Tabelle 130. Katalogsicht SYSCAT.HISTOGRAMTEMPLATES

Spaltenname	Datentyp	Nullwert?	Beschreibung
TEMPLATEID	INTEGER		Kennung (ID) für die Histogrammschablone.
TEMPLATENAME	VARCHAR(128)		Name der Histogrammschablone.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung der Histogrammschablone.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung der Histogrammschablone.
NUMBINS	INTEGER		Anzahl der Bins in der Histogrammschablone, einschließlich des letzten Bins, der über eine unbegrenzte Obergrenze verfügt.
REMARKS	VARCHAR (254)	Y	Vom Benutzer bereitgestellte Kommentare oder NULL.

SYSCAT.HISTOGRAMTEMPLATEUSE

Jede Zeile stellt eine Beziehung zwischen einem Workload-Management-Objekt, das Histogrammschablonen verwenden kann, und einer Histogrammschablone dar.

Tabelle 131. Katalogsicht SYSCAT.HISTOGRAMTEMPLATEUSE

Spaltenname	Datentyp	Nullwert?	Beschreibung
TEMPLATENAME	VARCHAR(128)	Y	Name der Histogrammschablone.
TEMPLATEID	INTEGER		Kennung (ID) für die Histogrammschablone.

Tabelle 131. Katalogsicht SYSCAT.HISTOGRAMTEMPLATEUSE (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
HISTOGRAMTYPE	CHAR (1)		Der Typ der Informationen, die von Histogrammen erfasst werden, die auf dieser Schablone basieren. <ul style="list-style-type: none"> • C = Histogramm des geschätzten Aufwands für Aktivitäten • E = Histogramm der Aktivitätsausführungszeit • I = Histogramm der Zwischenankunftszeit von Aktivitäten • L = Histogramm der Aktivitätslaufzeit • Q = Histogramm der Aktivitätswartezeit • R = Histogramm der Anforderungsausführungszeit
OBJECTTYPE	CHAR (1)		Der Typ des WLM-Objekts. <ul style="list-style-type: none"> • b = Serviceklasse • k = Arbeitsaktion
OBJECTID	INTEGER		Kennung (ID) für das WLM-Objekt.
SERVICECLASSNAME	VARCHAR(128)	Y	Name der Serviceklasse.
PARENTSERVICECLASSNAME	VARCHAR(128)	Y	Name der übergeordneten Serviceklasse.
WORKACTIONNAME	VARCHAR(128)	Y	Name der Arbeitsaktion.
WORKACTIONSETNAME	VARCHAR(128)	Y	Name des Arbeitsaktionssets.

SYSCAT.SERVICECLASSES

Jede Zeile stellt eine Serviceklasse dar.

Tabelle 132. Katalogsicht SYSCAT.SERVICECLASSES

Spaltenname	Datentyp	Nullwert?	Beschreibung
SERVICECLASSNAME	VARCHAR(128)		Name der Serviceklasse.
PARENTSERVICECLASSNAME	VARCHAR(128)	Y	Name der Serviceklasse der übergeordneten Servicesuperklasse.
SERVICECLASSID	SMALLINT		Kennung (ID) für die Serviceklasse.
PARENTID	SMALLINT		Kennung (ID) für die übergeordnete Serviceklasse für diese Serviceklasse. 0, wenn diese Serviceklasse eine Superserviceklasse ist.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung der Serviceklasse.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung der Serviceklasse.
ENABLED	CHAR (1)		Status der Serviceklasse. <ul style="list-style-type: none"> • N = Inaktiviert • Y = Aktiviert

Tabelle 132. Katalogsicht SYSCAT.SERVICECLASSES (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
AGENTPRIORITY	SMALLINT		Threadpriorität der Agenten in der Serviceklasse in Relation zur normalen Priorität der DB2-Threads. <ul style="list-style-type: none"> • -20 bis 20 (Linux und UNIX) • -6 bis 6 (Windows) • -32768 = nicht festgelegt
PREFETCHPRIORITY	CHAR (1)		Priorität des Vorableszugriffs der Agenten in der Serviceklasse. <ul style="list-style-type: none"> • H = Hoch • L = Niedrig • M = Mittel • Leer = nicht festgelegt
INBOUNDCORRELATOR	VARCHAR(128)	Y	Zur zukünftigen Verwendung.
OUTBOUNDCORRELATOR	VARCHAR(128)	Y	Zeichenfolge, die zum Verknüpfen der Serviceklasse mit einer Workload-Manager-Serviceklasse eines Betriebssystems verwendet wird.
COLLECTAGGACTDATA	CHAR (1)		Gibt an, welche zusammengefassten Aktivitätsdaten vom entsprechenden Ereignismonitor erfasst werden sollten. <ul style="list-style-type: none"> • B = Zusammengefasste Basisaktivitätsdaten erfassen • E = Erweiterte zusammengefasste Aktivitätsdaten erfassen • N = Keine
COLLECTAGGREQDATA	CHAR (1)		Gibt an, welche zusammengefassten Aktivitätsdaten vom entsprechenden Ereignismonitor erfasst werden sollten. <ul style="list-style-type: none"> • B = Zusammengefasste Basisanforderungsdaten erfassen • N = Keine
COLLECTACTDATA	CHAR (1)		Gibt an, welche Aktivitätsdaten vom entsprechenden Ereignismonitor erfasst werden sollten. <ul style="list-style-type: none"> • D = Aktivitätsdaten mit Details • N = Keine • V = Aktivitätsdaten mit Details und Werten • W = Aktivitätsdaten ohne Details
COLLECTACTPARTITION	CHAR (1)		Gibt an, wo die Aktivitätsdaten erfasst werden. <ul style="list-style-type: none"> • C = Datenbankpartition des Koordinators der Aktivität • D = Alle Datenbankpartitionen
REMARKS	VARCHAR (254)	Y	Vom Benutzer bereitgestellte Kommentare oder NULL.

SYSCAT.THRESHOLDS

Jede Zeile stellt einen Schwellenwert dar.

Tabelle 133. Katalogsicht SYSCAT.THRESHOLDS

Spaltenname	Datentyp	Nullwert?	Beschreibung
THRESHOLDNAME	VARCHAR(128)		Name des Schwellenwerts.
THRESHOLDID	INTEGER		Kennung (ID) für den Schwellenwert.
ORIGIN	CHAR (1)		Ursprung des Schwellenwerts. <ul style="list-style-type: none">• U = Schwellenwert wurde von einem Benutzer erstellt• W = Schwellenwert wurde mit Hilfe eines Arbeitsaktionssets erstellt
THRESHOLDCLASS	CHAR (1)		Klassifizierung des Schwellenwerts. <ul style="list-style-type: none">• A = Zusammengefasster Schwellenwert• C = Aktivitätsschwellenwert
THRESHOLDPREDICATE	VARCHAR(128)		Typ des Schwellenwerts. Die folgenden Werte sind möglich: <ul style="list-style-type: none">• CONCDDBC• CONCWCN• CONCWOC• CONNIDLETIME• DBCONN• ESTSQLCOST• ROWSRET• SCCONN• TEMPSPACE• TOTALTIMECONCDDBC
THRESHOLDPREDICATEID	SMALLINT		Kennung (ID) für das Schwellenwertvergleichselement.
DOMAIN	CHAR (2)		Domäne des Schwellenwerts. <ul style="list-style-type: none">• DB = Datenbank• SB = Serviceunterklasse• SP = Servicesuperklasse• WA = Arbeitsaktionsset• WD = Workloaddefinition
DOMAINID	INTEGER		Kennung (ID) für das Objekt, mit dem der Schwellenwert verknüpft ist. Dabei kann es sich um eine Serviceklasse, eine Arbeitsaktion oder eine eindeutige Workload-ID handeln. Wenn dies ein Datenbankschwellenwert ist, ist dieser Wert 0.
ENFORCEMENT	CHAR (1)		Der Umsetzungsbereich für den Schwellenwert. <ul style="list-style-type: none">• D = Datenbank• P = Datenbankpartition• W = Workloadvorkommen

Tabelle 133. Katalogsicht SYSCAT.THRESHOLDS (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
QUEUEING	CHAR (1)		<ul style="list-style-type: none"> • N = Der Schwellenwert wird nicht in die Warteschlange gestellt. • Y = Der Schwellenwert wird in die Warteschlange gestellt.
MAXVALUE	BIGINT		Vom Schwellenwert angegebene Obergrenze.
QUEUESIZE	INTEGER		Die Größe der Warteschlange, falls QUEUEING auf 'Y' gesetzt ist. Andernfalls -1.
COLLECTACTDATA	CHAR (1)		<p>Gibt an, welche Aktivitätsdaten vom entsprechenden Ereignismonitor erfasst werden sollen.</p> <ul style="list-style-type: none"> • A = Aktivitätsdaten ohne Details • D = Aktivitätsdaten mit Details • N = Keine • V = Aktivitätsdaten mit Details und Werten
COLLECTACTPARTITION	CHAR (1)		<p>Gibt an, wo die Aktivitätsdaten erfasst werden.</p> <ul style="list-style-type: none"> • C = Datenbankpartition des Koordinators der Aktivität • D = Alle Datenbankpartitionen
EXECUTION	CHAR (1)		<p>Gibt an, ob die Ausführung fortgesetzt wird, nachdem der Schwellenwert überschritten wurde.</p> <ul style="list-style-type: none"> • C = Ausführung wird fortgesetzt • S = Ausführung wird gestoppt
ENABLED	CHAR (1)		<ul style="list-style-type: none"> • N = Dieser Schwellenwert ist inaktiviert. • Y = Dieser Schwellenwert ist aktiviert.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung des Schwellenwerts.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung des Schwellenwerts.
REMARKS	VARCHAR (254)	Y	Vom Benutzer bereitgestellte Kommentare oder NULL.

SYSCAT.WORKACTIONS

Jede Zeile stellt eine Arbeitsaktion dar, die für ein Arbeitsaktionsset definiert ist.

Tabelle 134. Katalogsicht SYSCAT.WORKACTIONS

Spaltenname	Datentyp	Nullwert?	Beschreibung
ACTIONNAME	VARCHAR(128)		Name der Arbeitsaktion.
ACTIONID	INTEGER		Kennung (ID) für die Arbeitsaktion.
ACTIONSETNAME	VARCHAR(128)	Y	Name des Arbeitsaktionssets.

Tabelle 134. Katalogsicht SYSCAT.WORKACTIONS (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
ACTIONSETID	INTEGER		Kennung (ID) des Arbeitsaktionssets, zu dem diese Arbeitsaktion gehört. Diese Spalte verweist auf die Spalte ACTIONSETID in der Sicht SYSCAT.WORKACTIONSETS.
WORKCLASSNAME	VARCHAR(128)	Y	Name der Arbeitsklasse.
WORKCLASSID	INTEGER		Kennung (ID) für die Arbeitsklasse. Diese Spalte verweist auf die Spalte WORKCLASSID in der Sicht SYSCAT.WORKCLASSES.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung der Arbeitsaktion.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung der Arbeitsaktion.
ENABLED	CHAR (1)		<ul style="list-style-type: none"> • N = Diese Arbeitsaktion ist inaktiviert. • Y = Diese Arbeitsaktion ist aktiviert.

Tabelle 134. Katalogsicht SYSCAT.WORKACTIONS (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
ACTIONTYPE	CHAR (1)		<p>Der Aktionstyp, der für jede DB2-Aktivität ausgeführt wird, die den Arbeitsklassenattributen entspricht, die im entsprechenden Bereich der Arbeitsklasse angegeben sind. Bei dieser Spaltenbeschreibung verweist OBJECTTYPE auf die Spalte OBJECTTYPE in der Sicht SYSCAT.WORKACTIONSETS.</p> <ul style="list-style-type: none"> • B - Zusammengefasste Basisaktivitätsdaten erfassen. Dieser Aktionstyp kann nur angegeben werden, wenn OBJECTTYPE auf 'b' gesetzt ist (Serviceklasse). • C - Die Ausführung einer beliebigen DB2-Aktivität zulassen, die zu der Arbeitsklasse gehört, mit der diese Arbeitsaktion verknüpft ist, um den Zähler für diese Arbeitsklasse auszuführen und zu erhöhen. • D - Aktivitätsdaten mit Details in der Datenbankpartition des Koordinators der Aktivität erfassen. • E - Erweiterte zusammengefasste Aktivitätsdaten erfassen. Dieser Aktionstyp kann nur angegeben werden, wenn OBJECTTYPE auf 'b' gesetzt ist (Serviceklasse). • M - Einer Serviceunterklasse zuordnen. Dieser Aktionstyp kann nur angegeben werden, wenn OBJECTTYPE auf 'b' gesetzt ist (Serviceklasse). • P - Die Ausführung einer beliebigen DB2-Aktivität unterbinden, die zu der Arbeitsklasse gehört, mit der diese Arbeitsaktion verknüpft ist. • T - Die Aktion hat die Form eines Schwellenwerts. Dieser Aktionstyp kann nur angegeben werden, wenn OBJECTTYPE auf 'f' gesetzt ist (Schwellenwert). • U - Alle Aktivitäten, die über eine Verschachtelungsebene 0 verfügen, und alle Aktivitäten, die unter dieser Aktivität verschachtelt sind, einer Serviceunterklasse zuordnen. Dieser Aktionstyp kann nur angegeben werden, wenn OBJECTTYPE auf 'b' gesetzt ist (Serviceklasse).

Tabelle 134. Katalogsicht SYSCAT.WORKACTIONS (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
ACTIONTYPE (Forts.)			<ul style="list-style-type: none"> • V - Aktivitätsdaten mit Details und Werten in der Datenbankpartition des Koordinators der Aktivität erfassen. • W - Aktivitätsdaten ohne Details in der Datenbankpartition des Koordinators der Aktivität erfassen. • X - Aktivitätsdaten mit Details in der Datenbankpartition des Koordinators der Aktivität und Aktivitätsdaten in allen Datenbankpartitionen erfassen. • Y - Aktivitätsdaten mit Details und Werten in der Datenbankpartition des Koordinators der Aktivität und Aktivitätsdaten in allen Datenbankpartitionen erfassen. • Z - Aktivitätsdaten ohne Details in allen Datenbankpartitionen erfassen.
REFOBJECTID	INTEGER	Y	Wenn ACTIONTYPE den Wert 'M' (Zuordnung) oder 'N' (Zuordnung verschachtelt) hat, wird dieser Wert auf die ID der Serviceunterklasse gesetzt, der die DB2-Aktivität zugeordnet ist. Wenn ACTIONTYPE den Wert 'T' (Schwellenwert) hat, wird dieser Wert auf die ID des zu verwendenden Schwellenwerts gesetzt. Bei allen anderen Aktionen ist der Wert NULL.
REFOBJECTTYPE	VARCHAR(30)		Wenn ACTIONTYPE den Wert 'M' oder 'N' hat, wird dieser Wert auf 'SERVICE CLASS' gesetzt; wenn ACTIONTYPE den Wert 'T' hat, wird dieser Wert auf 'THRESHOLD' gesetzt; andernfalls ist der Wert NULL.

SYSCAT.WORKACTIONSETS

Jede Zeile stellt ein Arbeitsaktionsset dar.

Tabelle 135. Katalogsicht SYSCAT.WORKACTIONSETS

Spaltenname	Datentyp	Nullwert?	Beschreibung
ACTIONSETNAME	VARCHAR(128)		Name des Arbeitsaktionssets.
ACTIONSETID	INTEGER		Kennung (ID) für das Arbeitsaktionsset.
WORKCLASSETNAME	VARCHAR(128)	Y	Name des Arbeitsklassensets.
WORKCLASSETID	INTEGER		Die Kennung (ID) des Arbeitsklassensets, das dem durch OBJECTID angegebenen Objekt zugeordnet werden soll. Diese Spalte verweist auf WORKCLASSETID in der Sicht SYSCAT.WORKCLASSETS.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung des Arbeitsaktionssets.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung des Arbeitsaktionssets.

Tabelle 135. Katalogsicht SYSCAT.WORKACTIONSETS (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
ENABLED	CHAR (1)		<ul style="list-style-type: none"> • N = Dieses Arbeitsaktionsset ist inaktiviert. • Y = Dieses Arbeitsaktionsset ist aktiviert.
OBJECTTYPE	CHAR (1)		<ul style="list-style-type: none"> • b = Servicesuperklasse • Leer = Datenbank
OBJECTNAME	VARCHAR(128)	Y	Name der Serviceklasse.
OBJECTID	INTEGER		Die Kennung (ID) des Objekts, dem das (durch WORKCLASSETID angegebene) Arbeitsklassenset zugeordnet wird. Wenn OBJECTTYPE leer ist, hat OBJECTID den Wert -1. Wenn OBJECTTYPE den Wert 'b' hat, ist OBJECTID die ID der Servicesuperklasse.
REMARKS	VARCHAR (254)	Y	Vom Benutzer bereitgestellte Kommentare oder NULL.

SYSCAT.WORKCLASSES

Jede Zeile stellt eine für ein Arbeitsklassenset definierte Arbeitsklasse dar.

Tabelle 136. Katalogsicht SYSCAT.WORKCLASSES

Spaltenname	Datentyp	Nullwert?	Beschreibung
WORKCLASSNAME	VARCHAR(128)		Name der Arbeitsklasse.
WORKCLASSETNAME	VARCHAR(128)	Y	Name des Arbeitsklassensets.
WORKCLASSID	INTEGER		Kennung (ID) für die Arbeitsklasse.
WORKCLASSETID	INTEGER		Kennung (ID) des Arbeitsaktionssets, zu dem diese Arbeitsklasse gehört. Diese Spalte verweist auf die Spalte WORKCLASSETID in der Sicht SYSCAT.WORKCLASSETS.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung der Arbeitsklasse.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung der Arbeitsklasse.
WORKTYPE	SMALLINT		Der Typ der DB2-Aktivität. <ul style="list-style-type: none"> • 1 = ALL • 2 = READ • 3 = WRITE • 4 = CALL • 5 = DML • 6 = DDL • 7 = LOAD
RANGEUNITS	CHAR (1)		Die zu verwendenden Einheiten für die Unter- und Obergrenze des Bereichs. <ul style="list-style-type: none"> • C = Kardinalität • T = Timerons • Leer = Nicht anwendbar

Tabelle 136. Katalogsicht SYSCAT.WORKCLASSES (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
FROMVALUE	DOUBLE	Y	Die Untergrenze des Bereichs in den durch RANGEUNITS angegebenen Einheiten. Nullwert, wenn RANGEUNITS leer ist.
TOVALUE	DOUBLE	Y	Die Obergrenze des Bereichs in den durch RANGEUNITS angegebenen Einheiten. Nullwert, wenn RANGEUNITS leer ist. Der Wert -1 wird verwendet, um keine Obergrenze anzugeben.
ROUTINESHEMA	VARCHAR(128)	Y	Schemaname der Prozeduren, die über die Anweisung CALL aufgerufen werden. Nullwert, wenn WORKTYPE nicht auf 4 (CALL) oder 1 (ALL) gesetzt wurde.
EVALUATIONORDER	SMALLINT		Gibt die Bewertungsreihenfolge eindeutig an, die für das Auswählen einer Arbeitsklasse innerhalb eines Arbeitsklassensets verwendet wird.

SYSCAT.WORKCLASSETS

Jede Zeile stellt ein Arbeitsklassenset dar.

Tabelle 137. Katalogsicht SYSCAT.WORKCLASSETS

Spaltenname	Datentyp	Nullwert?	Beschreibung
WORKCLASSETNAME	VARCHAR(128)		Name des Arbeitsklassensets.
WORKCLASSETID	INTEGER		Kennung (ID) für das Arbeitsklassenset.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung des Arbeitsklassensets.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung des Arbeitsklassensets.
REMARKS	VARCHAR (254)	Y	Vom Benutzer bereitgestellte Kommentare oder NULL.

SYSCAT.WORKLOADAUTH

Jede Zeile stellt einen Benutzer, eine Gruppe oder eine Rolle dar, dem bzw. der das Zugriffsrecht USAGE für eine Workload erteilt wurde.

Tabelle 138. Katalogsicht SYSCAT.WORKLOADAUTH

Spaltenname	Datentyp	Nullwert?	Beschreibung
WORKLOADID	INTEGER		Kennung (ID) für die Workload.
WORKLOADNAME	VARCHAR(128)		Name der Workload.
GRANTOR	VARCHAR(128)		Berechtigungsgeber des Zugriffsrechts.
GRANTORTYPE	CHAR (1)		• U = Der Berechtigte ist ein Einzelbenutzer.
GRANTEE	VARCHAR(128)		Inhaber des Zugriffsrechts.

Tabelle 138. Katalogsicht SYSCAT.WORKLOADAUTH (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
GRANTEETYPE	CHAR (1)		<ul style="list-style-type: none"> • G = Der Berechtigte ist eine Gruppe • R = Der Berechtigte ist eine Rolle • U = Der Berechtigte ist ein Einzelbenutzer.
USAGEAUTH	CHAR (1)		Gibt an, ob der Berechtigte das Zugriffsrecht USAGE für die Workload hat. <ul style="list-style-type: none"> • N = Nicht zugewiesen • Y = Zugewiesen

SYSCAT.WORKLOADCONNATTR

Jede Zeile stellt ein Verbindungsattribut in der Definition einer Workload dar.

Tabelle 139. Katalogsicht SYSCAT.WORKLOADCONNATTR

Spaltenname	Datentyp	Nullwert?	Beschreibung
WORKLOADID	INTEGER		Kennung (ID) für die Workload.
WORKLOADNAME	VARCHAR(128)		Name der Workload.
CONNATTRTYPE	VARCHAR(30)		Typ des Verbindungsattributs. <ul style="list-style-type: none"> • 1 = APPLNAME • 2 = SYSTEM_USER • 3 = SESSION_USER • 4 = SESSION_USER GROUP • 5 = SESSION_USER ROLE • 6 = CURRENT CLIENT_USERID • 7 = CURRENT CLIENT_APPLNAME • 8 = CURRENT CLIENT_WRKSTNNAME • 9 = CURRENT CLIENT_ACCTNG
CONNATTRVALUE	VARCHAR(1000)		Wert des Verbindungsattributs.

SYSCAT.WORKLOADS

Jede Zeile stellt eine Workload dar.

Tabelle 140. Katalogsicht SYSCAT.WORKLOADS

Spaltenname	Datentyp	Nullwert?	Beschreibung
WORKLOADID	INTEGER		Kennung (ID) für die Workload.
WORKLOADNAME	VARCHAR(128)		Name der Workload.
EVALUATIONORDER	SMALLINT		Für das Auswählen einer Workload verwendete Bewertungsreihenfolge.
CREATE_TIME	TIMESTAMP		Zeitpunkt der Erstellung der Workload.
ALTER_TIME	TIMESTAMP		Zeitpunkt der letzten Änderung der Workload.
ENABLED	CHAR (1)		<ul style="list-style-type: none"> • N = Diese Workload ist inaktiviert. • Y = Diese Workload ist aktiviert.

Tabelle 140. Katalogsicht SYSCAT.WORKLOADS (Forts.)

Spaltenname	Datentyp	Nullwert?	Beschreibung
ALLOWACCESS	CHAR (1)		<ul style="list-style-type: none"> • N = Eine mit dieser Workload verknüpfte UOW wird zurückgewiesen. • Y = Eine mit dieser Workload verknüpfte UOW (Unit of Work) kann auf die Datenbank zugreifen.
SERVICECLASSNAME	VARCHAR(128)		Name der Serviceunterklasse, der eine (mit dieser Workload verknüpfte) UOW zugeordnet ist.
PARENTSERVICECLASSNAME	VARCHAR(128)	Y	Name der Servicesuperklasse, der eine (mit dieser Workload verknüpfte) UOW zugeordnet ist.
COLLECTAGGACTDATA	CHAR (1)		<p>Gibt an, welche zusammengefassten Aktivitätsdaten für die Workload vom entsprechenden Ereignismonitor erfasst werden sollten.</p> <ul style="list-style-type: none"> • N = Keine
COLLECTACTDATA	CHAR (1)		<p>Gibt an, welche Aktivitätsdaten vom entsprechenden Ereignismonitor erfasst werden sollten.</p> <ul style="list-style-type: none"> • D = Aktivitätsdaten mit Details • N = Keine • V = Aktivitätsdaten mit Details und Werten; gilt, wenn die Spalte COLLECT auf 'C' gesetzt wurde • W = Aktivitätsdaten ohne Details
COLLECTACTPARTITION	CHAR (1)		<p>Gibt an, wo die Aktivitätsdaten erfasst werden.</p> <ul style="list-style-type: none"> • C = Datenbankpartition des Koordinators der Aktivität • D = Alle Datenbankpartitionen
EXTERNALNAME	VARCHAR(128)	Y	Zur zukünftigen Verwendung reserviert.
REMARKS	VARCHAR (254)	Y	Vom Benutzer bereitgestellte Kommentare oder NULL.

Teil 6. Anhänge und Schlussteil

Anhang A. Aspekte von DDL-Anweisungen für Workload-Management

DDL-Anweisungen beim DB2-Workload-Management bestehen aus den Anweisungen CREATE, ALTER und DROP, die Sie zur Arbeit mit Serviceklassen, Workloads, Arbeitsklassensets, Arbeitsaktionssets, Schwellenwerten und Histogrammen verwenden.

Es gibt die folgenden DDL-Anweisungen für das Workload-Management:

- CREATE WORKLOAD, ALTER WORKLOAD und DROP WORKLOAD
- GRANT USAGE ON WORKLOAD und REVOKE USAGE ON WORKLOAD
- CREATE SERVICE CLASS, ALTER SERVICE CLASS und DROP SERVICE CLASS
- CREATE WORK CLASS SET, ALTER WORK CLASS SET und DROP WORK CLASS SET
- CREATE WORK ACTION SET, ALTER WORK ACTION SET und DROP WORK ACTION SET
- CREATE THRESHOLD, ALTER THRESHOLD und DROP THRESHOLD
- CREATE HISTOGRAM TEMPLATE, ALTER HISTOGRAM TEMPLATE und DROP HISTOGRAM TEMPLATE

DDL-Anweisungen beim Workload-Management unterscheiden sich von anderen DB2-DDL-Anweisungen:

- Über alle Datenbankpartitionen ist zu jeweils nur eine nicht festgeschriebene Workload-Management-DDL-Anweisung zulässig. Wenn eine nicht festgeschriebene Workload-Management-DDL-Anweisung vorhanden ist, warten nachfolgende Workload-Management-DDL-Anweisungen, bis für die nicht festgeschriebene Workload-Management-DDL-Anweisung entweder COMMIT oder ROLLBACK ausgeführt wurde. Workload-Management-DDL-Anweisungen werden in der Reihenfolge verarbeitet, in der sie abgesetzt werden.
- Auf jede Workload-Management-DDL-Anweisung muss eine COMMIT- oder ROLLBACK-Anweisung folgen.
- Eine Workload-Management-DDL-Anweisung kann nicht in einer XA-Transaktion abgesetzt werden. Nachdem eine Verbindung eine Workload-Management-DDL-Anweisung absetzt, muss dieselbe Verbindung direkt nach der Workload-Management-DDL-Anweisung eine COMMIT- oder ROLLBACK-Anweisung absetzen. Bei XA-Transaktionen ist es möglich, dass mehrere Verbindungen mit einer Transaktion verknüpft sind und dass jede der Verbindungen die Transaktion festschreiben oder rückgängig machen kann. In dieser Situation kann nicht sichergestellt werden, dass die Workload-Management-Umgebung korrekt implementiert wird.
- DB2 für z/OS erkennt keine Workload-Management-DDL-Anweisungen von DB2 Database für Linux, UNIX und Windows.

Viele Datenbankobjekte haben Eigner, und diese Eigner haben die Berechtigung, ihre eigenen Objekte zu ändern. Im Gegensatz zu den meisten Objekten haben Workload-Management-Objekte keine Eigner, da dies zu unvorhersehbaren Problemen führen könnte. Wenn z. B. eine Ressourcenzuordnungseinstellung für eine Serviceklasse geändert wird, wirkt sich die Änderung nicht nur auf die Serviceklasse selbst, sondern auch auf andere Serviceklassen in derselben Stufe aus.

Nehmen wir z. B. an, dass eine Servicesuperklasse zwei benutzerdefinierte Serviceunterklassen, A und B, hat und dass jede Serviceunterklasse einen anderen Eigner hat. Die Vorablesezugriffspriorität hat anfangs den Wert MEDIUM für die Standardserviceunterklasse und die beiden Serviceunterklassen A und B. Wenn der Eigner der Serviceunterklasse A deren Vorablesezugriffspriorität in HIGH ändern würde und viele Vorablesezugriffsanforderungen von dieser Serviceunterklasse ausgehen, erhielten Verbindungen zur Serviceunterklasse B und zur Standardunterklasse *nicht genügend* Vorabzugriffsdienste, und die Leistung der in diesen Serviceunterklassen ausgeführten Aktivitäten könnte leiden. Aus solchen Gründen haben DB2-Workload-Management-Objekte keine Eigner.

Anhang B. Integration von DB2-Workload-Management und AIX Workload Manager

Wenn DB2-Workload-Management mit AIX Workload Manager kombiniert wird, stehen weitere Funktionen zur Verfügung. Mit Hilfe von AIX Workload Manager kann der CPU-Anteil gesteuert werden, der den einzelnen Serviceklassen zugeordnet wird.

Es gibt u. a. die Möglichkeit, für jede Serviceklasse den minimalen, maximalen oder relativen CPU-Anteil festzulegen. Die Zuordnung zwischen DB2-Serviceklassen und Serviceklassen von AIX Workload Manager wird in der Definition der DB2-Serviceklasse mit der Option `OUTBOUND CORRELATOR` der Anweisung `CREATE SERVICE CLASS` oder `ALTER SERVICE CLASS` angegeben. DB2-Serviceklassen sind der einzige Punkt zur Integration zwischen DB2-Workload-Management und AIX Workload Manager.

In diesem Thema wird Folgendes beschrieben:

- Empfohlene Zuordnungen zwischen Serviceklassen von DB2 und AIX Workload Manager
- Anweisungen für die Definition von Zuordnungen zwischen DB2-Serviceklassen und Serviceklassen von AIX Workload Manager
- Einstellen von CPU-Steuerungen auf den Serviceklassen von AIX Workload Manager

Empfohlene Zuordnungen zwischen Serviceklassen von DB2 und AIX Workload Manager

Sie können DB2-Serviceklassen den Serviceklassen von AIX Workload Manager in jeder gewünschten Weise zuordnen. Sie sollten jedoch zunächst eine 1:1-Zuordnung von DB2-Serviceklassen zu Serviceklassen von AIX Workload Manager anstreben. Eine 1:1-Zuordnung ist ein guter Ausgangspunkt, da Aktivitäten in einer DB2-Serviceklasse mehr oder weniger das gleiche Leistungsziel haben sollten. Auf direktestem Weg erreichen Sie dieses Ziel, indem Sie die AIX-Ressourcen anpassen, die dieser Serviceklasse von AIX Workload Manager zugeordnet sind. Durch eine 1:1-Zuordnung zwischen einer DB2-Serviceklasse und einer Serviceklasse von AIX Workload Manager können Sie die AIX-Ressourcen für jede Serviceklasse individuell anpassen.

Die folgende Abbildung zeigt die Integration von DB2-Workload-Manager und AIX Workload Manager. Beachten Sie die 1:1-Zuordnung zwischen jeder DB2-Serviceklasse und jeder Serviceklasse von AIX Workload Manager auf der Ebene der Servicesuperklasse und der Serviceunterklasse.

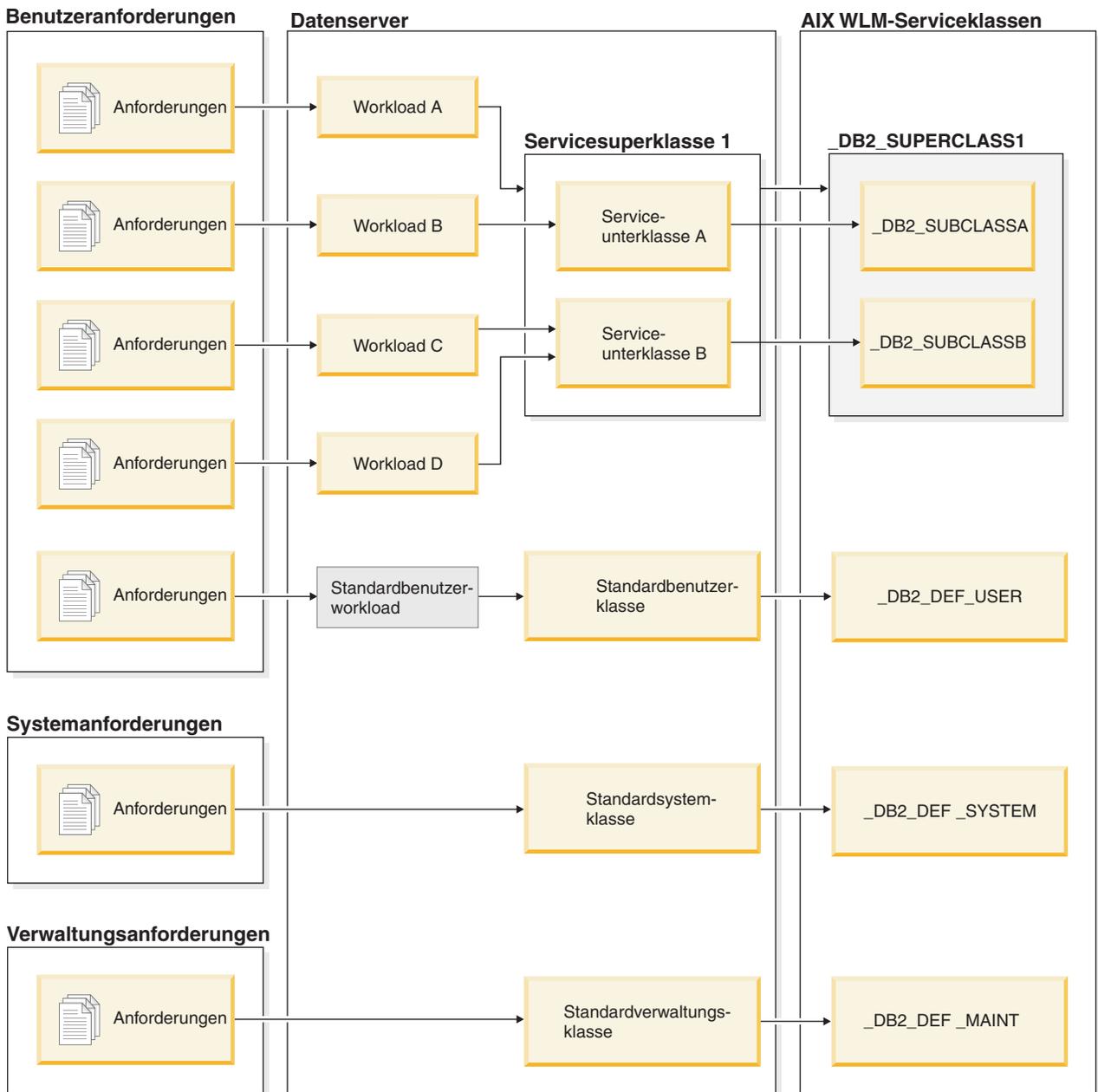


Abbildung 28. Integration von DB2-Workload-Manager und AIX Workload Manager

Wenn eine DB2-Umgebung aus einer einzigen Datenbank in einer einzigen DB2-Instanz besteht, wie im Beispiel in der obigen Abbildung, ist eine direkte Zuordnung zwischen DB2-Serviceklassen und Serviceklassen von AIX Workload Manager möglich. Jede DB2-Servicesuperklasse kann eine entsprechende Servicesuperklasse in AIX Workload Manager haben, und jede DB2-Serviceunterklasse kann einer entsprechenden AIX-Serviceunterklasse zugeordnet werden.

Die folgende Abbildung zeigt, dass in Situationen, in denen die DB2-Umgebung aus mehreren Datenbanken und DB2-Instanzen besteht, vier Ebenen Kandidaten für die Ressourcensteuerung sind.

DB2-Instanz

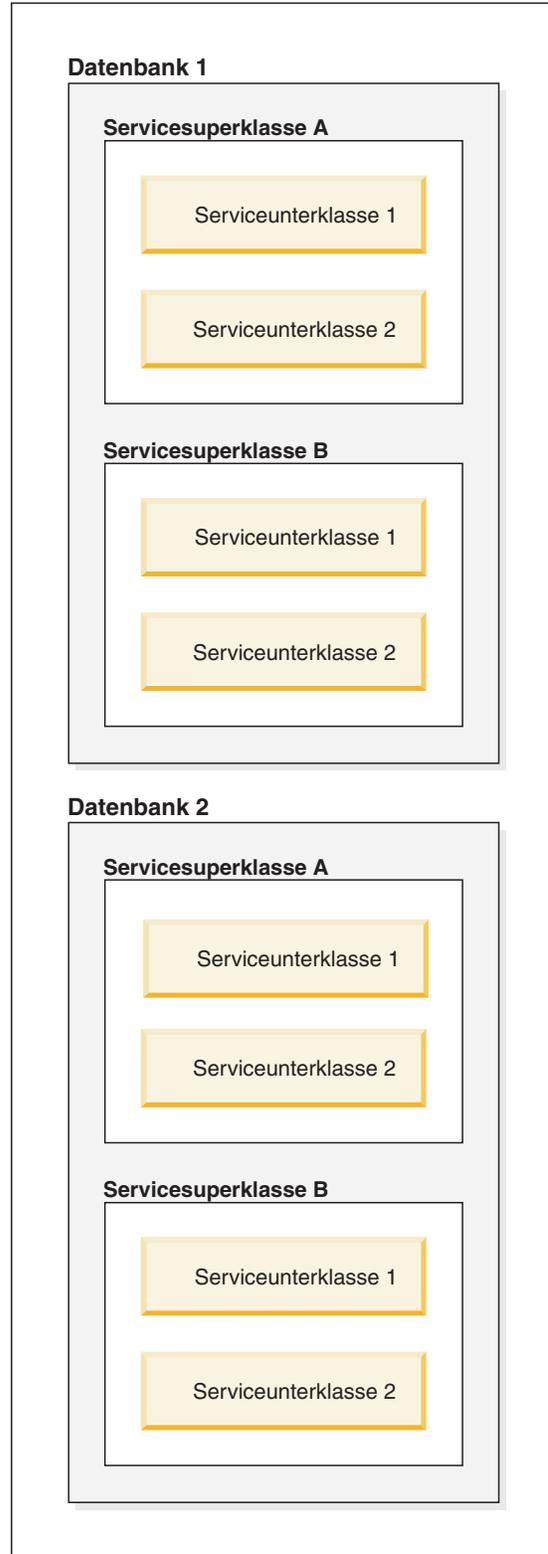


Abbildung 29. Ressourcensteuerungsebenen in einer DB2-Umgebung

Da AIX Workload Manager eine zweistufige Hierarchie mit Superklasse und Unterklasse unterstützt, können jeweils nur zwei Ebenen einer DB2-Umgebung den Serviceklassen von AIX Workload Manager zugeordnet werden. Im Folgenden sind einige Beispielkonfigurationen aufgeführt.

Die folgende Abbildung zeigt eine Möglichkeit, die 1:1-Zuordnung bei mehreren Datenbanken jeweils mit Superklassen zu erreichen. Hier hat jede Datenbank ihre eigene Superklasse von AIX Workload Manager, und jede DB2-Servicesuperklasse wird einer Unterklasse von AIX Workload Manager zugeordnet.

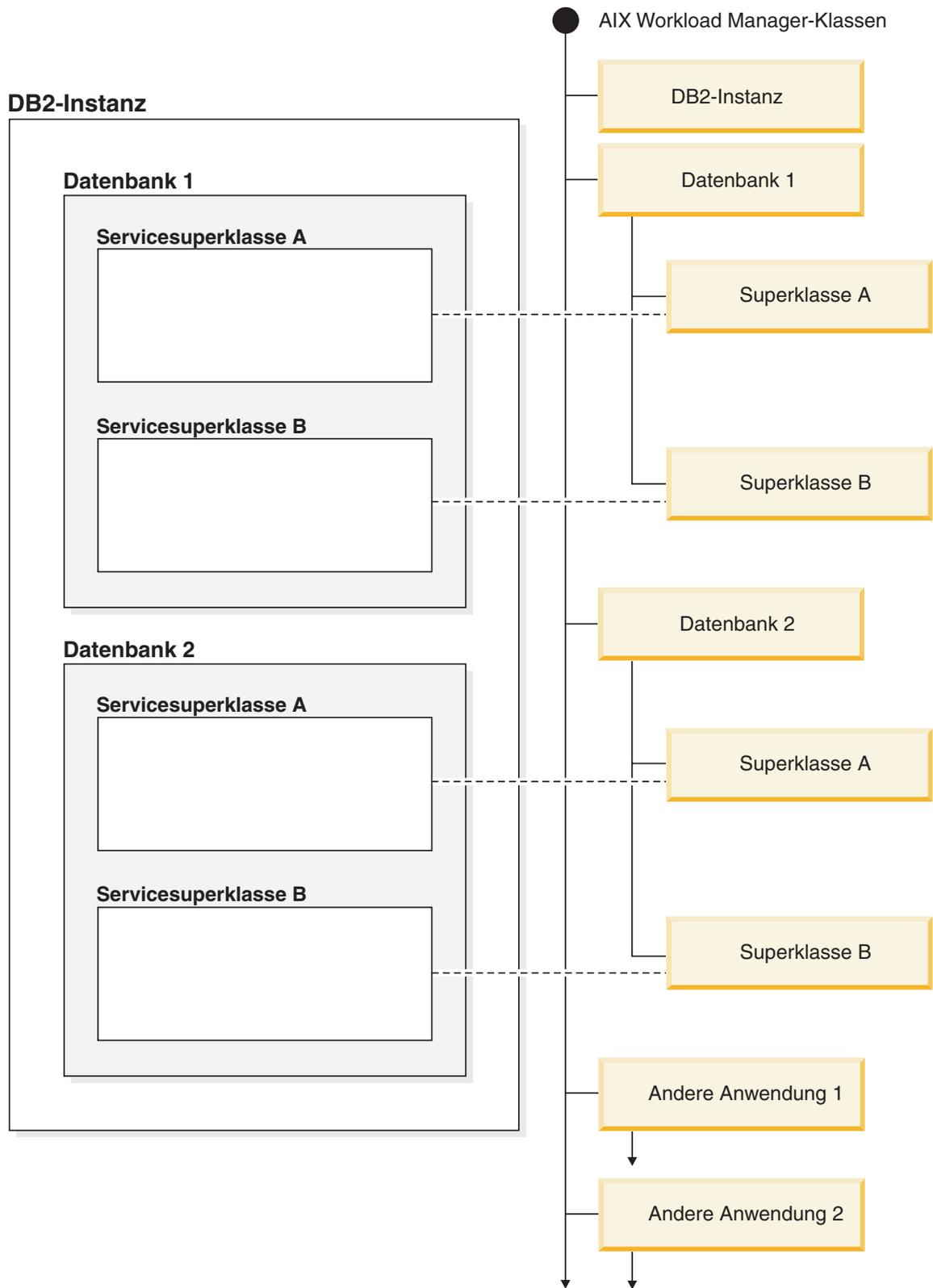


Abbildung 30. DB2-Serviceklassen, die AIX-Serviceklassen zugeordnet sind (nur mit DB2-Servicesuperklassen)

Eine alternative Konfiguration besteht darin, jede DB2-Servicesuperklasse ihrer eigenen Superklasse von AIX Workload Manager zuzuordnen, was in diesem Beispiel zu vier Superklassen führt. In dieser Situation wird die Datenbankebene der Ressourcensteuerung explizit in den Serviceklassendefinitionen von AIX Workload Manager dargestellt.

Die folgende Abbildung zeigt eine Möglichkeit, die 1:1-Zuordnung in Situationen zu erreichen, in denen Sie mehrere Datenbanken jeweils mit Servicesuperklassen und Serviceunterklassen haben. Hier wird jede Datenbank einer AIX-Superklasse und jede DB2-Serviceunterklasse einer Unterklasse von AIX Workload Manager zugeordnet. Die DB2-Servicesuperklasse wird in den Serviceklassendefinitionen von AIX Workload Manager nicht explizit gezeigt.

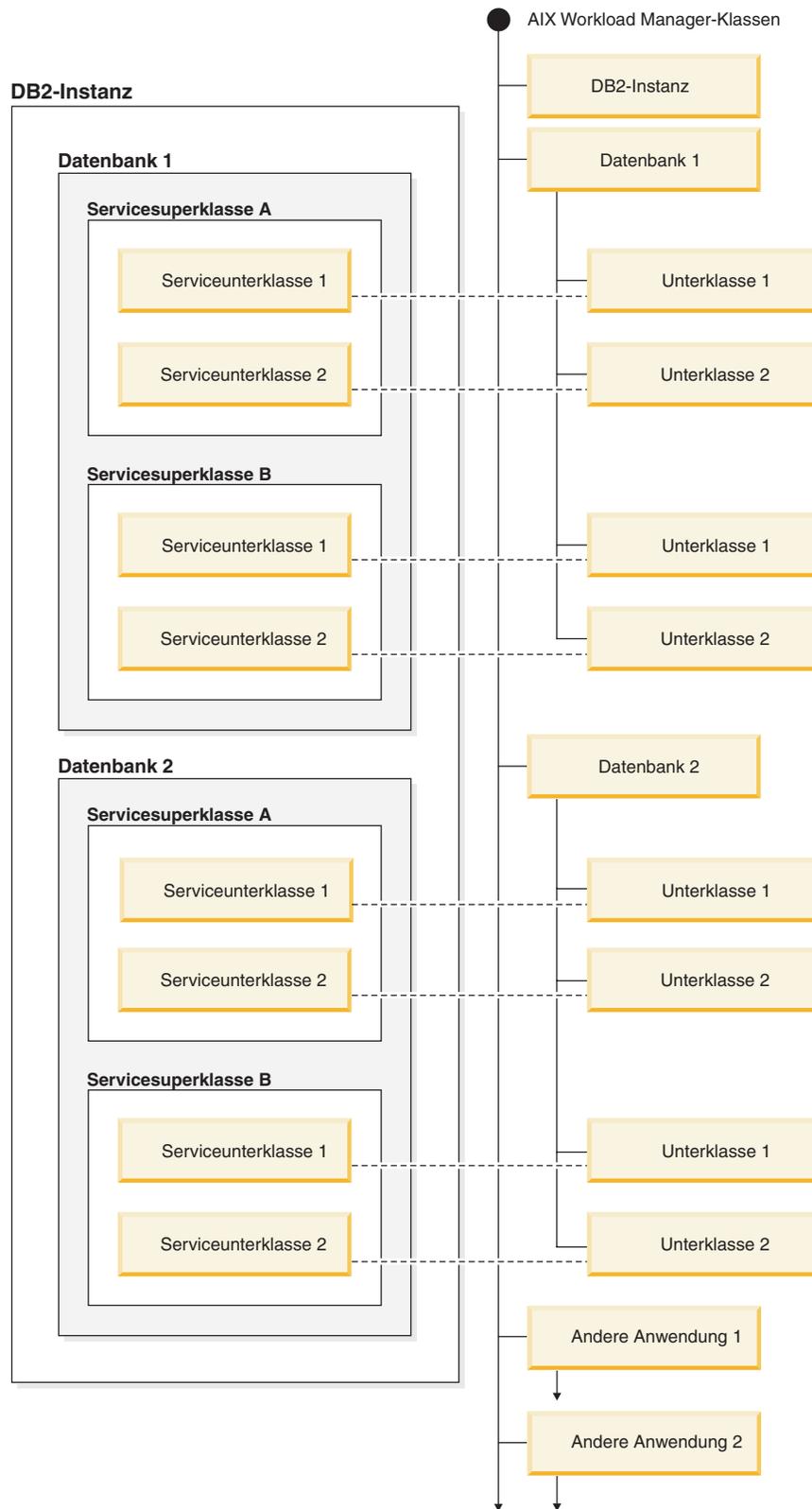


Abbildung 31. DB2-Serviceklassen, die Klassen von AIX Workload Manager zugeordnet sind (mit DB2-Serviceunterklassen)

Definieren von Zuordnungen zwischen DB2-Serviceklassen und Serviceklassen von AIX Workload Manager

Die Zuordnung zwischen DB2-Serviceklassen und Serviceklassen von AIX Workload Manager wird für die DB2-Serviceklasse mit dem Schlüsselwort `OUTBOUND CORRELATOR` der Anweisung `CREATE SERVICE CLASS` oder `ALTER SERVICE CLASS` angegeben.

Zum Einrichten der Serviceklassen von AIX Workload Manager mit dem DB2-Datenserver sind die folgenden Schritte auszuführen:

1. Erstellen Sie die DB2-Service-Superklassen und -Service-Unterklassen, und geben Sie die `OUTBOUND CORRELATOR`-Kennungen (Tags) an.
2. Erstellen Sie die entsprechenden AIX-Serviceklassen.
3. Erstellen Sie die zugehörigen Regeldateien von AIX Workload Manager, die die Zuordnungen von DB2-Workload-Management zu AIX Workload Manager enthalten. Verwenden Sie hierzu die `OUTBOUND CORRELATOR`-Kennungen in den Kennungsspalten.
4. Starten Sie AIX Workload Manager.
5. Setzen Sie gegebenenfalls diese Konfiguration von AIX Workload Manager auf aktiv.

Die folgenden Punkte erläutern, wie AIX Workload Manager Arbeit vom DB2-Datenserver nach der Änderung in DB2 Version 9.5 von einem Prozessmodell zu einem Threading-Modell unter UNIX und Linux handhabt. Wenn ein Thread einer DB2-Serviceklasse beitrifft, ruft der DB2-Datenserver die entsprechende API von AIX Workload Manager auf, um den Thread der entsprechenden AIX-Serviceklasse zuzuordnen. Der DB2-Datenserver sendet die AIX-Zielserviceklasse des Threads an AIX Workload Manager, indem er die im Parameter `OUTBOUND CORRELATOR` festgelegte Anwendungskennung übergibt.

Sie müssen sicherstellen, dass AIX Workload Manager korrekt installiert, konfiguriert und aktiv ist. Wenn der DB2-Datenserver nicht mit AIX Workload Manager kommunizieren kann, wird eine Nachricht in `db2diag.log` und im DB2-Administratorprotokoll protokolliert. Die Datenbankaktivität wird fortgesetzt.

Der DB2-Datenserver kann nicht erkennen, ob der Wert von `OUTBOUND CORRELATOR`, den er an AIX Workload Manager übergibt, von AIX Workload Manager erkannt wird. Sie müssen sicherstellen, dass der Wert, der für die DB2-Serviceklasse angegeben ist, den Anwendungskennungen entspricht, die DB2-Threads den AIX-Serviceklassen zuordnen. Wenn der Wert von `OUTBOUND CORRELATOR` von AIX Workload Manager nicht erkannt wird, wird die Datenbankaktivität fortgesetzt.

Andere wichtige Punkte:

- DB2-Serviceklassen können nicht die Vererbungsfunktion von AIX Workload Manager verwenden. Die Vererbung ist die Standardeinstellung für eine AIX-Serviceklasse. Die Vererbung muss explizit inaktiviert werden, indem das Vererbungsattribut auf `N0` gesetzt wird. Die Vererbung von AIX Workload Manager erzwingt die Zuordnung aller untergeordneten Threads und Prozesse zur selben Klasse, der auch der übergeordnete Thread oder Prozess zugeordnet ist. Wenn die Vererbung aktiviert ist, kann der DB2-Workload-Manager die AIX-Workload-Management-Klasse eines Threads mit Kennung nicht ändern. Durch diese Einschränkung wird jegliche Integration des DB2-Workload-Managers mit AIX Workload Manager unbrauchbar. Der DB2-Datenserver kann nicht erken-

nen, ob die Vererbung von AIX Workload Manager aktiviert ist, und setzt keine Fehlnachricht ab, wenn die Vererbung aktiviert ist.

- DB2-Serviceklassen sind nicht kompatibel zur manuellen Zuordnungsfunktion von AIX Workload Manager. Mit der manuellen Zuordnung können Benutzer einen Prozess manuell einer bestimmten Klasse von AIX Workload Manager zuordnen. Durch die manuelle Zuordnung des DB2-Prozesses werden alle Threads im Prozess einer Zielklasse von AIX Workload Manager zugeordnet. Dabei wird die DB2-Serviceklassen-Zuordnungslogik unterdrückt, und die Ergebnisse sind nicht vorhersagbar.
- AIX 5.3.H oder höher ist erforderlich.

Weitere Informationen zu AIX Workload Manager finden Sie im AIX Information Center unter <http://publib.boulder.ibm.com/infocenter/pseries/v5r3/index.jsp>.

Einstellen von CPU-Steuerungen auf den Serviceklassen von AIX Workload Manager

Mit Hilfe von AIX Workload Manager kann der CPU-Anteil gesteuert werden, der den einzelnen Serviceklassen zugeordnet wird. Es gibt u. a. die Möglichkeit, für jede Serviceklasse den minimalen, maximalen oder relativen CPU-Anteil festzulegen.

Bei der Integration von AIX Workload Manager mit dem DB2-Workload-Management wird nur die CPU-Zuordnungssteuerung unterstützt. Sie sollten keine Speicher- und E/A-Einstellungen für die AIX-Serviceklassen festlegen. DB2-Hauptspeicher auf Datenbankebene wird von allen Agenten von unterschiedlichen DB2-Serviceklassen gemeinsam genutzt. Sie können also die Hauptspeicherzuordnung nicht auf unterschiedliche Serviceklassen aufteilen. Die E/A-Steuerung der AIX-Version unterstützt nicht das neue Threading-Modell der DB2-Engine. Zur Steuerung der Ein-/Ausgabe können Sie das Attribut für Vorabesezugriffspriorität einer DB2-Serviceklasse verwenden, um unterschiedliche E/A-Prioritäten für unterschiedliche DB2-Serviceklassen anzugeben.

Wenn Sie AIX verwenden, um den einer Serviceklasse zugeordneten CPU-Anteil zu steuern, ändern Sie nicht gleichzeitig die Agentenprioritätseinstellung für diese DB2-Serviceklasse. Verwenden Sie nur einen dieser Mechanismen zur Steuerung des Zugriffs auf CPU-Ressourcen. Sie können für eine Serviceklasse nicht sowohl den Wert von AGENT PRIORITY als auch den Wert von OUTBOUND CORRELATOR festlegen. Weitere Informationen finden Sie in „CPU-Priorität und DB2-Serviceklassen“ auf Seite 33.

Einstellungen von AIX Workload Manager sollten auf allen physischen Computern, die zu einer Instanz gehören, konsistent sein. Wenn z. B. die Ressourceneinstellung für eine AIX-Serviceklasse auf einem Computer hoch eingestellt ist, sollte dieselbe Einstellung für diese AIX-Serviceklasse auf allen anderen Computern verwendet werden. Wenn die Einstellungen zur Ressourcennutzung über mehrere Computer inkonsistent sind, zeigen Anforderungen, die in derselben AIX-Serviceklasse ausgeführt werden, auf unterschiedlichen Datenbankpartitionen unterschiedliche Leistungswerte. Diese Situation kann zu einem schlechten Gesamtdurchsatz für Verbindungen in einer AIX-Serviceklasse führen.

Anhang C. Verarbeitung von gespeicherten Prozeduren in einer Workload-Management-Lösung

Jeder Aufruf einer gespeicherten Prozedur wird als eine Aktivität verarbeitet. Alle Datenbankaktivitäten, die von der gespeicherten Prozedur abgesetzt werden, werden als untergeordnete Aktivitäten der gespeicherten Prozedur verarbeitet. Wenn eine Zuordnungsarbeitsaktion auf eine gespeicherte Prozedur angewendet wird, können je nach Konfiguration untergeordnete Aktivitäten einer gespeicherten Prozedur in derselben Serviceunterklasse oder in anderen Serviceunterklassen als die übergeordnete Aktivität ausgeführt werden.

Schwellenwerte für den gemeinsamen Zugriff von Aktivitäten werden auf die gespeicherte Prozedur selbst und auf ihre untergeordneten Aktivitäten angewendet. Wenn die Ausführung der gespeicherten Prozedur in die Warteschlange gestellt wird, kann keine der zugehörigen untergeordneten Aktivitäten weiter verarbeitet werden. Wenn jedoch die Ausführung einer gespeicherten Prozedur beginnt, werden untergeordnete Aktivitäten eventuell in die Warteschlange gestellt.

Bei gespeicherten Prozeduren, die im abgeschirmten Modus ausgeführt werden, können Sie die Prozess-ID und die Thread-ID der Prozesse oder Threads der gespeicherten Prozedur mit der Tabellenfunktion `WLM_GET_SERVICE_CLASS_AGENTS` anzeigen.

Anhang D. Namenskonventionen

Für die Benennung aller Objekte, Benutzer und Gruppen gelten Regeln. Einige dieser Regeln sind für die Plattform spezifisch, auf der Sie arbeiten.

Es gibt zum Beispiel eine Regel für die Verwendung von Groß- und Kleinbuchstaben in einem Namen.

- Auf UNIX-Plattformen müssen Namen in Kleinbuchstaben angegeben werden.
- Auf Windows-Plattformen ist die Groß-, Kleinschreibung und gemischte Groß- und Kleinschreibung zulässig.

Wenn nichts anderes angegeben ist, dürfen alle Namen die folgenden Zeichen enthalten:

- A bis Z. In den meisten Namen werden die Zeichen A bis Z von Kleinbuchstaben in Großbuchstaben umgesetzt.
- 0 bis 9.
- ! % () { } . - ^ ~ _ (Unterstreichungszeichen) @, #, \$ und Leerzeichen.
- \ (Backslash).

Namen dürfen nicht mit einer Ziffer oder dem Unterstreichungszeichen beginnen.

Für SQL reservierte Wörter dürfen nicht als Namen von Tabellen, Sichten, Spalten, Indizes oder Berechtigungs-IDs verwendet werden.

In Abhängigkeit vom verwendeten Betriebssystem und dem Ort, an dem Sie mit der DB2-Datenbank arbeiten, sind möglicherweise weitere Sonderzeichen verfügbar, die ebenfalls verwendet werden können. Diese Zeichen können funktionieren, jedoch gibt es keine Garantie, dass sie funktionieren. Daher wird empfohlen, beim Benennen von Objekten in der Datenbank keine anderen als die oben aufgelisteten Sonderzeichen zu verwenden.

Für Benutzer- und Gruppennamen müssen die Regeln beachtet werden, die für bestimmte Betriebssysteme durch zugehörige Systeme erzwungen werden. Zum Beispiel sind auf Linux- und UNIX-Plattformen die folgenden Zeichen für Benutzernamen und Primärgruppennamen zulässig: a bis z in Kleinbuchstaben, 0 bis 9 und _ (Unterstreichungszeichen) für Namen, die nicht mit einer Ziffer zwischen 0 und 9 beginnen.

Längen dürfen die in SQL- und XML-Begrenzungen aufgeführten Längen nicht überschreiten.

Darüber hinaus müssen Sie auch die Namenskonventionen der einzelnen Objekte, die Namenskonventionen in einer Umgebung mit Unterstützung von Landessprachen (NLS) sowie die Namenskonventionen in einer Unicode-Umgebung beachten.

Einschränkungen für die Berechtigungs-ID (AUTHID): Ab Version 9.5 des DB2-Datenbanksystems können Sie eine 128-Byte-Berechtigungs-ID verwenden. Wenn die Berechtigungs-ID jedoch als Benutzer-ID oder Gruppename des Betriebssystems interpretiert wird, gelten die Einschränkungen des Betriebssystems (z. B. eine Begrenzung auf 8 oder 30 Zeichen für Benutzer-IDs und auf 30 Zeichen für Gruppennamen). Daher können Sie zwar eine 128-Byte-Berechtigungs-ID erteilen, jedoch ist es nicht möglich, als Benutzer, der diese Berechtigungs-ID hat, eine Ver-

bindung herzustellen. Wenn Sie ein eigenes Sicherheits-Plug-in schreiben, sind Sie in der Lage, die erweiterten Größen für die Berechtigungs-ID voll auszunutzen. Zum Beispiel können Sie an Ihr Sicherheits-Plug-in eine 30-Byte-Benutzer-ID übergeben, und das Plug-in kann eine 128-Byte-Berechtigungs-ID während der Authentifizierung zurückgeben, mit der Sie eine Verbindung herstellen können.

Anhang E. Rollen

Rollen vereinfachen die Verwaltung und das Management von Zugriffsrechten, indem sie eine äquivalente Funktionalität zu Gruppen bieten, jedoch ohne deren Einschränkungen. Eine Rolle ist ein Datenbankobjekt, das ein oder mehrere Zugriffsrechte zusammenfasst und Benutzern, Gruppen, PUBLIC oder anderen Rollen durch eine Anweisung GRANT oder einem gesicherten Kontext durch eine Anweisung CREATE TRUSTED CONTEXT oder ALTER TRUSTED CONTEXT erteilt werden kann. Eine Rolle kann für das Verbindungsattribut SESSION_USER ROLE in einer Auslastungsdefinition (Workloaddefinition) angegeben werden.

Rollen bieten verschiedene Vorteile, die das Management von Zugriffsrechten in einem Datenbanksystem vereinfachen:

- Sicherheitsadministratoren können den Zugriff auf ihre Datenbanken auf eine Weise steuern, die die Struktur ihrer Unternehmen widerspiegelt. (Sie können Rollen in der Datenbank erstellen, die die Aufgabenbereiche in ihren Unternehmen direkt abbilden.)
- Benutzern wird die Zugehörigkeit zu den Rollen erteilt, die ihren Zuständigkeiten entsprechen. Wenn sich Zuständigkeiten ändern, lässt sich die Rollen-zugehörigkeit problemlos neu erteilen und entziehen.
- Die Zuordnung von Zugriffsrechten wird vereinfacht. Anstatt die gleiche Gruppe von Zugriffsrechten jedem einzelnen Benutzer mit einem bestimmten Aufgabenbereich zu erteilen, kann der Administrator diese Gruppe von Zugriffsrechten einer Rolle erteilen, die diesen Aufgabenbereich darstellt, und anschließend diese Rolle jedem Benutzer mit diesem Aufgabenbereich erteilen.
- Die Zugriffsrechte einer Rolle können aktualisiert werden, und alle Benutzer, denen diese Rolle erteilt wurde, empfangen die Aktualisierung. Der Administrator braucht die Zugriffsrechte nicht für jeden Benutzer einzeln zu aktualisieren.
- Die Zugriffsrechte und Berechtigungen, die Rollen erteilt sind, werden bei jeder Erstellung von Sichten, Triggern, MQTs (Materialized Query Tables), statischem SQL und SQL-Routinen verwendet, während die Zugriffsrechte und Berechtigungen, die Gruppen (direkt oder indirekt) erteilt sind, nicht verwendet werden. Dies liegt daran, dass das DB2-Datenbanksystem nicht feststellen kann, wenn sich die Zugehörigkeit zu einer Gruppe ändert, da die Gruppe von der Software eines anderen Anbieters (z. B. dem Betriebssystem oder einem LDAP-Verzeichnis) verwaltet wird. Da Rollen innerhalb der Datenbank verwaltet werden, kann das DB2-Datenbanksystem feststellen, wenn sich eine Berechtigung ändert, und sich entsprechend verhalten. Rollen, die Gruppen erteilt sind, werden aus demselben Grund nicht berücksichtigt, aus dem Gruppen nicht berücksichtigt werden.
- Alle Rollen, die einem Benutzer zugeordnet sind, werden aktiviert, wenn dieser Benutzer eine Verbindung herstellt, so dass alle Zugriffsrechte und Berechtigungen, die Rollen erteilt sind, bei der Herstellung einer Verbindung durch einen Benutzer ausgewertet werden. Rollen können nicht explizit aktiviert oder inaktiviert werden.
- Der Sicherheitsadministrator kann das Management einer Rolle an andere Personen delegieren.

Alle DB2-Zugriffsrechte und -Berechtigungen, die innerhalb einer Datenbank erteilt werden können, können einer Rolle erteilt werden. Ausgenommen von dieser

Regel ist die Berechtigung SECADM (Sicherheitsadministrator). Zum Beispiel können einer Rolle beliebige der folgenden Berechtigungen und Zugriffsrechte erteilt werden:

- Die Datenbankberechtigungen DBADM, LOAD und IMPLICIT_SCHEMA
- Die Datenbankberechtigungen CONNECT, CREATETAB, CREATE_NOT_FENCED, BINDADD CREATE_EXTERNAL_ROUTINE oder QUIESCE_CONNECT
- Alle Zugriffsrechte für Datenbankobjekte (einschließlich CONTROL)

Die Rollen eines Benutzers werden automatisch aktiviert und bei der Berechtigungsverarbeitung berücksichtigt, wenn der Benutzer eine Verbindung zur Datenbank herstellt. Sie brauchen eine Rolle nicht durch die Anweisung SET ROLE zu aktivieren. Wenn Sie zum Beispiel eine Sicht, eine MQT, einen Trigger, ein Paket oder eine SQL-Routine erstellen, werden die Zugriffsrechte, über die Sie durch Rollen verfügen, angewendet. Die Zugriffsrechte, über die Sie durch Rollen verfügen, die Gruppen erteilt sind, deren Mitglied Sie sind, werden hingegen nicht angewendet.

Eine Rolle hat keinen Eigner. Der Sicherheitsadministrator kann mit Hilfe der Klausel WITH ADMIN OPTION der Anweisung GRANT das Management der Rolle an einen anderen Benutzer delegieren, so dass der andere Benutzer die Rollen-zugehörigkeit steuern kann.

Einschränkungen

Bei der Verwendung von Rollen sind einige Einschränkungen zu beachten:

- Eine Rolle kann nicht Eigner von Datenbankobjekten sein.
- Einer Rolle kann nicht die Berechtigung SECADM (Sicherheitsadministrator) erteilt werden.
- Berechtigungen und Rollen, die Gruppen erteilt sind, werden beim Erstellen der folgenden Datenbankobjekte nicht geprüft:
 - Pakete mit statischem SQL
 - Sichten
 - MQTs (Materialized Query Tables)
 - Trigger
 - SQL-Routinen

Beim Erstellen dieser Objekte werden nur die Rollen geprüft, die dem Benutzer, der das Objekt erstellt, oder PUBLIC direkt oder indirekt (z. B. durch eine Rollenhierarchie) erteilt sind.

Anhang F. Gesicherte Kontexte und Verbindungen

Ein gesicherter Kontext ist ein Datenbankobjekt, das eine Vertrauensbeziehung für eine Verbindung zwischen der Datenbank und einer externen Entität wie beispielsweise einem Anwendungsserver definiert.

Die Vertrauensbeziehung basiert auf der folgenden Gruppe von Attributen:

- Systemberechtigungs-ID: Ist dem Benutzer zugeordnet, der eine Datenbankverbindung herstellt.
- IP-Adresse (oder Domänenname): Ist dem Host zugeordnet, über den eine Datenbankverbindung hergestellt wird.
- Datenstromverschlüsselung: Stellt die Verschlüsselungseinstellung (sofern vorhanden) für die Datenübertragung zwischen dem Datenbankserver und dem Datenbankclient dar.

Wenn ein Benutzer eine Datenbankverbindung herstellt, überprüft das DB2-Datenbanksystem, ob die Verbindung mit der Definition eines Objekts für einen gesicherten Kontext in der Datenbank übereinstimmt. Wenn eine Übereinstimmung vorhanden ist, dann wird die Datenbankverbindung als gesichert eingestuft.

Eine gesicherte Verbindung ermöglicht dem Initiator dieser Verbindung das Anfordern weiterer Funktionen, die außerhalb der gesicherten Verbindung möglicherweise nicht verfügbar sind. Diese zusätzlichen Funktionen können abhängig davon, ob es sich um eine explizite oder implizite gesicherte Verbindung handelt, variieren.

Der Initiator einer expliziten gesicherten Verbindung verfügt über folgende Möglichkeiten:

- Wechseln der aktuellen Benutzer-ID der Verbindung und Angabe einer anderen Benutzer-ID mit oder ohne Authentifizierung.
- Anfordern zusätzlicher Zugriffsrechte über die Funktion für die Rollenvererbung in gesicherten Kontexten.

Eine implizite gesicherte Verbindung stellt eine gesicherte Verbindung dar, die nicht explizit angefordert wurde. Die implizite gesicherte Verbindung resultiert aus einer normalen Verbindungsanforderung und nicht aus der Anforderung einer expliziten gesicherten Verbindung. Zum Anfordern einer impliziten Verbindung sind keine Änderungen am Anwendungscode erforderlich. Darüber hinaus hat die Entscheidung zum Anfordern einer impliziten gesicherten Verbindung oder eines anderen Verbindungstyps keine Auswirkungen auf den Verbindungsrückkehrcode (wenn Sie eine explizite gesicherte Verbindung anfordern, gibt der Verbindungsrückkehrcode an, ob die Anforderung erfolgreich ausgeführt werden konnte oder nicht). Der Initiator einer impliziten gesicherten Verbindung kann zusätzliche Zugriffsrechte nur über die Funktion für die Rollenvererbung in gesicherten Kontexten anfordern. Ein Wechsel der Benutzer-ID ist hingegen nicht möglich.

Verbesserte Sicherheit durch Verwendung gesicherter Kontexte

Das auf drei Schichten basierende Anwendungsmodell erweitert das standardmäßige zweischichtige Client- und Servermodell durch eine Mittelschicht, die zwischen der Clientanwendung und dem Datenbankserver eingefügt wird. In den letzten Jahren fand dieses Modell mit dem verstärkten Einsatz webbasierter Technologien und der J2EE-Plattform (J2EE = Java 2 Enterprise Edition) zunehmend Ver-

breitung. Als Beispiel für ein Softwareprodukt, das das dreischichtige Anwendungsmodell unterstützt, kann IBM WebSphere Application Server (WAS) aufgeführt werden.

In einem dreischichtigen Anwendungsmodell ist die Mittelschicht für die Authentifizierung der Benutzer verantwortlich, die die Clientanwendungen ausführen, und darüber hinaus auch für die Verwaltung der Interaktion mit dem Datenbankserver. Traditionell erfolgt die gesamte Interaktion mit dem Datenbankserver über eine Datenbankverbindung, die von der Mittelschicht mit Hilfe einer Kombination aus Benutzer-ID und einem Berechtigungsnachweis hergestellt wird, mit der die Mittelschicht gegenüber dem Datenbankserver identifiziert wird. Dies bedeutet, dass der Datenbankserver für alle Berechtigungsprüfungs- und Protokollierungsoperationen, die für den Datenbankzugriff erforderlich sind, die Datenbankzugriffsrechte verwendet, die der Benutzer-ID der Mittelschicht zugeordnet sind. Dies gilt auch für die Zugriffsoperationen der Mittelschicht für einen Benutzer.

Obwohl das dreischichtige Anwendungsmodell zahlreiche Vorteile aufweist, verursacht die Ausführung der gesamten Interaktion mit dem Datenbankserver (z. B. eine Benutzeranforderung) unter der Berechtigungs-ID der Mittelschicht jedoch verschiedene Sicherheitsrisiken, die wie folgt zusammengefasst werden können:

- **Verlust der Benutzeridentität**
Für bestimmte Unternehmen ist es aus Gründen der Zugriffskontrolle wichtig, die Identität des Benutzers zu kennen, der auf die Datenbank zugreift.
- **Reduzierte Benutzerverantwortlichkeit**
Die Möglichkeit zur Feststellung der Verantwortlichkeit mit Hilfe entsprechender Protokollierungsoperationen ist ein Grundprinzip der Datenbanksicherheit. Dadurch, dass die Benutzeridentität nicht bekannt ist, wird die Unterscheidung von Transaktionen der Mittelschicht für eigene Zwecke und Transaktionen der Mittelschicht für einen Benutzer erheblich erschwert.
- **Gewährung nicht erforderlicher Zugriffsrechte für die Berechtigungs-ID der Mittelschicht**
Die Berechtigungs-ID der Mittelschicht muss über alle Zugriffsrechte verfügen, die zur Ausführung sämtlicher Anforderungen aller Benutzer erforderlich sind. Hierdurch entsteht das Sicherheitsproblem, dass Benutzern, die keinen Zugriff auf bestimmte Informationen benötigen, dieser Zugriff dennoch gewährt wird.
- **Beeinträchtigte Sicherheit**
Zusätzlich zum Problem der Zugriffsrechte, das im vorherigen Listenpunkt erläutert wurde, ist es bei diesem Ansatz erforderlich, dass die von der Mittelschicht für die Herstellung von Verbindungen verwendete Berechtigungs-ID über Zugriffsrechte für alle Ressourcen verfügen muss, auf die mit Benutzeranforderungen zugegriffen werden kann. Wenn diese Berechtigungs-ID der Mittelschicht in ihrer Sicherheit beeinträchtigt wird, dann sind alle diese Ressourcen einem erhöhten Sicherheitsrisiko ausgesetzt.
- **"Überlauf" zwischen Benutzern derselben Verbindung**
Änderungen eines vorherigen Benutzers können sich auf den aktuellen Benutzer auswirken.

Es besteht die Notwendigkeit zur Einrichtung von Verfahren, durch die die Identität und die Datenbankzugriffsrechte des aktuellen Benutzers für Datenbankanforderungen verwendet werden können, die von der Mittelschicht für diesen Benutzer ausgeführt werden. Der einfachste Ansatz zur Erreichung dieser Zielsetzung besteht darin, dass die Mittelschicht mit Hilfe der Benutzer-ID und des zugehörigen Kennworts eine neue Verbindung herstellt und die Benutzeranforderungen

dann über diese Verbindung leitet. Obwohl dieser Ansatz einfach ist, birgt er verschiedene Nachteile, die im Folgenden aufgeführt sind:

- Nichtanwendbarkeit für bestimmte Mittelschichten. Zahlreiche Server der Mittelschicht verfügen nicht über die Berechtigungsnachweise für die Benutzerauthentifizierung, die für die Herstellung einer Verbindung erforderlich sind.
- Leistungseinbußen. Mit der Herstellung einer neuen physischen Verbindung und der erneuten Authentifizierung des Benutzers auf dem Datenbankserver sind gewisse Leistungseinbußen verbunden.
- Verwaltungsaufwand. In bestimmten Fällen, in denen keine zentrale Sicherheitsfunktion definiert und eingesetzt wird oder in denen die einmalige Anmeldung (Single Sign-on) nicht verwendet wird, entsteht durch das Vorhandensein von zwei Benutzerdefinitionen (in der Mittelschicht und auf dem Server) ein erhöhter Verwaltungsaufwand. Hierdurch ist es erforderlich, das Kennwort an unterschiedlichen Stellen zu ändern.

Dieses Problem kann mit der Funktion für gesicherte Kontexte behoben werden. Der Sicherheitsadministrator kann in der Datenbank ein Objekt für einen gesicherten Kontext erstellen, mit dem eine Vertrauensbeziehung zwischen der Datenbank und der Mittelschicht definiert wird. Die Mittelschicht kann dann eine explizite gesicherte Verbindung zur Datenbank herstellen, wodurch sie die Möglichkeit erhält, die aktuelle Benutzer-ID für die Verbindung mit oder ohne Authentifizierung zu wechseln und eine andere Benutzer-ID zu verwenden. Zusätzlich zur Behebung des Problems mit der Identitätsprüfung für den Endbenutzer bieten gesicherte Kontexte einen weiteren Vorteil. Mit gesicherten Kontexten können Sie steuern, wann einem Datenbankbenutzer ein Zugriffsrecht gewährt wird. Wenn diese Möglichkeit nicht besteht, dann kann dies das gesamte Sicherheitskonzept des Systems beeinträchtigen. Zugriffsrechte können z. B. zu anderen Zwecken als den ursprünglich vorgesehenen verwendet werden. Der Sicherheitsadministrator kann einer Rolle ein oder auch mehrere Zugriffsrechte zuordnen und diese Rolle dann einem Objekt für einen gesicherten Kontext zuordnen. Nur gesicherte Datenbankverbindungen (explizit oder implizit), die mit der Definition dieses gesicherten Kontextes übereinstimmen, können die Zugriffsrechte, die dieser Rolle zugeordnet sind, nutzen.

Verbessern der Leistung

Bei Verwendung gesicherter Verbindungen können Sie die Leistung auf Grund der folgenden Vorteile maximieren:

- Beim Wechsel der aktuellen Benutzer-ID der Verbindung wird keine neue Verbindung hergestellt.
- Wenn in der Definition des gesicherten Kontextes die Authentifizierung der Benutzer-ID, zu der gewechselt werden soll, nicht als erforderlich definiert ist, dann entsteht der mit der Authentifizierung eines neuen Benutzers auf dem Datenbankserver verbundene Systemaufwand nicht.

Beispiel zum Erstellen eines gesicherten Kontextes

Der Sicherheitsadministrator erstellt das folgende Objekt eines gesicherten Kontextes:

```
CREATE TRUSTED CONTEXT CTX1
  BASED UPON CONNECTION USING SYSTEM AUTHID BENUTZER2
  ATTRIBUTES (ADDRESS '192.0.2.1')
  DEFAULT ROLE managerRole
  ENABLE
```

Wenn Benutzer *benutzer1* eine gesicherte Verbindung über die IP-Adresse 192.0.2.1 anfordert, gibt das DB2-Datenbanksystem eine Warnung (SQLSTATE 01679, SQLCODE +20360) zurück, um anzuzeigen, dass keine gesicherte Verbindung hergestellt werden konnte und dass dem Benutzer *benutzer1* nun eine nicht gesicherte Verbindung zugeordnet wurde. Wenn hingegen der Benutzer *benutzer2* eine gesicherte Verbindung über die IP-Adresse 192.0.2.1 anfordert, wird diese Anforderung ausgeführt, weil die Verbindungsattribute vom gesicherten Kontext CTX1 erfüllt werden. Nachdem der Benutzer *benutzer2* nun eine gesicherte Verbindung hergestellt hat, kann er nun alle Zugriffsrechte und Berechtigungen anfordern, die der Rolle *managerRole* des gesicherten Kontextes zugeordnet sind. Diese Zugriffsrechte und Berechtigungen stehen dem Benutzer *benutzer2* außerhalb des Geltungsbereichs dieser gesicherten Verbindung möglicherweise nicht zur Verfügung.

Anhang G. Übersicht über die technischen Informationen zu DB2

Die technischen Informationen zu DB2 stehen über die folgenden Tools und Methoden zur Verfügung:

- DB2-Informationszentrale
 - Themen (zu Tasks, Konzepten und Referenzinformationen)
 - Hilfe für DB2-Tools
 - Beispielprogramme
 - Lernprogramme
- DB2-Bücher
 - PDF-Dateien (für den Download verfügbar)
 - PDF-Dateien (auf der DB2-PDF-DVD)
 - Gedruckte Bücher
- Befehlszeilenhilfe
 - Hilfe für Befehle
 - Hilfe für Nachrichten

Anmerkung: Die Themen der DB2-Informationszentrale werden häufiger aktualisiert als die PDF- und Hardcopybücher. Um stets die neuesten Informationen zur Verfügung zu haben, sollten Sie die Dokumentationsaktualisierungen installieren, sobald diese verfügbar sind, oder die DB2-Informationszentrale unter ibm.com aufrufen.

Darüber hinaus können Sie auf zusätzliche technische Informationen zu DB2, wie beispielsweise technische Hinweise (Technotes), White Papers und IBM Redbooks, online über ibm.com zugreifen. Rufen Sie die Website 'DB2 Information Management - Software - Library' unter <http://www.ibm.com/software/data/sw-library/> auf.

Feedback zur Dokumentation

Senden Sie uns Ihr Feedback zur DB2-Dokumentation! Wenn Sie Anregungen zur Verbesserung der DB2-Dokumentation haben, senden Sie eine E-Mail an db2docs@ca.ibm.com. Das DB2-Dokumentationsteam bearbeitete das gesamte Feedback, kann jedoch nicht im Einzelnen auf Ihre E-Mails antworten. Nennen Sie uns, wenn möglich, konkrete Beispiele, sodass wir die Problemstellung besser beurteilen können. Wenn Sie uns Feedback zu einem bestimmten Thema oder einer bestimmten Hilfedatei senden, geben Sie den entsprechenden Titel sowie die URL an.

Verwenden Sie diese E-Mail-Adresse nicht, wenn Sie sich an die DB2-Kundenunterstützung wenden möchten. Wenn ein technisches Problem bei DB2 vorliegt, das Sie mit Hilfe der Dokumentation nicht beheben können, fordern Sie beim zuständigen IBM Service-Center Unterstützung an.

Bibliothek mit technischen Informationen zu DB2 im Hardcopy- oder PDF-Format

Die folgenden Tabellen enthalten eine Beschreibung der DB2-Bibliothek, die im IBM Publications Center unter www.ibm.com/shop/publications/order zur Verfügung steht. Über die folgende Adresse können Sie englische Handbücher im PDF-Format sowie übersetzte Versionen zu DB2 Version 9.5 herunterladen: www.ibm.com/support/docview.wss?rs=71&uid=swg2700947.

In den Tabellen sind die Bücher, die in gedrucktem Format zur Verfügung stehen, gekennzeichnet; möglicherweise sind diese in Ihrem Land oder Ihrer Region jedoch nicht verfügbar.

Tabelle 141. Technische Informationen zu DB2

Name	IBM Form	In gedrucktem Format verfügbar
<i>Administrative API Reference</i>	SC23-5842-00	Ja
<i>Administrative Routines and Views</i>	SC23-5843-00	Nein
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC23-5844-00	Ja
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC23-5845-00	Ja
<i>Command Reference</i>	SC23-5846-00	Ja
<i>Dienstprogramme für das Versetzen von Daten Handbuch und Referenz</i>	SC12-3917-00	Ja
<i>Datenrecovery und hohe Verfügbarkeit Handbuch und Referenz</i>	SC12-3919-00	Ja
<i>Datenserver, Datenbanken und Datenbankobjekte</i>	SC12-3912-00	Ja
<i>Datenbanksicherheit</i>	SC12-3914-00	Ja
<i>Developing ADO.NET and OLE DB Applications</i>	SC23-5851-00	Ja
<i>Developing Embedded SQL Applications</i>	SC23-5852-00	Ja
<i>Developing Java Applications</i>	SC23-5853-00	Ja
<i>Developing Perl and PHP Applications</i>	SC23-5854-00	Nein
<i>Developing User-defined Routines (SQL and External)</i>	SC23-5855-00	Ja
<i>Getting Started with Database Application Development</i>	GC23-5856-00	Ja
<i>Installation und Verwaltung von DB2 unter Linux und Windows - Erste Schritte</i>	GC12-3922-00	Ja
<i>Internationalisierung</i>	SC12-3916-00	Ja
<i>Fehlernachrichten, Band 1</i>	GI11-3098-00	Nein
<i>Fehlernachrichten, Band 2</i>	GI11-3099-00	Nein

Table 141. Technische Informationen zu DB2 (Forts.)

Name	IBM Form	In gedrucktem Format verfügbar
<i>Migration</i>	GC12-3921-00	Ja
<i>Net Search Extender Verwaltung und Benutzerhandbuch</i>	SC12-3979-00	Ja
Anmerkung: Der Inhalt dieses Dokuments ist nicht in der DB2-Informationszentrale enthalten.		
<i>Partitionierung und Clustering</i>	SC12-3915-00	Ja
<i>Query Patroller Verwaltung und Benutzerhandbuch</i>	SC12-3977-00	Ja
<i>IBM Data Server-Clients - Einstieg</i>	GC12-3924-00	Nein
<i>DB2-Server - Einstieg</i>	GC12-3923-00	Ja
<i>Spatial Extender und Geodetic Data Management Feature Benutzer- und Referenzhandbuch</i>	SC12-3978-00	Ja
<i>SQL Reference, Volume 1</i>	SC23-5861-00	Ja
<i>SQL Reference, Volume 2</i>	SC23-5862-00	Ja
<i>Systemmonitor Handbuch und Referenz</i>	SC12-3918-00	Ja
<i>Text Search Guide</i>	SC23-5866-00	Ja
<i>Fehlerbehebung</i>	GI11-3097-00	Nein
<i>Optimieren der Datenbankleistung</i>	SC12-3913-00	Ja
<i>Lernprogramm für Visual Explain</i>	SC12-3932-00	Nein
<i>Neue Funktionen</i>	SC12-3928-00	Ja
<i>Workload-Manager Handbuch und Referenz</i>	SC12-3929-00	Ja
<i>pureXML - Handbuch</i>	SC12-3930-00	Ja
<i>XQuery - Referenz</i>	SC12-3931-00	Nein

Table 142. Technische Informationen zu DB2 Connect

Name	IBM Form	In gedrucktem Format verfügbar
<i>DB2 Connect Personal Edition - Einstieg</i>	GC12-3926-00	Ja
<i>DB2 Connect-Server - Einstieg</i>	GC12-3927-00	Ja
<i>DB2 Connect Benutzerhandbuch</i>	SC12-3925-00	Ja

Table 143. Technische Informationen zu Information Integration

Name	IBM Form	In gedrucktem Format verfügbar
<i>Information Integration: Föderierte Systeme - Verwaltung</i>	SC12-3759-01	Ja

Table 143. Technische Informationen zu Information Integration (Forts.)

Name	IBM Form	In gedrucktem Format verfügbar
<i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1018-02	Ja
<i>Information Integration: Konfiguration föderierter Datenquellen</i>	SC12-3777-01	Nein
<i>Information Integration: SQL Replication - Handbuch und Referenz</i>	SC12-3782-01	Ja
<i>Information Integration: Replikation und Event-Publishing - Einführung</i>	GC12-3779-01	Ja

Bestellen gedruckter DB2-Bücher

Gedruckte DB2-Bücher können Sie in den meisten Ländern oder Regionen online bestellen. Das Bestellen gedruckter DB2-Bücher ist stets über den zuständigen IBM Ansprechpartner möglich. Beachten Sie hierbei bitte, dass einige Softcopybücher auf der DVD mit der *DB2-PDF-Dokumentation* nicht in gedruckter Form verfügbar sind. So sind beispielsweise die beiden Bände des Handbuchs *DB2 Fehlernachrichten* nicht in gedruckter Form erhältlich.

Gedruckte Versionen vieler DB2-Bücher, die auf der DVD mit der DB2-PDF-Dokumentation verfügbar sind, können gegen eine Gebühr bei IBM bestellt werden. Abhängig vom jeweiligen Land bzw. der jeweiligen Region können Sie Bücher möglicherweise online über das IBM Publications Center bestellen. Ist im jeweiligen Land bzw. der jeweiligen Region keine Onlinebestellung möglich, können Sie gedruckte DB2-Bücher stets über den zuständigen IBM Ansprechpartner bestellen. Nicht alle Bücher, die auf der DVD mit der DB2-PDF-Dokumentation verfügbar sind, können in gedruckter Form bestellt werden.

Anmerkung: Über <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5> haben Sie Zugriff auf die DB2-Informationszentrale, wo Sie die neueste und umfassendste DB2-Dokumentation finden.

Gehen Sie wie folgt vor, um gedruckte DB2-Bücher zu bestellen:

- Informationen dazu, ob in Ihrem Land oder Ihrer Region die Bestellung von gedruckten DB2-Büchern möglich ist, finden Sie auf der Website mit dem IBM Publications Center unter <http://www.ibm.com/shop/publications/order>. Wählen Sie ein Land, eine Region oder eine Sprache aus, um die Bestellinformationen für Veröffentlichungen aufzurufen, und führen Sie dann die entsprechenden Schritte des Bestellverfahrens für Ihr Land bzw. Ihre Region aus.
- Gehen Sie wie folgt vor, um gedruckte DB2-Bücher beim zuständigen IBM Ansprechpartner zu bestellen:
 1. Kontaktinformationen zum zuständigen Ansprechpartner finden Sie auf einer der folgenden Websites:
 - IBM Verzeichnis weltweiter Kontakte unter www.ibm.com/planetwide.
 - Website mit IBM Veröffentlichungen unter <http://www.ibm.com/shop/publications/order>. Wählen Sie das gewünschte Land, die gewünschte Region oder die gewünschte Sprache aus, um auf die entsprechende

Homepage mit Veröffentlichungen Ihres Landes bzw. Ihrer Region zuzugreifen. Folgen Sie auf dieser Seite dem Link für Informationen zu dieser Site ("About this Site").

2. Geben Sie bei Ihrem Anruf an, dass Sie eine DB2-Veröffentlichung bestellen möchten.
3. Teilen Sie dem zuständigen Ansprechpartner die Titel und Formularnummern der Bücher mit, die Sie bestellen möchten. Titel und Formularnummern finden Sie unter „Bibliothek mit technischen Informationen zu DB2 im Hardcopy- oder PDF-Format“ auf Seite 326.

Aufrufen der Hilfe für den SQL-Status über den Befehlszeilenprozessor

DB2 gibt für Bedingungen, die auf Grund einer SQL-Anweisung generiert werden können, einen SQLSTATE-Wert zurück. Die SQLSTATE-Hilfe erläutert die Bedeutung der SQL-Statuswerte und der SQL-Statusklassencodes.

Zum Aufrufen der Hilfe für SQL-Statuswerte müssen Sie den Befehlszeilenprozessor öffnen und Folgendes eingeben:

`? sqlstate` oder `? klassencode`

Hierbei steht *sqlstate* für einen gültigen fünfstelligen SQL-Statuswert und *klassencode* für die ersten beiden Ziffern dieses Statuswertes.

So kann beispielsweise durch die Eingabe von `? 08003` Hilfe für den SQL-Statuswert 08003 angezeigt werden, durch die Eingabe von `? 08` Hilfe für den Klassencode 08.

Zugriff auf verschiedene Versionen der DB2-Informationszentrale

Für Themen aus DB2 Version 9.5 lautet die URL der DB2-Informationszentrale <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>.

Für Themen aus DB2 Version 9 lautet die URL der DB2-Informationszentrale <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>.

Für Themen aus DB2 Version 8 lautet die URL der Informationszentrale (Version 8, 'Information - Unterstützung') <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>.

Anzeigen von Themen in der gewünschten Sprache in der DB2-Informationszentrale

In der DB2-Informationszentrale werden Themen, wenn möglich, in der Sprache angezeigt, die in den Vorgaben Ihres Browsers angegeben ist. Falls ein Thema nicht in die gewünschte Sprache übersetzt wurde, wird es in der DB2-Informationszentrale in Englisch angezeigt.

- Um Themen in der gewünschten Sprache im Browser 'Internet Explorer' anzuzeigen, gehen Sie wie folgt vor:
 1. Klicken Sie im Internet Explorer **Extras** —> **Internetoptionen...** —> **Sprachen...** an. Das Fenster **Spracheinstellung** wird geöffnet.
 2. Stellen Sie sicher, dass die gewünschte Sprache als erster Eintrag in der Liste angegeben ist.

- Klicken Sie den Knopf **Hinzufügen...** an, um eine neue Sprache zur Liste hinzuzufügen.

Anmerkung: Das Hinzufügen einer Sprache bedeutet nicht zwangsläufig, dass der Computer über die erforderlichen Schriftarten verfügt, um die Themen in der gewünschten Sprache anzuzeigen.

- Um eine Sprache an den Anfang der Liste zu verschieben, wählen Sie zunächst die gewünschte Sprache und anschließend den Knopf **Nach oben** aus, bis die Sprache an erster Stelle in der Liste steht.
3. Löschen Sie den Inhalt des Browser-Cache, und aktualisieren Sie anschließend die Seite, um die DB2-Informationszentrale in der gewünschten Sprache anzuzeigen.
- Um Themen in der gewünschten Sprache in einem Firefox- oder Mozilla-Browser anzuzeigen, gehen Sie wie folgt vor:
 1. Wählen Sie den Knopf im Bereich **Languages** des Dialogfensters **Tools** —> **Options** —> **Advanced** aus. Die Anzeige für die Auswahl der Sprache wird im Fenster mit den Einstellungen aufgerufen.
 2. Stellen Sie sicher, dass die gewünschte Sprache als erster Eintrag in der Liste angegeben ist.
 - Wenn Sie eine neue Sprache zur Liste hinzufügen möchten, klicken Sie den Knopf **Add...** an, um eine Sprache im entsprechenden Fenster auszuwählen.
 - Um eine Sprache an den Anfang der Liste zu verschieben, wählen Sie zunächst die gewünschte Sprache und anschließend den Knopf **Move Up** aus, bis die Sprache an erster Stelle in der Liste steht.
 3. Löschen Sie den Inhalt des Browser-Cache, und aktualisieren Sie anschließend die Seite, um die DB2-Informationszentrale in der gewünschten Sprache anzuzeigen.

Bei einigen Kombinationen aus Browser und Betriebssystem müssen Sie möglicherweise auch die Ländereinstellungen des Betriebssystems in die gewünschte Locale und Sprache ändern.

Aktualisieren der auf Ihrem Computer oder Intranet-Server installierten DB2-Informationszentrale

Wenn Sie die DB2-Informationszentrale lokal installiert haben, können Sie die gegebenenfalls von IBM bereitgestellten Aktualisierungen herunterladen und installieren.

Zur Aktualisierung der lokal installierten DB2-Informationszentrale sind die folgenden Schritte erforderlich:

1. Stoppen Sie die DB2-Informationszentrale auf Ihrem Computer, und starten Sie die Informationszentrale im Standalone-Modus erneut. Durch die Ausführung der Informationszentrale im Standalone-Modus wird verhindert, dass andere Benutzer in Ihrem Netz auf die Informationszentrale zugreifen, und ermöglicht Ihnen das Herunterladen und Anwenden von Aktualisierungen.
2. Verwenden Sie die Aktualisierungsfunktion, um zu prüfen, welche Aktualisierungen verfügbar sind. Falls Aktualisierungen verfügbar sind, die Sie installieren möchten, können Sie die Aktualisierungsfunktion verwenden, um diese herunterzuladen und zu installieren.

Anmerkung: Wenn es in der verwendeten Umgebung erforderlich ist, die Aktualisierungen für die DB2-Informationszentrale auf einer Maschine zu installieren, die nicht über eine Verbindung zum Internet verfügt, müssen Sie die Aktualisierungssite auf ein lokales Dateisystem spiegeln und dabei eine Maschine verwenden, die mit dem Internet verbunden ist und auf der die DB2-Informationszentrale installiert ist. Wenn viele Benutzer Ihres Netzes die Dokumentationsaktualisierungen installieren sollen, können Sie die Zeit, die jeder einzelne Benutzer für die Aktualisierungen benötigt, reduzieren, indem Sie die Aktualisierungssite lokal spiegeln und ein Proxy dafür erstellen. Wenn Aktualisierungspakete verfügbar sind, verwenden Sie die Aktualisierungsfunktion, um die Pakete herunterzuladen. Die Aktualisierungsfunktion ist jedoch nur im Standalone-Modus verfügbar.

3. Stoppen Sie die im Standalone-Modus gestartete Informationszentrale, und starten Sie die DB2-Informationszentrale auf Ihrem Computer erneut.

Anmerkung: Unter Windows Vista müssen Sie zur Ausführung der nachfolgend aufgeführten Befehle über Administratorberechtigung verfügen. Zum Starten einer Eingabeaufforderung oder eines Grafiktools mit vollen Administratorberechtigungen klicken Sie mit der rechten Maustaste auf die Verknüpfung, und wählen Sie **Als Administrator ausführen** aus.

Gehen Sie wie folgt vor, um die auf Ihrem Computer bzw. Intranet-Server installierte DB2-Informationszentrale zu aktualisieren:

1. Stoppen Sie die DB2-Informationszentrale.
 - Unter Windows: Klicken Sie **Start** → **Einstellungen** → **Systemsteuerung** → **Verwaltung** → **Dienste** an. Klicken Sie mit der rechten Maustaste die **DB2-Informationszentrale** an, und wählen Sie **Stoppen** aus.
 - Unter Linux: Geben Sie den folgenden Befehl ein:

```
/etc/init.d/db2icdv95 stop
```
2. Starten Sie die Informationszentrale im Standalone-Modus.
 - Unter Windows:
 - a. Öffnen Sie ein Befehlsfenster.
 - b. Navigieren Sie zu dem Pfad, in dem die Informationszentrale installiert ist. Standardmäßig ist die DB2-Informationszentrale im Verzeichnis <Programme>\IBM\DB2 Information Center\Version 9.5 installiert, wobei <Programme> das Verzeichnis der Programmdateien (Program Files) angibt.
 - c. Navigieren Sie vom Installationsverzeichnis in das Verzeichnis "doc\bin".
 - d. Führen Sie die Datei help_start.bat aus:

```
help_start.bat
```
 - Unter Linux:
 - a. Navigieren Sie zu dem Pfad, in dem die Informationszentrale installiert ist. Standardmäßig ist die DB2-Informationszentrale im Verzeichnis /opt/ibm/db2ic/V9.5 installiert.
 - b. Navigieren Sie vom Installationsverzeichnis in das Verzeichnis "doc/bin".
 - c. Führen Sie das Script help_start aus:

```
help_start
```

Der standardmäßig auf dem System verwendete Web-Browser wird aufgerufen und zeigt die Standalone-Informationszentrale an.

3. Klicken Sie den Aktualisierungsknopf (🔄) an. Klicken Sie im rechten Fenster der Informationszentrale den Knopf für die Suche nach Aktualisierungen an. Eine Liste der Aktualisierungen für die vorhandene Dokumentation wird angezeigt.
4. Wählen Sie zum Initiieren des Downloadprozesses die gewünschten Aktualisierungen aus, und klicken Sie anschließend den Knopf für die Installation der Aktualisierungen an.
5. Klicken Sie nach Abschluss des Download- und Installationsprozesses Fertig stellen an.
6. Stoppen Sie die im Standalone-Modus gestartete Informationszentrale.
 - Unter Windows: Navigieren Sie in das Verzeichnis "doc\bin" des Installationsverzeichnisses, und führen Sie die Datei help_end.bat aus:
help_end.bat

Anmerkung: Die Stapeldatei help_end enthält die Befehle, die erforderlich sind, um die Prozesse, die mit der Stapeldatei help_start gestartet wurden, ordnungsgemäß zu beenden. Verwenden Sie nicht die Tastenkombination Strg+C oder eine andere Methode, um help_start.bat zu beenden.

- Unter Linux: Navigieren Sie in das Verzeichnis "doc/bin" des Installationsverzeichnisses, und führen Sie das Script help_end aus:
help_end

Anmerkung: Das Script help_end enthält die Befehle, die erforderlich sind, um die Prozesse, die mit dem Script help_start gestartet wurden, ordnungsgemäß zu beenden. Verwenden Sie keine andere Methode, um das Script help_start zu beenden.

7. Starten Sie die DB2-Informationszentrale erneut.
 - Unter Windows: Klicken Sie **Start** → **Einstellungen** → **Systemsteuerung** → **Verwaltung** → **Dienste** an. Klicken Sie mit der rechten Maustaste die **DB2-Informationszentrale** an, und wählen Sie **Start** aus.
 - Unter Linux: Geben Sie den folgenden Befehl ein:
/etc/init.d/db2icdv95 start

In der aktualisierten DB2-Informationszentrale werden die neuen und aktualisierten Themen angezeigt.

DB2-Lernprogramme

Die DB2-Lernprogramme unterstützen Sie dabei, sich mit den unterschiedlichen Aspekten der DB2-Produkte vertraut zu machen. Die Lerneinheiten bieten eine in einzelne Schritte unterteilte Anleitung.

Vorbereitungen

Die XHTML-Version des Lernprogramms kann über die Informationszentrale unter <http://publib.boulder.ibm.com/infocenter/db2help/> angezeigt werden.

In einigen der Lerneinheiten werden Beispieldaten und Codebeispiele verwendet. Informationen zu bestimmten Voraussetzungen für die Ausführung der Tasks finden Sie in der Beschreibung des Lernprogramms.

DB2-Lernprogramme

Klicken Sie zum Anzeigen des Lernprogramms den Titel an.

„pureXML“ in *pureXML - Handbuch*

Einrichten einer DB2-Datenbank, um XML-Daten zu speichern und Basisoperationen mit dem nativen XML-Datenspeicher auszuführen.

„Visual Explain“ in *Lernprogramm für Visual Explain*

Analysieren, Optimieren und Anpassen von SQL-Anweisungen zur Leistungsverbesserung mit Hilfe von Visual Explain.

Informationen zur Fehlerbehebung in DB2

Eine breite Palette verschiedener Informationen zur Fehlerbestimmung und Fehlerbehebung steht zur Verfügung, um Sie bei der Verwendung von DB2-Produkten zu unterstützen.

DB2-Dokumentation

Informationen zur Fehlerbehebung stehen im Handbuch 'DB2-Fehlerbehebung' oder im Abschnitt zur Unterstützung und Fehlerbehebung der DB2-Informationszentrale zur Verfügung. Dort finden Sie Informationen dazu, wie Sie Probleme mit Hilfe der DB2-Diagnosetools und -Dienstprogramme eingrenzen und identifizieren können, Lösungen für einige der häufigsten Probleme sowie weitere Hinweise zur Behebung von Fehlern und Problemen, die bei der Verwendung der DB2-Produkte auftreten können.

DB2-Website mit technischer Unterstützung

Auf der DB2-Website mit technischer Unterstützung finden Sie Informationen zu Problemen und den möglichen Ursachen und Fehlerbehebungsmaßnahmen. Die Website mit technischer Unterstützung enthält Links zu den neuesten DB2-Veröffentlichungen, technischen Hinweisen (TechNotes), APARs (Authorized Program Analysis Reports) und Fehlerkorrekturen, Fixpacks sowie weiteren Ressourcen. Sie können diese Wissensbasis nach möglichen Lösungen für aufgetretene Probleme durchsuchen.

Rufen Sie die DB2-Website mit technischer Unterstützung unter <http://www.ibm.com/software/data/db2/udb/support.html> auf.

Bedingungen

Die Berechtigungen zur Nutzung dieser Veröffentlichungen werden Ihnen auf der Basis der folgenden Bedingungen gewährt.

Persönliche Nutzung: Sie dürfen diese Veröffentlichungen für Ihre persönliche, nicht kommerzielle Nutzung unter der Voraussetzung vervielfältigen, dass alle Eigentumsvermerke erhalten bleiben. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM nicht weitergeben, anzeigen oder abgeleitete Werke davon erstellen.

Kommerzielle Nutzung: Sie dürfen diese Veröffentlichungen nur innerhalb Ihres Unternehmens und unter der Voraussetzung, dass alle Eigentumsvermerke erhalten bleiben, vervielfältigen, weitergeben und anzeigen. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM außerhalb Ihres Unternehmens nicht vervielfältigen, weitergeben, anzeigen oder abgeleitete Werke davon erstellen.

Abgesehen von den hier gewährten Berechtigungen erhalten Sie keine weiteren Berechtigungen, Lizenzen oder Rechte (veröffentlicht oder stillschweigend) in Bezug auf die Veröffentlichungen oder darin enthaltene Informationen, Daten, Software oder geistiges Eigentum.

IBM behält sich das Recht vor, die in diesem Dokument gewährten Berechtigungen nach eigenem Ermessen zurückzuziehen, wenn sich die Nutzung der Veröffentlichungen für IBM als nachteilig erweist oder wenn die obigen Nutzungsbestimmungen nicht genau befolgt werden.

Sie dürfen diese Informationen nur in Übereinstimmung mit allen anwendbaren Gesetzen und Vorschriften, einschließlich aller US-amerikanischen Exportgesetze und Verordnungen, herunterladen und exportieren.

IBM übernimmt keine Gewährleistung für den Inhalt dieser Informationen. Diese Veröffentlichungen werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne eine ausdrückliche oder stillschweigende Gewährleistung für die Handelsüblichkeit, die Verwendungsfähigkeit oder die Freiheit der Rechte Dritter zur Verfügung gestellt.

Anhang H. Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. An Stelle der Produkte, Programme oder Services können auch andere ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder andere Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb der Produkte, Programme oder Dienstleistungen in Verbindung mit Fremdprodukten und Fremddienstleistungen liegt beim Kunden, soweit nicht ausdrücklich solche Verbindungen erwähnt sind.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Dieses Dokument enthält möglicherweise Links oder Verweise auf Websites und Ressourcen anderer Anbieter. Es bestehen keine Zusicherungen, Gewährleistungen oder Verpflichtungen von IBM hinsichtlich der Websites oder Ressourcen anderer Anbieter, auf die im vorliegenden Dokument verwiesen wird, Zugriff besteht oder Links vorhanden sind. Ein Link auf eine Website eines anderen Anbieters bedeutet nicht, dass IBM den Inhalt und die Verwendung dieser Website billigt oder deren Eigentümer anerkennt. Darüber hinaus ist IBM nicht an Transaktionen beteiligt und übernimmt keine Verantwortung für Transaktionen zwischen Ihnen und anderen Anbietern, auch wenn die Informationen (oder Links) zu diesen Anbietern auf einer IBM Website zur Verfügung stehen. IBM ist nicht für die Verfügbarkeit solcher externen Sites oder Ressourcen verantwortlich und übernimmt keine Verantwortung oder Haftung für Inhalte, Services, Produkte oder sonstiges Material, die bzw. das auf diesen oder über diese Sites oder Ressourcen verfügbar sind. Die Software anderer Anbieter unterliegt den Lizenzbedingungen der jeweiligen Software.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Dokument aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung sowie der Allgemeinen Geschäftsbedingungen von IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Garantie, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden, Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellsprache geschrieben sind. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (*Name Ihrer Firma*) (*Jahr*). Teile des vorliegenden Codes wurden aus Musterprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. *_Jahr/Jahre angeben_*. Alle Rechte vorbehalten.

Marken

Namen von Unternehmen, Produkten oder Services in den Dokumenten der Dokumentationsbibliothek von DB2 Version 9.5 können Marken oder Servicemarken der International Business Machines Corporation oder anderer Unternehmen sein. Informationen zu den Marken der IBM Corporation in den USA und/oder anderen Ländern finden Sie unter <http://www.ibm.com/legal/copytrade.shtml>.

Folgende Namen sind Marken oder eingetragene Marken anderer Unternehmen und wurden in mindestens einem der Dokumente in der DB2-Dokumentationsbibliothek verwendet.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Intel, das Intel-Logo, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken der Intel Corporation in den USA und/oder anderen Ländern.

Java und alle Java-basierten Marken sind Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und/oder anderen Ländern.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Adobe, das Adobe-Logo, PostScript und das PostScript-Logo sind eingetragene Marken oder Marken von Adobe Systems Incorporated in den USA und/oder anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicennamen können Marken anderer Hersteller sein.

Index

A

- Abfragen
 - Monitorelemente
 - queue_assignments_total 267
 - queue_size_top 268
 - queue_time_total 268
- ACTIVITYTOTALTIME, Aktivitätsschwellenwert 67
- Agenten
 - Agentenpriorität und Serviceklassen 33
 - Nutzung nach Serviceklasse untersuchen 192
- AIX Workload Manager
 - CPU-Priorität 33
- Aktivieren einer Workload 56
- Aktivierungszeit
 - Monitorelemente
 - last_wlm_reset 265
- Aktivitäten
 - abbrechen 146
 - abbrechen, Beispiel 188
 - Analyse, Beispiel 187
 - Arbeitsaktionen anwenden 99
 - blockierte Aktivitäten ermitteln 188
 - Daten erfassen zu 143
 - Details mit WLM_CAPTURE_ACTIVITY_IN_PROGRESS erfassen 126
 - für spätere Analyse erfassen, Beispiel 190
 - Informationen in den Designadvisor importieren 145
 - mit WLM_CANCEL_ACTIVITY abbrechen 126
 - Monitorelemente
 - act_total 252
 - activity_collected 249
 - activity_id 250
 - activity_secondary_id 250
 - activity_type 251
 - coord_act_aborted_total 255
 - coord_act_completed_total 256
 - coord_act_rejected_total 257
 - niedriger geschätzter Aufwand und hohe Laufzeit 200
 - Rogue 146
 - Status in Serviceklasse 34
 - Übersicht 20
 - verschachtelt 20
 - zu Arbeitsklassen zuordnen 98
 - zu Serviceklassen zuordnen 29
- Aktivitäten, Ereignismonitor 18
- Aktivitätsschwellenwerte 63
- Aktualisierungen
 - DB2-Informationszentrale 330
 - Informationszentrale 330
- Ändern
 - Schwellenwert 82
 - Serviceklassen 38
 - Workload 54
- Anwendungsprogrammierschnittstellen
 - sqleseti
 - Workloadzuordnung 158
- Arbeitsaktionen
 - ändern 110
 - auf Aktivität anwenden 99
 - Beispiel 89
 - Domäne des Arbeitsaktionssets 94
 - Arbeitsaktionen (*Forts.*)
 - erstellen 107
 - inaktivieren 112
 - löschen 112
 - unterstützte Schwellenwerte 97
 - Zweck 85
- Arbeitsaktionssets
 - ändern 105
 - Arbeitsaktion
 - unterstützte Schwellenwerte 97
 - Beispiel 174
 - Domäne und zulässige Arbeitsaktionen 94
 - erstellen 104
 - inaktivieren 106
 - löschen 106
 - Übersicht 87
 - zur Bestimmung der Typen der ausgeführten Arbeit verwenden 176
 - Zweck 85
- Arbeitsklassen
 - Aktivitäten zuordnen 98
 - ALL, Schlüsselwort, Beispiel 172
 - ändern 117
 - Arbeitstypzuordnung zu SQL-Anweisungen 91
 - Auswertungsreihenfolge 93
 - erstellen 114
 - löschen 117
 - Übersicht 7
 - unterstützte Schwellenwerte für Arbeitsaktionen 97
 - Zweck 85
- Arbeitsklassen, Beispiel 89
- Arbeitsklassensets
 - ändern 113
 - Arbeitsklasse
 - Auswertungsreihenfolge 93
 - CALL verwalten, Beispiel 171
 - erstellen 113
 - löschen 114
 - Übersicht 85
 - Zweck 85
- Ausführung
 - Monitorelemente
 - act_exec_time 252
- Auslastungen
 - Monitorelemente
 - wlo_completed_total 280
 - workload_id 282
 - workload_name 282
 - workload_occurrence_id 283
 - Auswertungsreihenfolge von Schwellenwerten 74
 - Auswertungsreihenfolge von Workloads 48

B

- Bedingungen
 - Verwendung der Veröffentlichungen 333
- Befehle
 - SET WORKLOAD 51, 285
- Beispiele
 - Arbeitsaktionsset und Schwellenwert 174

- Beispiele (*Forts.*)
 - Arbeitsklassen
 - ALL, Schlüsselwort 172
 - Arbeitsklassensetverwaltung von CALL 171
 - Serviceklassen 153
 - Verbindungen zu Workloads zuordnen 162
- Bemerkungen 335
- Bereiche
 - Monitorelemente
 - unterer Grenzwert 253
- Bestellen von DB2-Büchern 328
- Betriebssystem
 - Integration von DB2-Workload-Management 305
- Bins
 - Zweck 182

C

- CALL, Anweisung
 - Klassifizierung nach Schema 85
- CONCURRENTDBCOORDACTIVITIES, zusammengefasster Schwellenwert 73
- CONCURRENTWORKLOADACTIVITIES, zusammengefasster Schwellenwert 71
- CONCURRENTWORKLOADOCCURRENCES, zusammengefasster Schwellenwert 70
- CONNECTIONIDLETIME, Aktivitätsschwellenwert 64

D

- Data-Warehouse
 - Agent
 - Nutzung nach Serviceklasse untersuchen 192
 - Aktivitäten 20
 - Daten erfassen zu 143
 - zu Serviceklassen zuordnen 29
 - Aktivitätsschwellenwerte
 - abbrechen 146
 - Analyse, Beispiel 187
 - Arbeitsaktion anwenden auf 99
 - blockierte Aktivitäten ermitteln 188
 - für spätere Analyse erfassen, Beispiel 190
 - Informationen in den Designadvisor importieren 145
 - niedriger geschätzter Aufwand und hohe Laufzeit 200
 - Rogue 146
 - zu Arbeitsklassen zuordnen 98
 - Arbeitsaktion
 - ändern 110
 - auf Aktivität anwenden 99
 - inaktivieren 112
 - löschen 112
 - unterstützte Schwellenwerte 97
 - Arbeitsaktionen
 - Domäne des Arbeitsaktionssets 94
 - erstellen 107
 - Arbeitsaktionsset
 - ändern 105
 - Beispiel 174
 - CALL verwalten, Beispiel 171
 - erstellen 104
 - inaktivieren 106
 - löschen 106
 - zur Bestimmung der Typen der ausgeführten Arbeit verwenden 176
 - Arbeitsaktionssets
 - Übersicht 87

- Data-Warehouse (*Forts.*)
 - Arbeitsklasse
 - ändern 117
 - Auswertungsreihenfolge 93
 - Klassifizierung der Anweisung CALL nach Schema 85
 - löschen 117
 - Übersicht 7
 - Arbeitsklassen
 - ALL, Schlüsselwort, Beispiel 172
 - Arbeitsstypzuordnung zu SQL-Anweisungen 91
 - erstellen 114
 - Arbeitsklassenset
 - ändern 113
 - CALL verwalten, Beispiel 171
 - erstellen 113
 - löschen 114
 - Arbeitsklassensets 85
 - auf unterschiedlichen Ebenen überwachen 177
 - Ereignismonitor 127
 - Erkennung von Aktivitäten 20
 - Histogramme
 - Zweck 133, 182
 - Histogrammschablone
 - ändern 148
 - erstellen 148
 - Integration von DB2- und Betriebssystem-Workload-Management 305
 - optimieren
 - mit Kapazitätsplanungsdaten, Beispiel 193
 - ohne Kapazitätsplanungsdaten, Beispiel 195
 - Schwellenwert
 - ändern 82
 - Beispiel 174
 - erstellen 76
 - Verstöße überwachen 142
 - Zusammenfassung 63
 - Schwellenwerte
 - ACTIVITYTOTALTIME 67
 - Aktivitätsschwellenwerte 63
 - Auswertungsreihenfolge 74
 - Bereichsauflösung von Aktivitätsschwellenwerten 68
 - CONCURRENTDBCOORDACTIVITIES 73
 - CONCURRENTWORKLOADACTIVITIES 71
 - CONCURRENTWORKLOADOCCURRENCES 70
 - CONCURRENTWORKLOADOCCURRENCES, TOTALDBPARTITIONCONNECTIONS und TOTALSCPARTITIONCONNECTIONS zusammen verwendet 170
 - CONNECTIONIDLETIME 64
 - ESTIMATEDSQLCOST 64
 - SQLROWSRETURNED 66
 - SQLTEMPSPACE 65
 - TOTALDBPARTITIONCONNECTIONS 68
 - TOTALSCPARTITIONCONNECTIONS 69
 - Verwendungsbeispiel 169
 - zusammengefasste Schwellenwerte 63
 - Zweck 61
 - Serviceklasse
 - Leistungsminderung analysieren, Beispiel 184
 - Zeitpunktstatistikdaten ermitteln 180
 - Serviceklassenhierarchie 25
 - Standardworkloads 49
 - Statistikdaten
 - mit Statistikereignismonitor erfassen 137
 - Tabellenfunktion
 - Daten zusammenfassen 181

- Data-Warehouse (*Forts.*)
 - Tabellenfunktion (*Forts.*)
 - ergänzt mit Überwachungsprogramm für Momentaufnahme 125
 - Tabellenfunktionen 124, 140
 - Übersicht 85
 - Überwachung
 - Übersicht 121
 - Verbindungen
 - zu Workloads zuordnen 162
 - Workload
 - Leistungsminde­rung analysieren, Beispiel 186
 - Übersicht 6
 - Workload-Management 121
 - Agentenpriorität 33
 - Arbeitsaktionen, Beispiel 89
 - Arbeitsklassen, Beispiel 89
 - DDL-Aspekte 303
 - Echtzeitüberwachung 15
 - Ermittlungsphase 5
 - gespeicherte Prozeduren 315
 - Konzepte, Überblick 3
 - Langzeitüberwachung 18
 - Leistungsmodellierung 147
 - Objekteigentumsrecht 303
 - Phasen 3
 - Serviceklassen 25
 - Serviceklassen, Beispiel 153
 - Statistikdaten für Objekte 128
 - Statistikdaten zurücksetzen 141
 - Tabellenfunktion, Nutzungsbeispiel 17
 - Überwachungsphase, Überblick 14
 - Verbindungsattribute zum Zuordnen zur Workload 43
 - Verwaltung von Arbeit, Überblick 9
 - Vorab­lese­zugriffs­priorität 33
 - Workloads
 - Auswertung 48
 - mehrere Verbindungsattributwerte, Zuordnung 167
 - Übersicht 43
 - Zuordnung 158
 - Zugriffsrecht USAGE für Workload erteilen 57
 - Zugriffsrecht USAGE für Workload widerrufen 59
- Datenbankobjekte
 - Rollen 319
 - Workload-Management
 - Eigentumsrecht 303
- Datenbankzugriff für die Workload verhindern 56
- Datenbankzugriff für Workload erlauben 55
- Datenerfassung und -zusammenfassung mit Tabellenfunktionen 181
- DB2-Informationszentrale
 - Aktualisierung 330
 - in verschiedenen Sprachen anzeigen 329
 - Versionen 329
- DDL
 - Workload-Management-Aspekte 303
- Designadvisor
 - Aktivitätsinformationen importieren 145
- Dokumentation
 - Nutzungsbedingungen 333
 - PDF oder Hardcopy 326

E

- Eigentumsrecht von Workload-Management-Objekten 303
- Ereignismonitor
 - Aktivitätsdaten erfassen 143

- Ereignismonitor (*Forts.*)
 - DB2ACTIVITIES 127
 - DB2STATISTICS 127
 - DB2THRESHOLDVIOLATIONS 127
 - Schwellenwertverstöße überwachen 142
 - Statistikdaten für Workload-Management erfassen 137
 - Typen 18, 127
 - WLMEVMON.DDL 127
 - Workload-Management, Einsatzmöglichkeiten 127
- Erkannte Aktivitäten 20
- Erstellen
 - Schwellenwert 76
 - Serviceklassen 36
 - Workload 52
- Erteilen
 - Zugriffsrecht USAGE für Workload 57
- ESTIMATEDSQLCOST, Aktivitätsschwellenwert 64

F

- Fehlerbehebung
 - Lernprogramme 333
 - Onlineinformationen 333
- Fehlerbestimmung
 - Lernprogramme 333
 - Onlineinformationen 333
- Funktionen
 - Tabellenfunktionen
 - WLM_GET_ACTIVITY_DETAILS 209
 - WLM_GET_QUEUE_STATS 216
 - WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES 227
 - WLM_GET_SERVICE_CLASS_AGENTS 221
 - WLM_GET_SERVICE_SUBCLASS_STATS 231
 - WLM_GET_SERVICE_SUPERCLASS_STATS 236
 - WLM_GET_WORK_ACTION_SET_STATS 238
 - WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES 240
 - WLM_GET_WORKLOAD_STATS 244

G

- Gedruckte Bücher
 - bestellen 328
- Gesicherte Kontexte 321
- Gesicherte Verbindungen 321
- Gespeicherte Prozeduren
 - als eine Aktivität verarbeitet 315
 - WLM_CANCEL_ACTIVITY 126
 - WLM_CAPTURE_ACTIVITY_IN_PROGRESS 126
 - WLM_COLLECT_STATS 126

H

- Hilfe
 - anzeigen 329
 - für SQL-Anweisungen 329
- Histogramme
 - Monitorelemente
 - histogram_type 264
 - number_in_bin 266
 - top 279
 - Schablone löschen 149
 - Zweck 133, 182
- Histogrammschablonen
 - ändern 148

Histogrammschablonen (*Forts.*)
 Änderungen bei der Zurücksetzung von Statistik-
 daten 141
 erstellen 148
Höchstwerte
 Monitorelemente
 concurrent_act_top 254
 concurrent_connection_top 254
 concurrent_wlo_act_top 255
 concurrent_wlo_top 255
 coord_act_lifetime_top 256
 cost_estimate_top 258
 rows_returned_top 270
 temp_tablespace_top 274

I

Inaktivieren
 Arbeitsaktion 112
 Arbeitsaktionsset 106
Inaktivieren einer Workload 57
Informationszentrale
 Aktualisierung 330
 in verschiedenen Sprachen anzeigen 329
 Versionen 329
Integration von DB2- und Betriebssystem-Workload-Management 305

K

Katalogsichten
 HISTOGRAMTEMPLATEBINS 289
 HISTOGRAMTEMPLATES 289
 HISTOGRAMTEMPLATEUSE 289
 SERVICECLASSES 290
 THRESHOLDS 292
 WORKACTIONS 293
 WORKACTIONSETS 296
 WORKCLASSES 297
 WORKCLASSETS 298
 WORKLOADAUTH 298
 WORKLOADCONNATTR 299
 WORKLOADS 299

Kennung

 Monitorelemente
 arm_correlator 253
 bin_id 253
 db_work_action_set_id 263
 db_work_class_id 263
 parent_activity_id 266
 parent_uow_id 266
 sc_work_action_set_id 270
 sc_work_class_id 271
 service_class_id 272
 work_action_set_id 280
 work_class_id 281

Konfigurationsparameter

 wlm_collect_int, Konfigurationsparameter 287

L

Leistung

 Data-Warehouse mit Kapazitätsplanung optimieren 193
 Data-Warehouse ohne Kapazitätsplanung optimieren 195

Leistungsmodellierung

 Workload-Management 147

Lernprogramme

 Fehlerbehebung und Fehlerbestimmung 333
 Visual Explain 332

Löschen

 Arbeitsaktionsset 106
 Arbeitsklassenset 114
 Histogrammschablone 149
 Schwellenwert 82
 Serviceklassen 41
 Workloads 59

M

Monitorelemente

 Abfragen

 queue_assignments_total 267
 queue_size_top 268
 queue_time_total 268

 Aktivierungszeit

 last_wlm_reset 265

 Aktivitäten

 act_total 252
 activity_collected 249
 activity_id 250
 activity_secondary_id 250
 activity_type 251
 coord_act_aborted_total 255
 coord_act_completed_total 256
 coord_act_rejected_total 257

 Ausführung

 act_exec_time 252

 Auslastungen

 wlo_completed_total 280
 workload_id 282
 workload_name 282
 workload_occurrence_id 283

 Bereiche

 unterer Grenzwert 253
 coord_act_est_cost_avg 261
 coord_act_exec_time_avg 260
 coord_act_interarrival_time_avg 262
 coord_act_lifetime_avg 258
 coord_act_queue_time_avg 259

 Histogramme

 histogram_type 264
 number_in_bin 266
 oberer Grenzwert 279

 Höchstwerte

 concurrent_act_top 254
 concurrent_connection_top 254
 concurrent_wlo_act_top 255
 concurrent_wlo_top 255
 coord_act_lifetime_top 256
 cost_estimate_top 258
 rows_returned_top 270
 temp_tablespace_top 274

 Kennung

 arm_correlator 253
 bin_id 253
 db_work_action_set_id 263
 db_work_class_id 263
 parent_activity_id 266
 parent_uow_id 266
 sc_work_action_set_id 270
 sc_work_class_id 271
 service_class_id 272
 work_action_set_id 280

Monitorelemente (Forts.)

Kennung (Forts.)

work_class_id 281

Namen

service_subclass_name 272

service_superclass_name 272

work_action_set_name 280

work_class_name 281

Partitionen

coord_partition_num 257

request_exec_time_avg 261

Schwellenwerte

num_threshold_violations 265

threshold_action 274

threshold_domain 274

threshold_maxvalue 275

threshold_name 276

threshold_predicate 276

threshold_queuesize 276

thresholdid 277

section_env 271

UOW (Units of Work)

uow_id 279

Workload-Management

Übersicht 249

Zeilen

rows_fetched 268

rows_modified 269

rows_returned 269

Zeit

prep_time 267

time_completed 277

time_created 278

time_of_violation 278

time_started 278

Zeitmarken

activate_timestamp 249

statistics_timestamp 273

N

Namen

Monitorelemente

service_subclass_name 272

service_superclass_name 272

work_action_set_name 280

work_class_name 281

Namenskonventionen

allgemein 317

Einschränkungen 317

allgemein 317

Nicht erkannte Aktivitäten 20

P

Partitionen

Monitorelemente

coord_partition_num 257

Position der Workload in der Workloadliste 48

Prozeduren

gespeicherte Prozeduren

WLM_CANCEL_ACTIVITY 205

WLM_CAPTURE_ACTIVITY_IN_PROGRESS 206

WLM_COLLECT_STATS 208

R

Rollen 319

Routinen

WLM_CANCEL_ACTIVITY

Aktivität abbrechen, Beispiel 188

S

Schemata

Klassifizierung der Anweisung CALL 85

Schwellenwerte

ACTIVITYTOTALTIME 67

Aktion 61

Aktivitätsschwellenwerte 63

ändern 82

Anwendung auf eine gespeicherte Prozedur 315

Auswertungsreihenfolge 74

Beispiel 174

Bereichsauflösung von Aktivitätsschwellenwerten 68

CONCURRENTDBCOORDACTIVITIES 73

CONCURRENTWORKLOADACTIVITIES 71

CONCURRENTWORKLOADOCCURRENCES 70

CONCURRENTWORKLOADOCCURRENCES, TOTALD-

BPARTITIONCONNECTIONS und TOTALSCPARTITI-

ONCONNECTIONS zusammen verwendet 170

CONNECTIONIDLETIME 64

Domäne 61

erstellen 76

ESTIMATEDSQLCOST 64

in Arbeitsaktionen unterstützte 97

löschen 82

Monitorelemente

num_threshold_violations 265

threshold_action 274

threshold_domain 274

threshold_maxvalue 275

threshold_name 276

threshold_predicate 276

threshold_queuesize 276

thresholdid 277

prädiktiv 61

reaktiv 61

SQLROWSRETURNED 66

SQLTEMPSPACE 65

TOTALDBPARTITIONCONNECTIONS 68

TOTALSCPARTITIONCONNECTIONS 69

Umsetzungsbereich 61

unterstützte Arbeitsklassifikationen 97

Verstöße überwachen 142

Verwendungsbeispiel 169

Warteschlange 61

Zusammenfassung 63

Zweck 61

Schwellenwertverstöße, Ereignismonitor 18

Scripts

WLMEMON.DDL 127

Serviceklassen

Agentenpriorität 33

Aktivitäten zuordnen 29

Aktivitätenstatus 34

ändern 38

Änderungen bei der Zurücksetzung von Statistik-

daten 141

Beispiel 153

erstellen 36

Leistungsminderung analysieren, Beispiel 184

- Serviceklassen (*Forts.*)
 - löschen 41
 - nicht verfolgte Entitäten 35
 - Servicesuperklassen
 - Standard 27
 - Serviceunterklasse
 - Standard 27
 - Superklassen- und Unterklassenhierarchie 25
 - Übersicht 25
 - Verbindungsstatus 34
 - Vorabesezugriffspriorität 33
 - Zeitpunktstatistikdaten ermitteln 180
- Servicesuperklassen
 - ändern 38
 - erstellen 36
 - löschen 41
 - Überwachungsdaten, Überblick 121
- Serviceunterklasse
 - ändern 38
 - erstellen 36
 - löschen 41
 - Überwachungsdaten, Überblick 121
- SET WORKLOAD, Befehl 51, 285
- Sicherheit
 - mit gesicherten Kontexten 321
- SQL-Anweisungen
 - Hilfe anzeigen 329
 - zu Arbeitstypen zuordnen 91
- sqleseti-API
 - Workloadzuordnung 158
- SQLROWSRETURNED, Aktivitätsschwellenwert 66
- SQLTEMPSPACE, Aktivitätsschwellenwert 65
- Standardbenutzerservicesuperklasse (SYSDEFAULTUSERCLASS) 27
- Standardsystemservicesuperklasse (SYSDEFAULTSYSTEMCLASS) 27
- Standardverwaltungsservicesuperklasse (SYSDEFAULTMAINTENANCECLASS) 27
- Standardworkloads 49
- Statistikdaten
 - für Workload-Management erfassen 137
 - mit WLM_COLLECT_STATS zurücksetzen 126
 - Workload-Management-Objekte 128
- Statistikdaten, Ereignismonitor 18
- SYSDEFAULTMAINTENANCECLASS (Standardverwaltungsservicesuperklasse) 27
- SYSDEFAULTSYSTEMCLASS (Standardsystemservicesuperklasse) 27
- SYSDEFAULTUSERCLASS (Standardbenutzerservicesuperklasse) 27

T

- Tabellenbereiche
 - SQLTEMPSPACE, Schwellenwert 65
- Tabellenfunktionen
 - auf unterschiedlichen Ebenen überwachen 177
 - Daten zusammenfassen 181
 - ergänzt mit Überwachungsprogramm für Momentaufnahme 125
 - Nutzungsbeispiel 17
 - WLM_COLLECT_STATS 141
 - WLM_GET_ACTIVITY_DETAILS 124
 - WLM_GET_QUEUE_STATS 140, 216
 - WLM_GET_SERVICE_CLASS_AGENTS 124, 192
 - WLM_GET_SERVICE_CLASS_STATS
 - Leistungsminderung analysieren, Beispiel 184

- Tabellenfunktionen (*Forts.*)
 - WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES 124
 - blockierte Aktivitäten ermitteln 188
 - WLM_GET_SERVICE_SUBCLASS_STATS 140, 180
 - Leistungsminderung analysieren, Beispiel 186
 - WLM_GET_SERVICE_SUPERCLASS_STATS 140
 - WLM_GET_WORK_ACTION_SET_STATS
 - Aktivitäten analysieren, Beispiel 187
 - Anzahl der Aktivitäten ermitteln, Beispiel 187
 - WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES 124
 - WLM_GET_WORKLOAD_STATS 140
- TOTALDBPARTITIONCONNECTIONS, zusammengefasster Schwellenwert 68
- TOTALSCPARTITIONCONNECTIONS, zusammengefasster Schwellenwert 69
- Typen von Ereignismonitoren 127

U

- Übersicht über die Dokumentation 325
- Überwachung
 - Echtzeitüberwachung 15
 - Entitäten, die nicht von Serviceklassen verfolgt werden 35
 - Langzeitüberwachung 18
 - Übersicht 121
- Überwachungsdaten, Überblick 121
- Überwachungsprogramm für Momentaufnahme
 - zur Ergänzung von Tabellenfunktionen 125
- Umsetzungsbereich für Schwellenwert 61
- UOW (Unit of Work)
 - einer Workload zuordnen 43
 - mit Workload abgleichen
 - Beispiel 164
 - zur Standardworkload zuordnen 49
- UOW (Units of Work)
 - Monitorelemente
 - uow_id 279

V

- Verbindungen
 - Status in Serviceklasse 34
 - temporär 69
 - zu Workloads zuordnen
 - Beispiel 162
 - Zuordnung
 - Workload-Verbindungsattribute mit mehreren Werten 167
 - zur Standardverwaltungsworkload zuordnen 51
 - zur Workload zuordnen 48
- Verbindungen zu Workloads zuordnen 48
- Verbindungsattribute
 - UOW einer Workload zuordnen 43
- Visual Explain
 - Lernprogramm 332
- Vorabesezugriffspriorität
 - Serviceklassen 33

W

- Warteschlangen
 - Vorabesezugriff 33
- Warteschlangenschwellenwerte 61

- Widerrufen
 - Zugriffsrecht USAGE für Workload 59
- WLM 20
- WLM_CANCEL_ACTIVITY, Prozedur 205
- WLM_CAPTURE_ACTIVITY_IN_PROGRESS, Prozedur 206
- wlm_collect_int, Konfigurationsparameter 287
- WLM_COLLECT_STATS 208
- WLM_COLLECT_STATS, Tabellenfunktion 141
- WLM_GET_ACTIVITY_DETAILS, Tabellenfunktion 124, 140, 209
- WLM_GET_QUEUE_STATS, Tabellenfunktion 140, 216
- WLM_GET_SERVICE_CLASS_AGENTS
 - Agentennutzung nach Serviceklasse untersuchen 192
- WLM_GET_SERVICE_CLASS_AGENTS, Tabellenfunktion 124, 221
- WLM_GET_SERVICE_CLASS_STATS
 - Leistungsminderung analysieren, Beispiel 184
- WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES
 - blockierte Aktivitäten ermitteln 188
 - Daten zusammenfassen 181
- WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES, Tabellenfunktion 124, 227
- WLM_GET_SERVICE_SUBCLASS_STATS
 - Daten zusammenfassen 181
 - Leistungsminderung analysieren, Beispiel 186
- WLM_GET_SERVICE_SUBCLASS_STATS, Tabellenfunktion 140, 231
- WLM_GET_SERVICE_SUPERCLASS_STATS, Tabellenfunktion 140, 236
- WLM_GET_WORK_ACTION_SET_STATS
 - Aktivitäten analysieren, Beispiel 187
 - Anzahl der Aktivitäten ermitteln, Beispiel 187
- WLM_GET_WORK_ACTION_SET_STATS, Tabellenfunktion 140, 238
- WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES
 - Daten zusammenfassen 181
- WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES, Tabellenfunktion 124, 240
- WLM_GET_WORKLOAD_STATS, Tabellenfunktion 140, 244
- Workload
 - Zuordnung
 - Beispiele 158
- Workload-Management 20
 - Agent
 - Nutzung nach Serviceklasse untersuchen 192
 - Agentenpriorität für Serviceklassen 33
 - Aktivitäten
 - Daten erfassen zu 143
 - Status in Serviceklasse 34
 - zu Serviceklassen zuordnen 29
 - Aktivitätsschwellenwerte
 - abbrechen 146
 - Analyse, Beispiel 187
 - Arbeitsaktion anwenden auf 99
 - blockierte Aktivitäten ermitteln 188
 - für spätere Analyse erfassen, Beispiel 190
 - Informationen in den Designadvisor importieren 145
 - niedriger geschätzter Aufwand und hohe Laufzeit 200
 - Rogue 146
 - zu Arbeitsklassen zuordnen 98
 - ändern
 - Serviceklassen 38
 - Arbeitsaktion
 - ändern 110
 - auf Aktivität anwenden 99
 - inaktivieren 112
 - löschen 112
- Workload-Management (*Forts.*)
 - Arbeitsaktion (*Forts.*)
 - unterstützte Schwellenwerte 97
 - Arbeitsaktionen
 - Domäne des Arbeitsaktionssets 94
 - erstellen 107
 - Arbeitsaktionsset
 - ändern 105
 - Beispiel 174
 - erstellen 104
 - inaktivieren 106
 - löschen 106
 - zur Bestimmung der Typen der ausgeführten Arbeit verwenden 176
 - Arbeitsaktionssets
 - Übersicht 87
 - Arbeitsklasse
 - ändern 117
 - Auswertungsreihenfolge 93
 - Klassifizierung der Anweisung CALL nach Schema 85
 - löschen 117
 - Übersicht 7
 - Arbeitsklassen
 - ALL, Schlüsselwort, Beispiel 172
 - Arbeitstypzuordnung zu SQL-Anweisungen 91
 - erstellen 114
 - Arbeitsklassenset
 - ändern 113
 - CALL verwalten, Beispiel 171
 - erstellen 113
 - löschen 114
 - Arbeitsklassensets 85
 - auf unterschiedlichen Ebenen überwachen 177
 - Befehle
 - SET WORKLOAD 285
 - DDL-Aspekte 303
 - Echtzeitüberwachung 15
 - Entitäten, die nicht von Serviceklassen verfolgt werden 35
 - Ereignismonitor 127
 - Ermittlungsphase 5
 - erstellen
 - Serviceklassen 36
 - gespeicherte Prozeduren ausführen 315
 - Histogramme
 - Schablone löschen 149
 - Zweck 133, 182
 - Histogrammschablone
 - ändern 148
 - erstellen 148
 - Integration mit Betriebssystem-Workload-Manager 305
 - Konzepte, Überblick 3
 - Langzeitüberwachung 18
 - Leistungsmodellierung 147
 - löschen
 - Schwellenwert 82
 - Serviceklassen 41
 - Workloads 59
 - Monitorelemente
 - Übersicht 249
 - Musteranwendung 20
 - Objekteigentumsrecht 303
 - optimieren
 - mit Kapazitätsplanungsdaten, Beispiel 193
 - ohne Kapazitätsplanungsdaten, Beispiel 195
 - Phasen 3
 - Schwellenwert
 - ändern 82

Workload-Management (*Forts.*)

- Schwellenwert (*Forts.*)
 - Beispiel 174
 - erstellen 76
 - Verstöße überwachen 142
 - Zusammenfassung 63
- Schwellenwerte
 - ACTIVITYTOTALTIME 67
 - Aktivitätsschwellenwerte 63
 - Auswertungsreihenfolge 74
 - Bereichsauflösung von Aktivitätsschwellenwerten 68
 - CONCURRENTDBCOORDACTIVITIES 73
 - CONCURRENTWORKLOADACTIVITIES 71
 - CONCURRENTWORKLOADOCCURRENCES 70
 - CONCURRENTWORKLOADOCCURRENCES, TOTALDBPARTITIONCONNECTIONS und TOTALSCPARTITIONCONNECTIONS zusammen verwendet 170
 - CONNECTIONIDLETIME 64
 - ESTIMATEDSQLCOST 64
 - SQLROWSRETURNED 66
 - SQLTEMPSPACE 65
 - TOTALDBPARTITIONCONNECTIONS 68
 - TOTALSCPARTITIONCONNECTIONS 69
 - Verwendungsbeispiel 169
 - zusammengefasste Schwellenwerte 63
 - Zweck 61
- Serviceklasse
 - Leistungsminderung analysieren, Beispiel 184
 - Zeitpunktstatistikdaten ermitteln 180
- Serviceklassen
 - Beispiel 153
 - Vorablesezugriffspriorität 33
- Serviceklassenhierarchie 25
- Statistikdaten
 - mit Statistikereignismonitor erfassen 137
- Statistikdaten für Objekte 128
- Statistikdaten zurücksetzen 141
- Tabellenfunktion
 - Daten zusammenfassen 181
 - ergänzt mit Überwachungsprogramm für Momentaufnahme 125
 - Nutzungsbeispiel 17
- Tabellenfunktionen 124, 140
- Übersicht 85
- Überwachung
 - Übersicht 121
- Überwachungsdaten, Überblick 121
- Überwachungsphase, Überblick 14
- UOW (Unit of Work)
 - einer Workload zuordnen 164
- Verbindungen
 - Status in Serviceklasse 34
- Verbindungen zu Workloads zuordnen 48
- Verwaltung von Arbeit, Überblick 9
- Workload
 - einer UOW zuordnen 164
 - Leistungsminderung analysieren, Beispiel 186
 - Übersicht 6
- Workloads
 - aktivieren 56
 - ändern 54
 - Datenbankzugriff erlauben 55
 - Datenbankzugriff verhindern 56
 - erstellen 52
 - inaktivieren 57

Workload-Management (*Forts.*)

- Workloads (*Forts.*)
 - Standardverwaltungsworkload, Verbindung zuordnen 51
 - Standardworkloads 49
 - Übersicht 43
 - Verbindungsattribute zur Zuordnung zur Workload 43
 - Zuordnung 158
 - Zuordnung bei mehreren Verbindungsattributwerten 167
 - Zugriffsrecht USAGE für Workload erteilen 57
 - Zugriffsrecht USAGE für Workload widerrufen 59
- Workload-Manager
 - Aktivitäten 20
 - Erkennung von Aktivitäten 20
- Workloads
 - aktivieren 56
 - ändern 54
 - Arbeitsaktionssets 101
 - Datenbankzugriff erlauben 55
 - Datenbankzugriff verhindern 56
 - erstellen 52
 - inaktivieren 57
 - Leistungsminderung analysieren, Beispiel 186
 - löschen 59
 - mit UOW abgleichen
 - Beispiel 164
 - Reihenfolge in Auswertungsliste 48
 - Standarddefinitionen 43
 - Standardworkloads 49
 - Übersicht 6, 43
 - Überwachungsdaten, Überblick 121
 - USAGE, Zugriffsrecht
 - erteilen 57
 - widerrufen 59
 - Verbindungsattribute mit mehreren Werten
 - Zuordnung 167
 - Verbindungsattribute zum Zuordnen zur Workload 43
 - verglichen mit Arbeitsaktionssets 101
 - verglichen mit Workloads 101
 - Zuordnen einer Verbindung zur Standardverwaltungsworkload 51
 - Zuordnen von Verbindungen
 - Beispiel 162
 - Zuordnung 48

Z

Zeilen

- Monitorelemente
 - rows_fetched 268
 - rows_modified 269
 - rows_returned 269
 - rows_returned_top 270
- Zeit
 - Monitorelemente
 - prep_time 267
 - time_completed 277
 - time_created 278
 - time_of_violation 278
 - time_started 278
- Zeitmarken
 - Monitorelemente
 - activate_timestamp 249
 - statistics_timestamp 273
- Zugriffsrechte
 - Rollen 319

Zusammenfassen von Daten 181
Zusammengefasste Schwellenwerte 63



SC12-3929-00

