



IT-Tage 2015

Datenbanken

14.12. - 18.12.2015

Frankfurt am Main

Impressum

Alkmene Verlags- und Mediengesellschaft mbH

Wittelsbacherallee 107
60385 Frankfurt am Main

Telefon: +49 (0) 69 / 94 31 97 34

Fax: +49 (0) 69 / 48 00 27 99

E-Mail: andrea.held@alkmene-verlag.de

Sitz in 60385 Frankfurt am Main, Wittelsbacherallee 107
Eingetragen bei dem Amtsgericht Frankfurt am Main HRB 97162

USt-IdNr.: DE290850181

Geschäftsführerin: Andrea Held

Über uns

Historie

Anfang Mai 2014 startet das Online-Portal **Informatik Aktuell** mit dem ersten Artikel "[Ist deklarativ wirklich instruktiv?](#)" von Dr. Jürgen Lampe. Die Motivation: Unabhängige Berichterstattung und hochwertige Inhalte für IT- und Business-Entscheider, Entwickler und Administratoren. Heute schreiben für die **Informatik Aktuell** 140 namhafte Expertenautoren u. a. aus den Bereichen Softwareentwicklung, Softwarearchitektur, Datenbanken, Netzwerke, Servertechnologien, IT-Sicherheit, Projektmanagement und IT-Recht. Stetig wachsende Nutzerzahlen über die verschiedenen Onlinekanäle zeigen uns, dass wir mit unserem Format richtig liegen, unseren Lesern ein seriöses, nicht werbeüberladenes IT-Portal mit aktuellen, tiefgehenden Inhalten anzubieten. Über 16.000 [Facebook](#)-Fans und ca. 7.000 [Newsletter](#)-Abonnenten erhalten neben neuen Artikeln aktuelle News aus den verschiedenen IT-Bereichen.

Inhalte / Zielgruppe

Informatik Aktuell behandelt ausschließlich Themen aus dem professionellen Unternehmensumfeld: Anwendungsentwicklung, Anwendungsarchitektur, Methoden in der Anwendungsentwicklung, Hochverfügbarkeit, Datenbanken, Big Data oder Themen aus dem Projektmanagementbereich wie Agile Softwareentwicklung. Es verwundert also nicht, dass die Besucher der Informatik Aktuell vorwiegend aus dem professionellen Umfeld kommen. **Informatik Aktuell** ermöglicht ihrer Zielgruppe tief gehende Einblicke in aktuelle Entwicklungstechnologien und Methoden. Verstärkt werden wir in nächster Zeit Artikel zu den Themen DevOps, Big Data und Cloud anbieten.

Einmal jährlich im Dezember veranstaltet **Informatik Aktuell** die **IT-Tage** in Frankfurt am Main. Hochkarätige Experten liefern dabei das aktuelle Know-how zu jeweils einem Themenschwerpunkt. Mehrmals jährlich gibt es kleinere Schwerpunkt-Veranstaltungen unter dem Titel "Wissen kompakt": In ein bis zwei Tagen vermitteln wir in "deep dives" Know-how zu wichtigen Kernthemen und Neuerungen.

Vision

Unser Ziel ist es, **Informatik Aktuell** als unabhängiges Online-Medium mit dem Nutzer-Fokus IT-Experten und Entscheider weiter auf dem Markt zu etablieren. Dabei wollen wir weder die Masse mit Consumer-Themen erreichen noch den Lesefluss durch aggressive Werbeeinblendungen stören. Uns wird ein gewissenhaftes Lektorat immer wichtiger sein als hochfrequentes Artikel- und News-Streaming.

Weiterhin wollen wir mit den **IT-Tagen** alljährlich die große Bandbreite aktueller und zukunftsorientierter Technologien abdecken und IT-Professionals aus führenden Unternehmen aller Branchen entscheidende Impulse geben. 2016 werden die **IT-Tage** an drei Tagen und in zahlreichen parallelen Tracks zu Themen wie DevOps, Software-Entwicklung, Datenbanken, Big Data und NoSQL, Storage und Betriebssysteme rund 300 Teilnehmer/innen auf den aktuellen Stand neuer Technologien bringen.

Jetzt schon einplanen:

IT-Tage 2016

13. - 15.12.2016
Frankfurt am Main

und

Wissen Kompakt - MS SQL Server

Wie muss ich bei der Performance-Analyse eines Microsoft SQL Servers vorgehen? Wie kann ich Schwachstellen und Engpässe identifizieren und diese beheben? Microsoft Certified Master und MS SQL Server-Experte **Uwe Ricken** verrät es Ihnen. Wir bieten Ihnen ein zweitägiges Seminar Anfang März sowie Anfang Juni 2016 in Frankfurt/M. im Hotel Maritim an der Messe an. Jeder Tag ist einzeln buchbar. Die Verpflegung inkl. Pausensnacks, Getränken und reichhaltigem Mittagessen ist im Preis enthalten.

SQL Server: Performance-Analyse und Optimierung

Montag, 07.03.2016 und Mittwoch 01.06.2016

MS SQL Server: Indexanalyse und Performancesteigerung

Dienstag, 08.03.2016 und Donnerstag 02.06.2016

Mehr Informationen: <http://www.informatik-aktuell.de>

Programm / Inhalt

Montag, 14.12.2015

- | | | |
|---------------|-------------------------|-----------------------------------------------------------------------------------------------|
| 09:30 - 10:15 | Ralph Kemperdick | <u>Microsoft Azure als Datenplattform</u> |
| 10:30 - 11:15 | Robert Panther | <u>SQL Server Indizes - Verschiedene Varianten im Überblick</u> |
| 11:30 - 12:15 | Maic Beher | <u>SQL Server 2014: Security und Verschlüsselungsmechanismen</u> |
| 13:30 - 14:15 | Uwe Ricken | <u>SQL Server: Connection, Session, Request - Monitoring ohne teure Tools</u> |
| 14:30 - 15:15 | Uwe Ricken | <u>SQL Server: Richtig Indizieren – Tipps und Tricks aus der Praxis</u> |
| 15:30 - 16:15 | William Durkin | <u>SQL Server 2016: Warum soll ich mir das antun?</u> |

Dienstag, 15.12.2015 (Workshop SQL Server)

- | | | |
|---------------|-------------------|------------------------------------------------------------------------------------------------------|
| 09:30 - 10:15 | Uwe Ricken | <u>Richtige Konfiguration von Dienstkonto und System für schnellere Prozesse</u> |
| 10:30 - 11:15 | Uwe Ricken | <u>Korrekte Systemparameter für einen guten Start</u> |
| 11:30 - 12:15 | Uwe Ricken | <u>TEMPDB - Tücken und Fallstricke bei der Konfiguration</u> |
| 13:30 - 14:15 | Uwe Ricken | <u>Konfiguration von Datenbanken für unterschiedliche Workloads wie OLTP und DWH</u> |
| 14:30 - 15:15 | Uwe Ricken | <u>Buffer Pool und PLE - Optimale Nutzung von RAM</u> |
| 15:30 - 16:15 | Uwe Ricken | <u>Indexe - ein Garant für schnelle Abfragen?</u> |

Mittwoch, 16.12.2015

- | | | |
|---------------|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| 09:00 - 09:45 | Patrick Guillebert | <u>Cassandra NoSQL - Architektur und Anwendungsfälle</u> |
| 09:45 - 10:30 | Mario Briana | <u>Wann machen NoSQL-Datenbanken Sinn?</u> |
| 10:45 - 11:45 | Jörg Kleinz | <u>NoSQL Graphdatenbanken – nicht nur für Big Data</u> |
| 11:45 - 12:30 | Michael Wittig | <u>Datenbanken und Big Data: Datenbank am Limit – Spaltenorientierung als Ausweg</u> |
| 13:30 - 14:15 | Stephan Kaps | <u>Flyway vs. LiquiBase – Battle der Datenbank-migrationstools</u> |
| 14:30 - 15:15 | Wolfgang Epting | <u>Testdaten – versteckte Geschäftschance oder immanentes Sicherheitsrisiko?</u> |
| 15:30 - 16:15 | Stefan Hummel | <u>DB2 In-Memory - Eine Technologie nicht nur für typisches OLAP</u> |
| 16:30 - 17:15 | Philipp Krenn | <u>Cloud Computing und Big Data, Platform as a Service: Lasst uns die Datenbank in der (private oder public) Cloud hosten</u> |

Donnerstag, 17.12.2015

- | | | |
|---------------|--------------------------|--------------------------------------------------------------------------------------------------|
| 09:30 - 10:15 | Martin Klier | <u>Oracle 12c: In-Memory Column Store</u> |
| 10:30 - 11:15 | Marek Adar | <u>Oracle 12c Multitenant: Architektur und Anwendung</u> |
| 11:30 - 12:15 | Marek Adar | <u>Oracle 12c: Oracle Recovery Manager New Features</u> |
| 13:30 - 14:15 | Udo Fohrmann | <u>Adaptive Dynamic Sampling in Oracle 12c</u> |
| 14:30 - 15:15 | Jérôme Witt | <u>Oracle RDBMS – Grid Infrastructure 12c: Failover Cluster - Best Practices</u> |
| 15:30 - 16:15 | Dierk Lenz | <u>Oracle 12c Upgrade - Tipps und Tricks aus der Praxis</u> |
| 16:30 - 17:15 | Angelika Gallwitz | <u>Statistische Auswertungen in Oracle mit Statspack und AWR</u> |

Freitag, 18.12.2015

- | | | |
|---------------|---------------------------|---------------------------------------------------------------------------------------------------------|
| 09:30 - 10:15 | Norbert Rieger | <u>Performance-Optimierung von Oracle Datenbanken "nur" mit der Standard Edition</u> |
| 10:30 - 11:15 | Markus Winand | <u>Den Suchraum des Optimizers gestalten</u> |
| 11:30 - 12:15 | Jonas Gassenmeyer: | <u>Materialized Views in Oracle</u> |
| 13:30 - 14:15 | Randolf Geist | <u>Oracle Parallel Execution – Analyse und Troubleshooting</u> |
| 14:30 - 15:15 | Thorsten Bruhns | <u>Monitoring von Oracle-Datenbanken mit check_mk</u> |
| 15:30 - 16:15 | Andreas Jordan | <u>Flashback – Warum Oracle Zeitreisen anbieten kann, andere Datenbanken aber nicht</u> |

Montag, 14. Dezember 2015, 09:30 - 10:15 Uhr

Ralph Kemperdick:

Microsoft Azure als Datenplattform

Übersicht

In dieser Session möchten wir eine Orientierung geben, welche Daten-Services auf Azure die geeignete Plattform für eine App bzw. eine Anwendung sein können.

Die Session konzentriert sich auf die Platform as a Service (PaaS) mit einem SQL Interface. Es wird Azure SQL Server, Azure SQL DW, DocumentDB, Stream Analytics, Spark/Scala/Hive und Data Lake Analytics betrachtet und Unterschiede herausgearbeitet. Live Demos begleiten die einzelnen Themen in der Session. Ferner werden Argumente für und gegen Cloud basierte Services diskutiert.

Der Sprecher: Ralph Kemperdick



Ralph Kemperdick ist seit vielen Jahren bei Microsoft Deutschland für die Systemberatung im Bereich Datenbanken tätig. Er arbeitet gemeinsam mit dem weltweit agierenden SQL Server Customer Advisory Team an den größten Data Warehouse Projekten in Deutschland. Vor seiner Zeit bei Microsoft hat er Erfahrung in den Bereichen Softwareentwicklung und Marketing gesammelt sowie Softwareentwicklungsprojekte in Telekommunikation, Handel und Industrie geleitet. Seit 1984 beschäftigte er sich schwerpunktmäßig mit Datenbanken und Architekturen. Heute liegt sein Schwerpunkt in den Bereichen Datawarehouse-Infrastrukturen, Analytische Applikationen und Data Mining-Verfahren. Er versteht sich als Sprachrohr der Microsoft-Kunden zu der Datenbank-Gruppe die derzeit mit der Weiterentwicklung des SQL Servers beschäftigt ist.

Kontakt: Ralph.Kemperdick@microsoft.com

Montag, 14. Dezember 2015, 10:30 - 11:15 Uhr

Robert Panther:

SQL Server Indizes - Verschiedene Varianten im Überblick

Übersicht

Indizes sind ein entscheidender Faktor für die Performance von Datenbanken. Mittlerweile bietet SQL Server eine ganze Reihe von verschiedenen Indexformen: Angefangen von den klassischen gruppierten und nicht-gruppierten Indizes über Volltextindizes, XML-Indizes, räumliche Indizes bis hin zu den mit SQL 2012 eingeführten Columnstore Indizes zu denen mit SQL 2014 noch die gruppierte Variante dazu kam.

Dieser Vortrag vermittelt einen Überblick über die verschiedenen Indexformen und zeigt Kriterien auf, mit denen entschieden werden kann, welche Variante für welchen Einsatzzweck am besten geeignet ist. Dazu wird auch beleuchtet, was bei den jeweiligen Indextypen zu beachten ist, damit diese effizient genutzt werden können.

Der Sprecher: Robert Panther



Der Diplom-Informatiker (FH) Robert Panther ist hauptberuflich als Executive Consultant für CGI in Deutschland tätig. Daneben schreibt er Fachartikel und Bücher zu verschiedenen Themen rund um sein Spezialgebiet SQL Server. Im Mai 2012 veröffentlichte er bei Microsoft Press in der Reihe "Richtig einsteigen" eines der ersten deutschsprachigen Bücher zu SQL Server 2012. Bei entwickler.press erschien Anfang 2010 sein SQL Server Performance-Ratgeber, 2014 kam ein kompaktes Buch zur SQL-Abfrageoptimierung dazu. Dazu ist Robert Panther regelmäßig als Sprecher bei IT-Konferenzen (u. a. BASTA!, SQLCON, Frankfurter Datenbanktage) zu sehen und auch in der Professional Association for SQL Server (PASS) aktiv. Mit Microsoft SQL Server beschäftigt er sich bereits seit der Version 6.0 intensiv und hat hier bereits als Entwickler, Administrator und Datenbank-Architekt zahlreiche Projekte erfolgreich mitgestaltet. Seine Themenschwerpunkte sind insbesondere Performance Tuning, die SQL Server Integration Services sowie SQL Server Express.

Kontakt: robert.panther@cgi.com

Montag, 14. Dezember 2015, 11:30 - 12:15 Uhr

Maic Beher:

SQL Server 2014: Security und Verschlüsselungsmechanismen

Security

Security- Arbeiten mit Dienstkonten

Verwaltete Dienstkonten (MSA)

- Wird in der AD verwaltet
- Kann den SPN setzen
- Tauscht regelmäßig Kennwörter
- Nicht für FCI geeignet

Gruppenverwaltete Dienstkonten (gMSA)

- Wird offiziell noch nicht von SQL Server unterstützt

Security – Windows Groups Login

- Neues ab SQL Server 2012
- Standard Schema für Windows Gruppen
- es gilt die niedrigste ID
- Standardsprache auf Datenbankebene
- nicht wirksam, wenn dem Login eine Standardsprache zugewiesen wurde

Security – ‚Best Practice‘ in Contained Databases

- Windowsbasierte User benutzen
- Benutzer mit Passwörtern können kein Kerberos
- Alter any User vermeiden
- Zugriff nur auf die DB begrenzt
- Identische User erstellen
- Trustworthy Eigenschaft aktivieren
- Keine identischen ‚Benutzer mit Passwort‘ und ‚Anmeldenamen‘
- wenn vorhanden, dann an Instanz anmelden

Security – Anmeldeinformationen (Credentials)

- Encryption Provider für Anmeldeinformationen
- speichert, verwaltet Verschlüsselungsschlüssel außerhalb des SQL Servers
- bietet Multi-Faktor Authentifizierung (1,2,3)
- Separation-of-Duties
- Securityverwaltung! = Datenverwaltung

- Enterprise Feature
- Kann externe Schlüsselverwalter benutzen
- SmartCard
- USB Geräte
- EKM Module

Verschlüsselung

Verschlüsselung - Begrifflichkeit

- Symmetrische Schlüssel
 - Ent- und Verschlüsseln mit dem selben Schlüssel
- Asymmetrische Schlüssel
 - Ent- und Verschlüsseln mit einem Schlüsselpaar
 - Privater Schlüssel
 - Öffentlicher Schlüssel
- Hashalgorithmus
 - Erzeugt eine Prüfsumme mit festgelegter Länge

Verschlüsselung - Eigenschaften

Symmetrischer Schlüssel

- Schnell
- Sicher
- Größenneutral

Asymmetrischer Schlüssel

- einfache Schlüsselübergabe

Zertifikat

- PKI bestätigter öffentlicher Schlüssel eines asymmetrischen Schlüsselpaars

Verschlüsselung – Datenbankbenutzer mit Zertifikaten

Voraussetzungen:

- Datenbank Master Key muss erstellt sein
- Zertifikat muss in zwei Dateien vorliegen
- öffentlicher Schlüssel im CER/DER Format
- privater Schlüssel im PVK Format

Tool zum Erstellen der Dateien:

- PVKConverter.EXE

Verschlüsselung – Asymmetrischer Schlüssel

- Zweck asymmetrischer Schlüssel
- Schützt den symmetrischen Verschlüsselungsschlüssel
- Kann (eigenschränkt) Daten verschlüsseln
- Signiert Datenbankobjekte
- Können importiert werden

Aber:

- Können NICHT Verbindungen verschlüsseln
- Können nicht exportiert werden

Verschlüsselung – Asymmetrischer Schlüssel (Benutzer)

Benutzer auf Basis eines asymmetrischen Schlüssels

- Können sich nicht anmelden
- Können Berechtigungen erhalten
- Können Module signieren

Verschlüsselung – Signierte Prozeduren

- Mehrmals signierfähig
- Ändern einer Prozedur löscht die Signierung

SQL 2016 Angekündigte Features

Row Level Security

gewährt Zugriff basierend auf

- Gruppenmitgliedschaft
- Ausführungskontext
- portiert Datensicherheit aus Applikationsebene auf Datenbankebene
- basiert auf ‚Sicherheitsrichtlinien‘ (Security Policy)
- benötigt Inline-Tabellenwertfunktionen als Prädikat
- Funktion muss mit Schemabindung erstellt werden
- pro DML nur ein Prädikat

Always Encrypted

- Spaltenbasierte Verschlüsselung
- trennt Datenbesitz und Datenverwaltung
- transparent für die Applikation
- benötigt:
 - einen speziellen Treiber
 - Column Master Key und Column Encryption Key

Der Sprecher: Maic Beher



Maic Beher ist als freiberuflicher Consultant in Industrie, Mittelstand und internationalem Finanzumfeld seit über 15 Jahren unterwegs. Seine Schwerpunkte befinden sich dabei im Aufbau angepasster Betriebskonzepte mit Hochverfügbarkeitskomponenten, beim Design der Datenbanksicherheit im Verbund mit den weiteren Fachbereichen und bei den Lösung von Performanceproblemen. Als Microsoft zertifizierter Trainer schult er regelmäßig die SQL MOC Kurse, hält aber auch auf Kunden zugeschnittene Kurse an. Des Weiteren spricht er regelmäßig im Rahmen der SQL PASS-Veranstaltungen.

Kontakt: maic.beher@trivadis.com

Montag, 14. Dezember 2015, 13:30 - 14:15 Uhr

Uwe Ricken:

SQL Server: Connection, Session, Request - Monitoring ohne teure Tools

Agenda

- Vom Client zur Datenbank
- Verbindungen -> Sitzungen -> Anforderungen
- Wo und wann wartet Microsoft SQL Server
- Welche Aufgaben werden ausgeführt?
- Wartezustände und Warteschlangen

Verbindungen zu SQL Server

- Eine Connection beschreibt den physikalischen Verbindungsaufbau von einem Client zu Microsoft SQL Server.
- Über eine authentifizierte Verbindung kann eine Sitzung aufgebaut werden, mit der auf Objekte des Microsoft SQL Server zugegriffen werden kann.

Hierarchie der dmo's

Management View	Information
sys.dm_exec_connections	physikalischer Aufbau (Client/ Server)
sys.dm_exec_sessions	Aktive Server- und Benutzersitzungen innerhalb einer physikalischen Verbindung
sys.dm_exec_requests	Aktuell ausgeführte Anforderungen innerhalb einer Benutzersitzung

Von der Verbindung zur Anforderung

- sys.dm_exec_connections
 - Gibt Informationen über die zu dieser Instanz von SQL Server hergestellten Verbindungen zurück.
- sys.dm_exec_sessions
 - Gibt eine Zeile pro authentifizierter Sitzung in SQL Server zurück.
- sys.dm_exec_requests
 - Gibt Informationen über jede einzelne Anforderung, die in SQL Server ausgeführt wird, zurück.

Wenn Microsoft SQL Server wartet

- sys.dm_os_wait_stats
 - Gibt Informationen zu allen Wartevorgängen in den Threads zurück, die ausgeführt wurden.
- sys.dm_os_waiting_tasks

- Gibt Informationen zur Warteschlange von Tasks zurück, die auf eine Ressource warten.
- `sys.dm_os_tasks`
 - Gibt eine Zeile für jeden aktiven Task in der SQL Server-Instanz zurück.

Wenn Microsoft SQL Server wartet

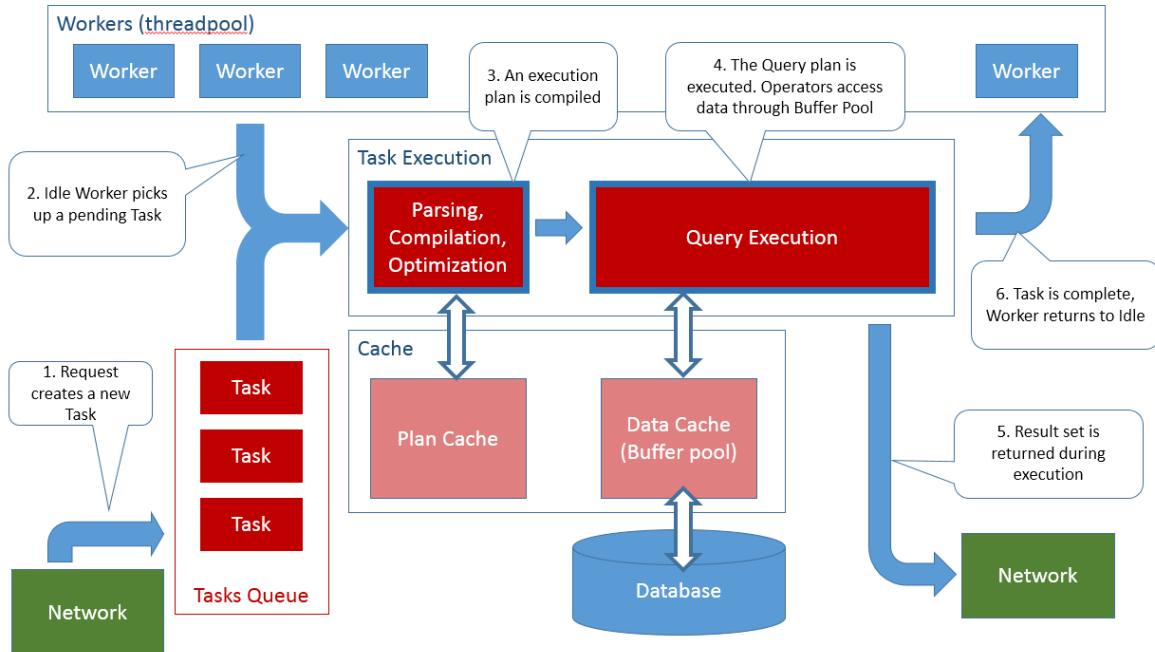
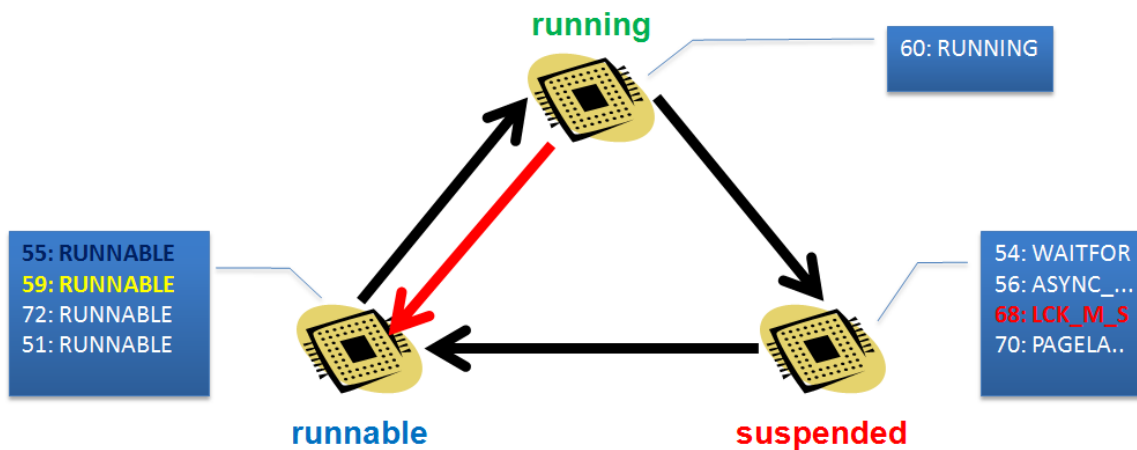
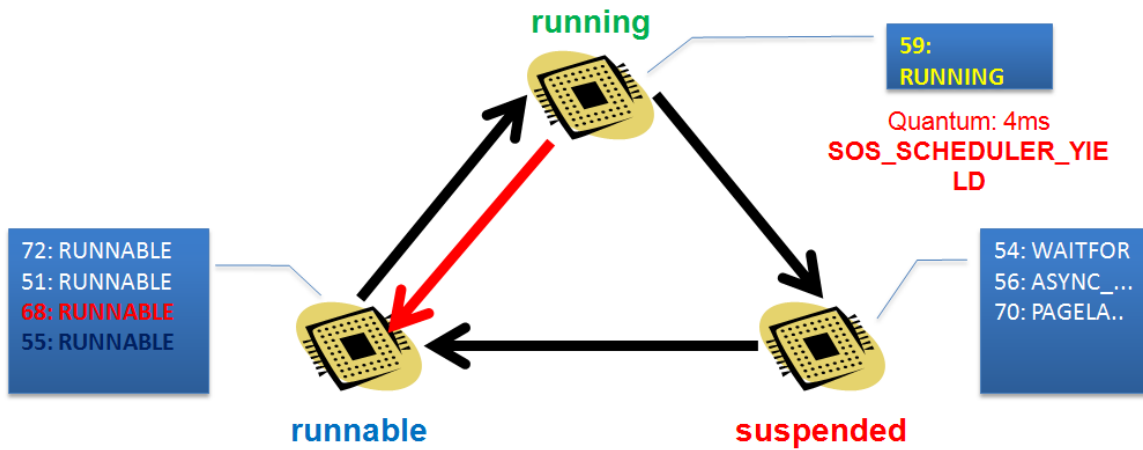
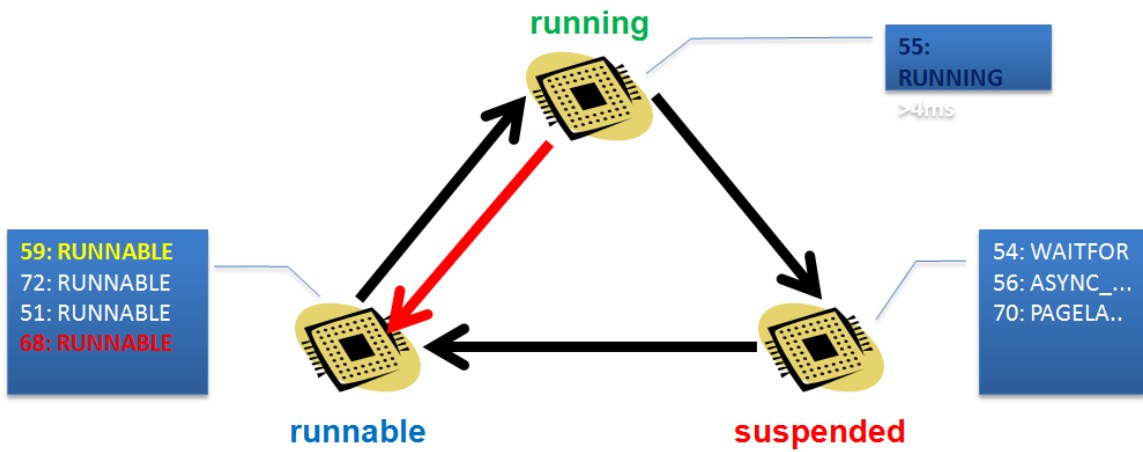


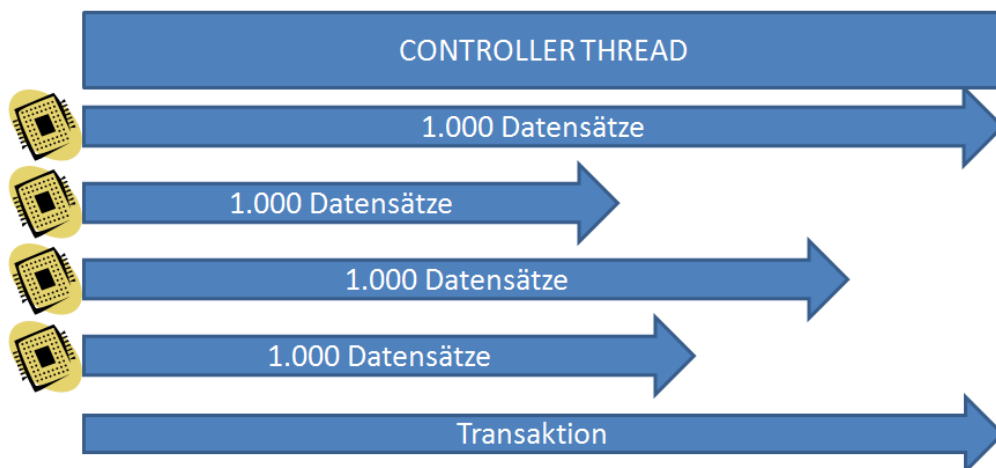
Abbildung: <http://rusanu.com/2013/08/01/understanding-how-sql-server-executes-a-query>

Wenn Microsoft SQL Server wartet

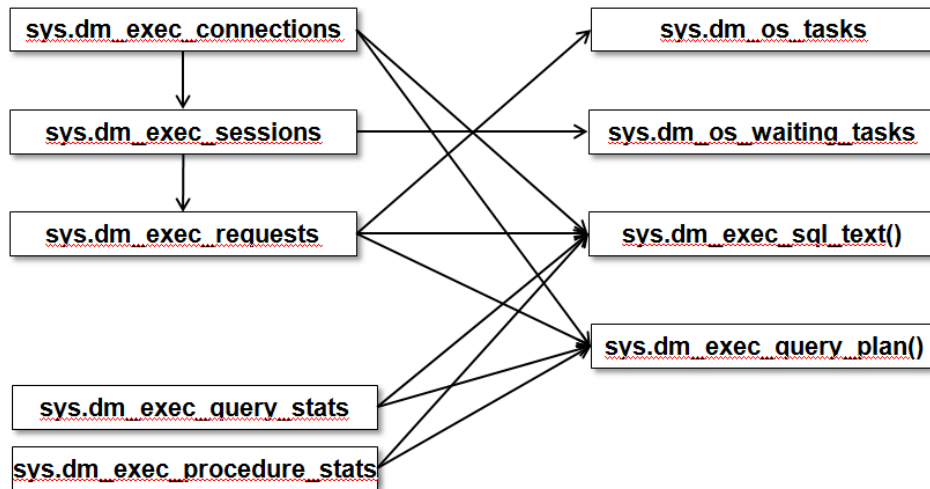




CXPACKET



Zusammenfassung



Der Sprecher: Uwe Ricken



Uwe Ricken ist seit 1992 im IT-Umfeld tätig. Den Weg zum Microsoft SQL Server hat er im Jahre 1999 gefunden, als er den Auftrag für die Erstellung einer Mitgliederverwaltung für die Amerikanische Handelskammer in Deutschland erhielt. Dieses System wurde anschließend in fünf weiteren europäischen Ländern implementiert. Die ursprüngliche Leidenschaft für die Entwicklung von Datenbankmodellen mit Microsoft SQL Server wurde dann ab dem Jahre 2007 zusätzlich um die Funktionen und Aufgaben als DBA bei der Deutschen Bank in Frankfurt am Main erweitert. Mehr als 6 Jahre Praxiserfahrung als DBA und mehr als 14 Jahre Erfahrung als Entwickler von Datenbankmodellen wurden im Mai 2013 mit der Zertifizierung zum 7. deutschen „Microsoft Certified Master – SQL Server 2008“ gekrönt. Um das Jahr 2013 erfolgreich abzuschließen, wurde im Juli 2013 zum ersten Mal der MVP-Award von Microsoft für seine Unterstützung der Microsoft SQL Server Community verliehen. Damit war Uwe Ricken der erste MCM + MVP in Deutschland.

Kontakt: SQLTraining@db-berater.de

Montag, 14. Dezember 2015, 14:30 - 15:15 Uhr

Uwe Ricken:

SQL Server: Richtig Indizieren – Tipps und Tricks aus der Praxis

Information

- Alle gezeigten Beispiele basieren auf echten Vorfällen und/oder Fragen aus der SQL Server Community
- Jede der in dieser Präsentation gezeigten Demos ist in der SSMS Solution numerisch sortiert und kann hier vollständig heruntergeladen werden:
<http://db-berater.blogspot.de>
- Sofern zu den geschilderten Sachverhalten Blogbeiträge vorhanden sind, werden sie unter der Präsentationsfolie angegeben.
(Alle Blogbeiträge sind ausschließlich in deutsch verfasst!)

Agenda

- Probleme bei partitionierten Views 200
- ISNULL und SARGable Abfragen 200
- Bedeutung von Datentypen für Prädikate 300
- Einfluss von Fremdschlüssel-Einschränkungen 200
- Hierarchie von Datentypen 300
- Seiteneffekte bei unterschiedlicher COLLATION 300
- Definition kombinierter Indexe 400
- Sort Spills 400

Partitionierte Views

Szenario

- Ein Unternehmen teilt seine Auftrags-tabelle in zwei separate Tabellen auf.
 - [dbo].[ActualOrders]
 - [dbo].[HistoryOrders]
- Das Entwicklerteam hat – basierend auf den getrennten Daten – eine partitionierte View erstellt, die vom Business für Abfragen verwendet werden konnte.

Lösung

- Verwendung von CHECK Constraints auf den Attributen, die für die Trennung der Daten steuernd sind (z. B. OrderDate).
- Wenn Daten in den historischen Bereich verschoben werden sollen, müssen die CHECK Constraints gelöscht werden. Anschließend können die Daten verschoben werden.
- Nachdem die Daten verschoben wurden, müssen die CHECK Constraints erneut mit angepassten Grenzwerten implementiert werden.

<http://db-berater.blogspot.de/2013/07/optimierung-von-partitioned-views.html>

ISNULL und SARGable Abfragen

Szenario

- Der Entwickler hat in einem Produktivsystem eine Testtabelle erstellt, die auf identischen Metadaten einer Produktionstabelle basiert. Informationen darüber, ob ein Attribut NULLABLE oder NOT NULLABLE ist, wurde nicht überprüft. Die Tabelle wurde auf Basis von Standards erstellt!
- Eine Abfrage des Entwicklers gegen die Testtabelle läuft sehr performant; hingegen wird die Abfrage gegen die Produktionstabelle sehr langsam ausgeführt.
- Was passiert hier?

Lösung

- Die Funktion ISNULL sollte nicht für NULLable Attribute verwendet werden. Besser ist eine explizite Prüfung auf NULL.
e.g. `WHERE CCode IS NULL OR CCode = 'AA'`
- Wenn ein Attribut nicht NULL sein darf, sollte das immer durch eine Einschränkung vordefiniert werden.

<http://db-berater.blogspot.de/2014/08/isnull-als-praedikat-seek-oder-scan.html>

Korrekte Datentypen für Prädikate

Szenario

- Eine neue Applikation wurde in Betrieb genommen und man stellte fest, dass Abfragen auf die Kundennummern sehr langsam liefen.
- Das Attribut [Customer_Id] ist indiziert aber der Ausführungsplan zeigt immer einen INDEX SCAN anstatt eines – performanten – INDEX SEEK.
- Was passiert hier?

Lösung

- Stellen Sie sicher, dass immer die korrekten Datentypen für Prädikatswerte verwendet werden.
- Wenn Sie sich nicht sicher sind, überprüfen Sie den Ausführungsplan
 - Sind "residual" Prädikate vorhanden
 - Aktivieren Sie TF 9130 um teure Filteroperatoren sichtbar zu machen

<http://db-berater.blogspot.de/2014/05/warum-korrekte-datentypen-fur-where.html>

Einfluss von Fremdschlüssel auf Performance

Szenario

- Der Entwickler hat zwei neue Tabellen in das Datenmodell implementiert. Diese Tabellen haben zwar eine logische Beziehung zueinander; wurden aber nicht explizit durch Fremdschlüssel definiert.

- Diese Tabellen
 - [dbo].[Customers]
 - [dbo].[Invoices]sind Bestandteil komplexer Abfragen und laufen teilweise schlecht
- Wie kann man Abfragen über diese Tabellen optimieren?
Was passiert hier?

Lösung

- Die Benutzung von Fremdschlüssel veranlasst den Query Optimizer, die Abfragen zu „simplifizieren“
- Ein Fremdschlüssel verhindert die Prüfung auf vorhandene Werte in der Referenztabelle, wenn keine Ausgabe von Attributen der Referenztabelle erfolgen soll!
- Eine Fremdschlüssel-Einschränkung stellt sicher, dass in der Detailtabelle ausschließlich Werte vorhanden sind, die sich in der Haupttabelle befinden.

<http://db-berater.blogspot.de/2013/12/wie-fremdschlüssel-einschränkungen.html>

Hierarchie von Datentypen

Szenario

- Der Entwickler liefert eine neue Stored Procedure für die Befüllung einer Listview aus. Diese Prozedur besitzt einen Übergabeparameter.
- Die Stored Procedure verhält sich im Laufzeitverhalten sehr langsam!
- Was ist der Fehler?
Wie kann man die Prozedur optimieren?
Was geht hier vor?

Lösung

- Stellen Sie immer sicher, dass auch wirklich die korrekten Datentypen als Prädikate verwendet werden.
- Untersuchen Sie den Ausführungsplan auf CONVERT_IMPLICIT-Befehle. Sie sind in der Regel ein Indiz für falsche Datentypen.

<http://db-berater.blogspot.de/2014/11/rangfolge-von-datentypen-auswirkung-auf.html>

Seiteneffekt von falscher Sortierung

Szenario

- Eine Datenbankapplikation benutzt TEMPDB sehr stark. In vielen Fällen verwenden Stored Procedures TEMPDB, um Daten in einer temporären Tabelle zwischen zu speichern, um sie anschließend in JOINS zu verwenden.
- Das Laufzeitverhalten in der Produktionsumgebung ist miserabel, da in allen Fällen INDEX SCANS statt INDEX SEEKS durchgeführt werden.
- Der Entwickler testet die Prozedur in seiner Umgebung und kann keine Fehler feststellen.
- Was passiert hier?

Lösung

- Überprüfung der COLLATION von Microsoft SQL Server!
- Überprüfung der COLLATION von TEMPDB!
(ist generell die gleiche COLLATION wie beim Server)
- Expliziter Hinweis auf COLLATION im JOIN für die Tabelle mit den wenigen Datensätzen.
- Benutzung von @Table Variables wenn es sich nur um kleine Datenmengen handelt!

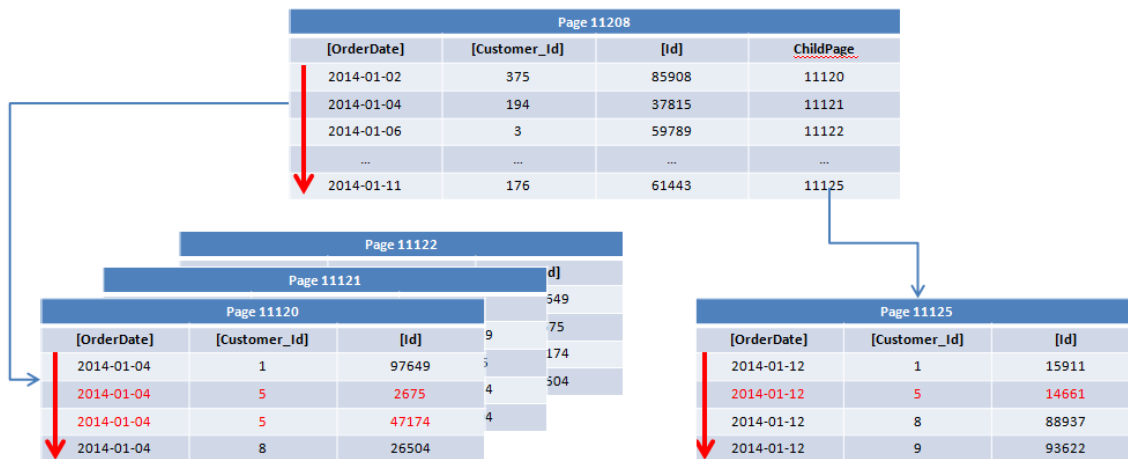
<http://db-berater.blogspot.de/2014/05/sortierungskonflikte-auswirkungen-auf.html>

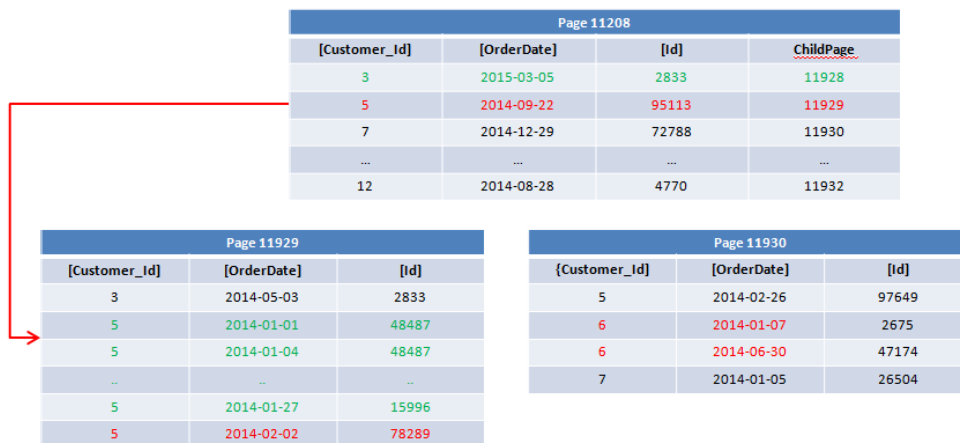
Probleme mit kombinierten Indexen

Szenario

- Der Entwickler bekam den Auftrag, die Performance einer Abfrage zu verbessern, die zwei Attribute der Tabelle abfragt.
- Aus diesem Grund erstellte der Entwickler einen neuen Index, dessen Schlüsselattribute die beiden Attribute repräsentierten, die regelmäßig abgefragt werden.
- Bei der Reihenfolge der Attribute im Index hat der Entwickler die Kardinalität beider Werte berücksichtigt!
- Obwohl der Index für die Abfrage verwendet wurde, war kein Performancevorteil erkennbar.
- Was passiert hier?

<http://db-berater.blogspot.de/2014/12/kombinierte-indexe-richtig-definieren.html>





Page 11208			
[Customer_Id]	[OrderDate]	[Id]	ChildPage
3	2015-03-05	2833	11928
5	2014-09-22	95113	11929
7	2014-12-29	72788	11930
...
12	2014-08-28	4770	11932

Page 11929		
[Customer_Id]	[OrderDate]	[Id]
3	2014-05-03	2833
5	2014-01-01	48487
5	2014-01-04	48487
...
5	2014-01-27	15996
5	2014-02-02	78289

Page 11930		
{Customer_Id}	[OrderDate]	[Id]
5	2014-02-26	97649
6	2014-01-07	2675
6	2014-06-30	47174
7	2014-01-05	26504

Lösung

- Überprüfung der Kardinalität der verwendeten Attribute ist Grundvoraussetzung für einen guten Index!
- Sollten Abfragen geplant sein, die RANGE SCANS verursachen, sollte der Index so gestaltet sein, dass die Attribute für den RANGE SCAN nicht die führenden Attribute repräsentieren!

<http://db-berater.blogspot.de/2014/05/sortierungskonflikte-auswirkungen-auf.html>

Sort Spills

Szenario

- Die Abfrage gegen eine zentrale Tabelle der Applikation läuft bis zu einer bestimmten Anzahl von Datensätzen optimal und schnell.
- Sobald nur eine einzige Zeile mehr im Abfrageergebnis vorhanden ist, ändert sich das Laufzeitverhalten der Abfrage dramatisch; sie wird bis zu vier mal langsamer!
- Ein Index konnte nicht auf das Attribut gelegt werden, da nicht im Vorfeld klar ist, welches Attribut tatsächlich vom Anwender für das Abfragekriterium verwendet wird - NAVISION
- Was passiert hier?

<http://db-berater.blogspot.de/2013/12/sort-in-tempdb-teure-aber-vermeidbare.html>

Der Sprecher: Uwe Ricken



Uwe Ricken ist seit 1992 im IT-Umfeld tätig. Den Weg zum Microsoft SQL Server hat er im Jahre 1999 gefunden, als er den Auftrag für die Erstellung einer Mitgliederverwaltung für die Amerikanische Handelskammer in Deutschland erhielt. Dieses System wurde anschließend in fünf weiteren europäischen Ländern implementiert. Die ursprüngliche Leidenschaft für die Entwicklung von Datenbankmodellen mit Microsoft SQL Server wurde dann ab dem Jahre 2007 zusätzlich um die Funktionen und Aufgaben als DBA bei der Deutschen Bank in Frankfurt am Main erweitert. Mehr als 6 Jahre Praxiserfahrung als DBA und mehr als 14 Jahre Erfahrung als Entwickler von Datenbankmodellen wurden im Mai 2013 mit der Zertifizierung zum 7. deutschen „Microsoft Certified Master – SQL Server 2008“ gekrönt. Um das Jahr 2013 erfolgreich abzuschließen, wurde im Juli 2013 zum ersten Mal der MVP-Award von Microsoft für seine Unterstützung der Microsoft SQL Server Community verliehen. Damit war Uwe Ricken der erste MCM + MVP in Deutschland.

Kontakt: SQLTraining@db-berater.de

Montag, 14. Dezember 2015, 15:30 - 16:15 Uhr

William Durkin:

SQL Server 2016: Warum soll ich mir das antun?

„If it ain't broke, don't fix it!“ (Unknown)

SQL 2016 : Release ????

CTP 3.0 in Oktober veröffentlicht

SQL 2005 : End Of Life: April 12 2016

Momentan nur „Extended Support“

Dynamic Data Masking

```

25
26 CREATE TABLE dbo.MaskedTable
27 (
28     Col1 INT NOT NULL PRIMARY KEY CLUSTERED,
29     Col2 VARCHAR(255) MASKED WITH (FUNCTION = 'default()'),
30     Col3 VARCHAR(255) MASKED WITH (FUNCTION = 'email()'),
31     Col4 INT MASKED WITH (FUNCTION = 'default()'),
32     Col5 VARCHAR(255) MASKED WITH (FUNCTION = 'partial(5,"XXXXXXX",0)')
33 );
34 INSERT INTO dbo.MaskedTable
35 ( Col1, Col2, Col3, Col4, Col5 )
36 VALUES ( 1, 'Testdata for DDM', 'Test@Test.com', 100, 'More test data' ),
37 ( 2, 'IT Tage Frankfurt 2015', 'office@informatik-aktuell.de', 200, 'pool on the roof must have a leak' );
38
39 -- SELECT als Tabellenbesitzer = Ich darf die Daten sehen
40 SELECT USER_NAME() AS WerBinIch,*
41 FROM dbo.MaskedTable
42 GO
43
44 -- SELECT als "TestUser" = Ich darf die Daten *nicht* sehen
45 EXECUTE AS USER = 'ZeroCool'
46 SELECT USER_NAME() AS WerBinIch, *
47 FROM dbo.MaskedTable
48 REVERT

```

100 %

Results Messages

	WerBinIch	Col1	Col2	Col3	Col4	Col5
1	dbo	1	Testdata for DDM	Test@Test.com	100	More test data
2	dbo	2	IT Tage Frankfurt 2015	office@informatik-aktuell.de	200	pool on the roof must have a leak

	WerBinIch	Col1	Col2	Col3	Col4	Col5
1	ZeroCool	1	xxxx	TXXX@XXXX.com	0	More XXXXXXXX
2	ZeroCool	2	xxxx	oXXX@XXXX.com	0	pool XXXXXXXX

```
CREATE TABLE dbo.MaskedTable
```

```
(
    Col1 INT NOT NULL PRIMARY KEY CLUSTERED,
    Col2 VARCHAR(255) MASKED WITH (FUNCTION = 'default()'),
    Col3 VARCHAR(255) MASKED WITH (FUNCTION = 'email()'),
    Col4 INT MASKED WITH (FUNCTION = 'default()'),
    Col5 VARCHAR(255) MASKED WITH (FUNCTION = 'partial(5,"XXXXXXX",0)')
);
```

Col1	Col2	Col3	Col4	Col5
1	Testdata for DDM	Test@Test.com	100	More test data
2	IT Tage Frankfurt 2015	office@informatik-aktuell.de	200	pool on the roof must have a leak

Col1	Col2	Col3	Col4	Col5
1	XXXX	TXXX@XXXX.com	0	More XXXXXXXX
2	XXXX	oXXX@XXXX.com	0	pool XXXXXXXX

Query Store

General
 Operation Mode (Actual): Read Write
 Operation Mode (Requested): Read Write

Monitoring
 Data Flush Interval (Minutes): 15
 Statistics Collection Interval: 1 Hour

Query Store Retention
 Max Size (MB): 100
 Query Store Capture Mode: All
 Size Based Cleanup Mode: Off
 Stale Query Threshold (Days): 367

Data Flush Interval (Minutes)
 The frequency at which query store data is flushed and persisted to disk.

Current Disk Usage

- PassSummit: 7.3 MB
- Query Store Available: 100.0 MB
- Query Store Used: 0.0 MB

Purge Query Data

- Query Store
 - Regressed Queries
 - Overall Resource Consumption
 - Top Resource Consuming Queries
 - Tracked Queries

Top 25 regressed queries during the last hour for database PassSummit

Metric: Duration (µs) | Statistic: Total

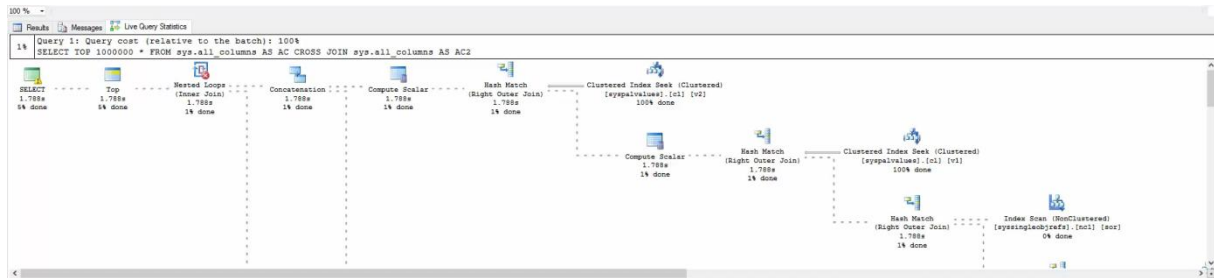
Plan 8 [not forced]

Query 1: Query cost (relative to the batch): 100%
 SELECT actual_state FROM sys.database_query_store_options

Confirmation: Do you want to force plan 8 for query 8?

Buttons: Yes, No

Live Query Statistics



Was gibt es noch?

- Reporting Services Update
- TRUNCATE TABLE auf Partition-Ebene
- Row Level Security: Funktionsgesteuerte Security auf Reihenebene
- Stretch Database – automatisches verschieben von „alten“ Daten in Azure
- Load Balanced Read-only in Availability Groups

Der Sprecher: William Durkin



William Durkin ist unabhängiger SQL Server-Consultant und auf SQL Server-Performance Tuning, Hochverfügbarkeit und Systemmigrationen/-upgrades spezialisiert. Geboren in England und seit 2001 im niedersächsischen Emsland zuhause, arbeitet er seit über 10 Jahren mit SQL Servern und hat Erfahrungen als Datenbankentwickler, Datenbankadministrator und Systemarchitekt gesammelt. William hat auf diversen IT-Konferenzen in ganz Europa Vorträge zu Themen rund um SQL Server gehalten und ist zudem Leiter der Regionalgruppe Emsland bei der PASS Deutschland e.V.

Kontakt: contact@williamdurkin.com

Dienstag, 15. Dezember 2015, 09:30 - 16:15 Uhr / ganztägig

Uwe Ricken:

Workshop - Schwerpunkt: MS SQL Server

Wann ist eine Untersuchung erforderlich?

- Berater / Consultant
 - Abnahme eines durch Drittanbieter installierten Systems
 - Problemanalyse
 - Optimierungsanalyse
- DBA
 - Problemanalyse
 - Optimierungsanalyse (naja)

Agenda

- Betriebssystem
 - Überprüfung der Umgebung des Dienstkontos
001 - A01 - System Environment - Security.sql
 - Formatierte Blockgröße der vorhandenen Speichermedien
001 - A01 - System Environment - Security.sql
 - Berechtigungen des Dienstkontos
001 - A01 - System Environment - Security.sql
- Systemkonfiguration
 - CPU / RAM / IO
001 - A02 - System Environment - Drive Latency.sql
- Datenbankkonfiguration
 - System- / Benutzerdatenbanken
 - TEMPDB (!!!)
 - Sizing / Growth / Storage
 - Verwendung
- CPU / Memory / PLE / Storage / Serverprotokolle
- SQL Server Agent Jobs
- Backups
- Indexierung
 - Wie viele Indexe sind in der Datenbank vorhanden?
 - Wie viele Indexe werden für einzelne Tabellen genutzt?
 - Wie werden Indexe genutzt?
 - Welche Indexe können gelöscht werden?
 - Wo können neue Indexe die Performance verbessern
- Statistiken
- Wie wird der Plan Cache verwendet
 - Werden AdHoc-Abfragen verwendet?
 - Werden parametrisierte Abfragen verwendet?

Wenn Microsoft SQL Server wartet

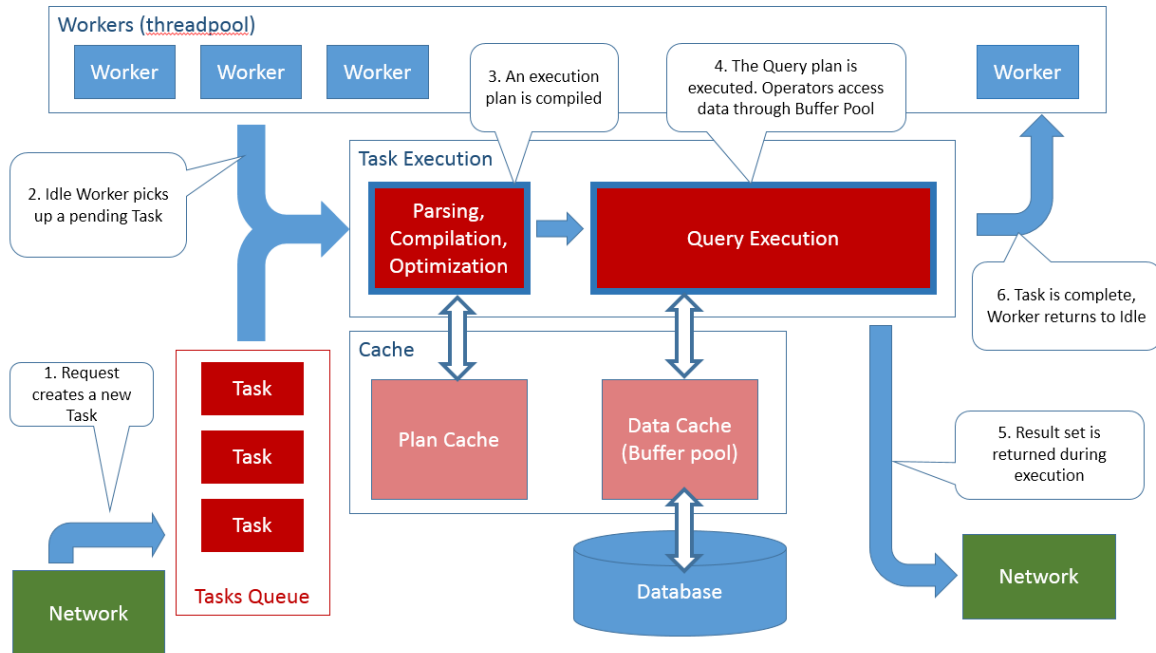
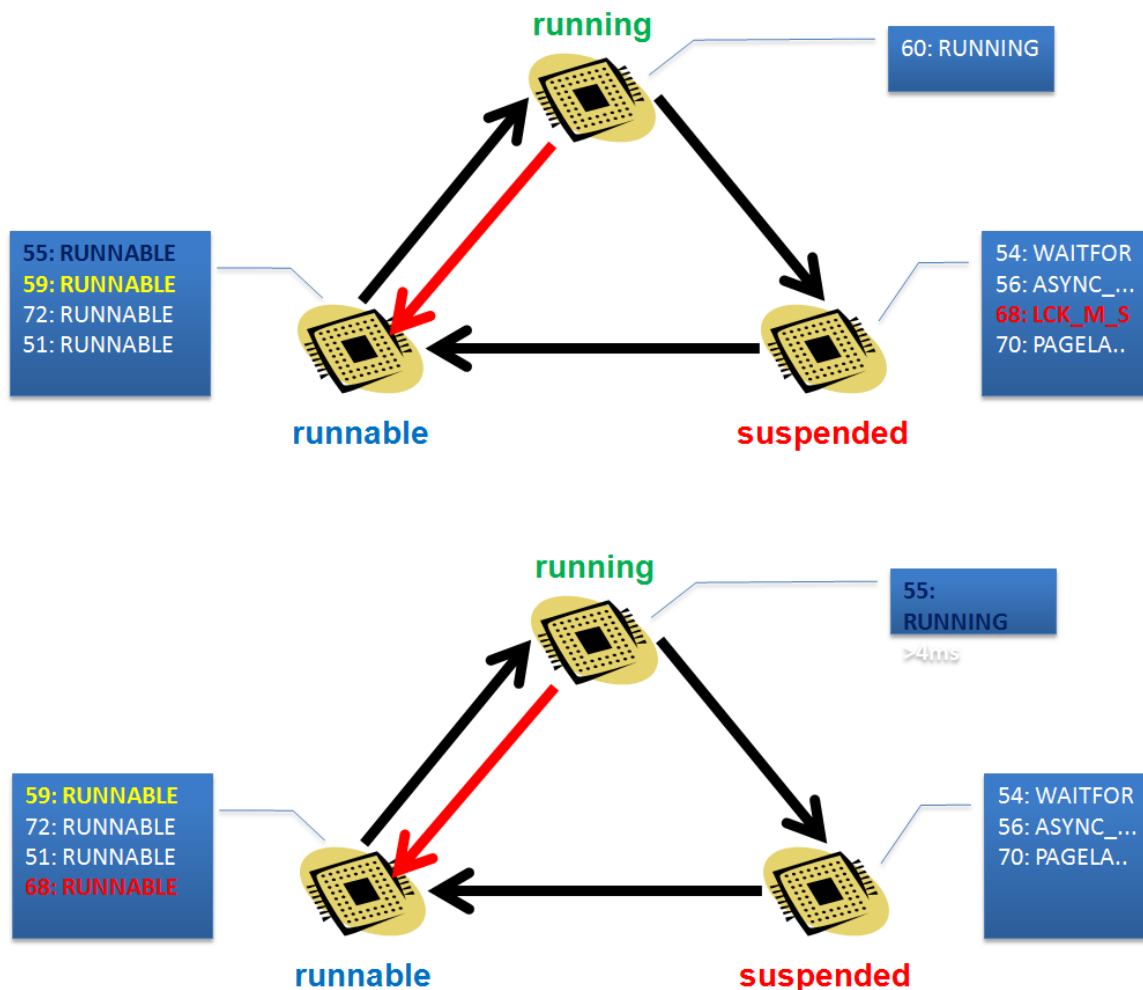
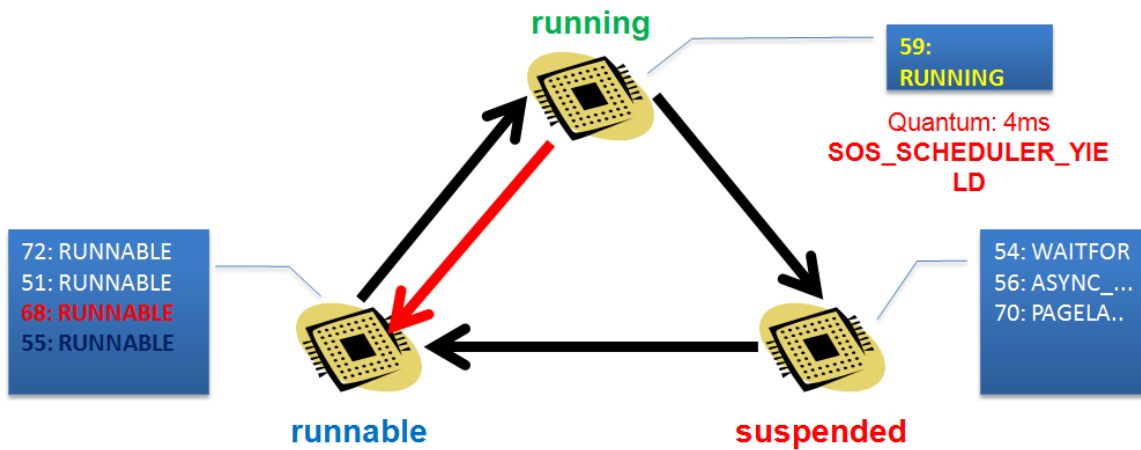


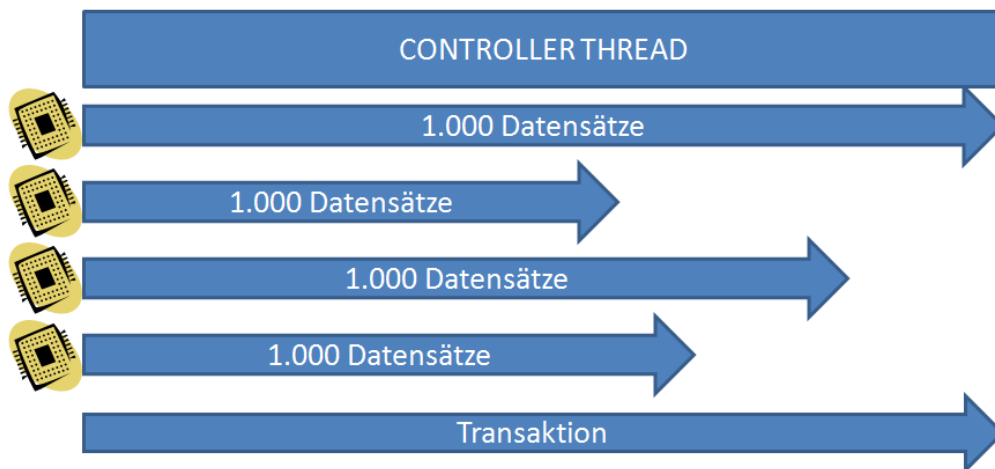
Abbildung: <http://rusanu.com/2013/08/01/understanding-how-sql-server-executes-a-query>





CXPACKET

- occurs when a parallel query is executing
 - Examine the long-running parallel query and tune it if necessary.
 - To ensure “efficient” parallelism, check that the statistics on tables and indexes used by the query are up to date.



SOS_SCHEDULER_YIELD

- This common wait isn't necessarily a problem
- It indicates that SQL Server CPU scheduling is working efficiently.
 - RUNNING | SUSPENDED | RUNNABLE

ASYNC_IO_COMPLETION

- This occurs when SQL Server is waiting for asynchronous I/O operations to finish
- Look for additional waits in the top ranking
 - IO_COMPLETION
 - LOGMGR
 - WRITELOG

- PAGEIOLATCH_*

ASYNC_NETWORK_IO

- Your application code may not be retrieving data correctly!
- There are very large data sets being requested by the client!
- An application is requesting large amounts of data!

THREADPOOL

- SQL Server assigns a scheduler to each CPU core to manage internal threads.
- Some of these threads are worker threads that are assigned to query requests on a one-to-one basis.
- If every worker thread is busy, additional requests can't be assigned straight away.
- SQL Server attempts to manage requests more efficiently by creating a pool of worker threads, so the next available worker thread from the pool will pick up the task.
- If every worker thread remains busy, SQL Server can't accept new requests and will hang.

Cores	32 bit	64 bit
<=4	256	512
5 – 8	288	576
9 – 16	352	704
32	480	960

WRITELOG

- This wait occurs when data in the log cache is being flushed to disk.
- Whenever a data page is updated, it's written to the buffer cache and the log cache.
- Data in the log cache is then written to the transaction log file on the physical disk once the transaction is complete.

PAGEIOLATCH_??

- Buffer Latch for a page object on the disk
 - DT (destroy)
 - EX (exclusive)
 - KP (keep)
 - SH (shared)
 - UP (update)

PAGELATCH_??

- PAGELATCH_* latches are lightweight, non-configurable internal locks used by SQL Server's storage engine to protect concurrent access to pages in the BUFFER POOL

Untersuchung des Betriebssystems

- Physikalische / Virtuelle Umgebung
 - Wie viele CORES / RAM?
 - Gibt es mehrere HDD im System?
 - Wie sind die HDD formatiert?

Betriebsumfeld des SQL Server

- Wie viel RAM wird dem SQL Server zugeteilt?
- Kann SQL Server alle Prozessoren verwenden?
- Welche Sicherheitseinstellungen sind konfiguriert worden?
 - Windows Authentifizierung
 - Mixed Authentifizierung
- Welche Berechtigungen hat das Dienstkonto?
 - Ist Instant File Initialization aktiviert?
 - Ist „Lock Pages in Memory“ aktiviert?
- Datenbankeinstellungen
 - FILLFACTOR
 - Standardverzeichnisse für Datenbanken
- Erweiterte Konfigurationseinstellungen
 - Max Degree of Parallelism
 - AdHoc Workloads
 - Gibt es konfigurierte Ressource-Governour-Einstellungen

Schwachstelle TEMPDB

- Verwendungsarten
 - READ COMMITTED SNAPSHOT ISOLATION
 - HASH- /SORT-Operationen
 - Durch Benutzer angelegte temporäre Tabellen
 - Tabellenvariablen
- Alle Operationen wollen Platz in TEMPDB reservieren
- Reservierung erfolgt über PFS / GAM / SGAM

Konfiguration der Datenbanken

- Wie sind Datenbanken auf Storage verteilt?
- Gibt es eine Trennung zwischen DATA und LOG?
- Welchen Wachstumswert haben Datenbanken als Standard eingestellt?
- Wie groß sind die Datenbanken?
- Wie viel Platz ist in den Datenbanken noch frei?

Dokumentation

- Beginnen Sie mit Informationen zu Ihrer Person
- Beschreiben Sie den Grund für die Erstellung des Berichts
 - Gab es Probleme, die – plötzlich – aufgetreten sind?
 - Ist es eine initiale Untersuchung vor einem Go Live?
- Weisen Sie darauf hin, dass eine Hardware- Untersuchung nicht Bestandteil der Dokumentation ist, wenn Sie keinen Zugriff auf die einzelnen Hardwarekomponenten haben
- Legen Sie eine Struktur für Ihr Dokument fest, dass der Untersuchungsreihenfolge entspricht!
- Begründungen durch Fußnoten mit Verweisen zur Dokumentation begleiten
 - BITTE KEINE FORENBEITRÄGE!
 - Möglichst nur Links zum Hersteller

Der Sprecher: Uwe Ricken



Uwe Ricken ist seit 1992 im IT-Umfeld tätig. Den Weg zum Microsoft SQL Server hat er im Jahre 1999 gefunden, als er den Auftrag für die Erstellung einer Mitgliederverwaltung für die Amerikanische Handelskammer in Deutschland erhielt. Dieses System wurde anschließend in fünf weiteren europäischen Ländern implementiert. Die ursprüngliche Leidenschaft für die Entwicklung von Datenbankmodellen mit Microsoft SQL Server wurde dann ab dem Jahre 2007 zusätzlich um die Funktionen und Aufgaben als DBA bei der Deutschen Bank in Frankfurt am Main erweitert. Mehr als 6 Jahre Praxiserfahrung als DBA und mehr als 14 Jahre Erfahrung als Entwickler von Datenbankmodellen wurden im Mai 2013 mit der Zertifizierung zum 7. deutschen „Microsoft Certified Master – SQL Server 2008“ gekrönt. Um das Jahr 2013 erfolgreich abzuschließen, wurde im Juli 2013 zum ersten Mal der MVP-Award von Microsoft für seine Unterstützung der Microsoft SQL Server Community verliehen. Damit war Uwe Ricken der erste MCM + MVP in Deutschland.

Kontakt: SQLTraining@db-berater.de

Jetzt schon einplanen:

Wissen Kompakt – MS SQL Server

Wir bieten Ihnen ein zweitägiges Seminar **mit Uwe Ricken** Anfang März sowie Anfang Juni 2016 in Frankfurt/M. im Hotel Maritim an der Messe an. Jeder Tag ist einzeln buchbar.

SQL Server: Performance-Analyse und Optimierung

Montag, 07.03.2016 und Mittwoch 01.06.2016

MS SQL Server: Indexanalyse und Performancesteigerung

Dienstag, 08.03.2016 und Donnerstag 02.06.2016

Mehr Informationen: <http://www.informatik-aktuell.de>

Mittwoch, 16. Dezember 2015, 09:00 - 09:45 Uhr

Patrick Guillebert:

Cassandra NoSQL - Architektur und Anwendungsfälle

Übersicht

Was macht die NoSQL-Datenbank Cassandra so skalierbar und hochverfügbar?

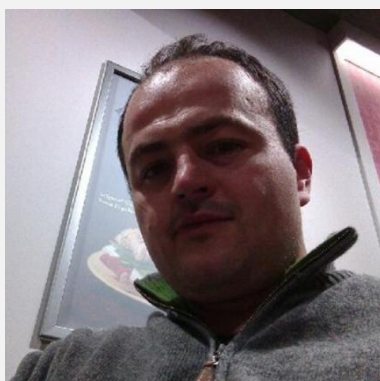
Cassandra ist eine verteilte, spaltenorientierte NoSQL-Datenbank, die leicht weiterentwickelt werden kann und die Replikationsmechanismen von Dynamo nutzt, gleichzeitig aber nach außen die Datenstruktur von BigTable anbietet. Sie ist auf hohe Skalierbarkeit und Ausfallsicherheit bei großen, verteilten Systemen ausgelegt. Die Daten werden in Schlüssel-Wert-Relationen abgelegt.

Cassandra ist offen und in Java implementiert. Die Implementierung wurde als freie Software unter den Bedingungen von Version 2 der Apache-Lizenz veröffentlicht. Im März 2009 wurde das Projekt bei der Apache Software Foundation in den Apache Incubator aufgenommen. Im Februar 2010 wurde Cassandra von der Apache Software Foundation zum „Top-Level“-Projekt erklärt. Die 2011 erschienene Version 0.8 führte die Cassandra Query Language (CQL) ein, eine Abfragesprache mit SQL-ähnlicher Syntax. Cassandra wird bei Twitter, Digg und Reddit genutzt. Auch Facebook verwendete Cassandra lange Zeit und bediente Millionen von Nutzern.

Erfahren Sie in 45 Minuten die wichtigsten Konzepte die dies ermöglichen – untermauert von realen Anwendungsfällen.

Wir erläutern Ihnen die Topologie eines Clusters, die Request Coordination, als auch das Replication- und Consistency-Management welche die Stärken von Cassandra im Detail aufzeigen.

Der Sprecher: Patrick Guillebert



Patrick Guillebert ist seit 16 Jahren in der IT-Branche tätig, insbesondere in der Erprobung/Testing und Anwendung neuer Technologien auf innovative Projekte im Bereich der SaaS-Anwendungen und PaaS-Plattformen. Nach 7 Jahren als CTO in Startup-Unternehmen arbeitet Hr. Guillebert bei DataStax als Solutions Architect, um Unternehmen bei der Umsetzung der Cassandra und der DataStax Enterprise-Plattform erfolgreich zu unterstützen.

Kontakt: thomas.gregg@datastax.com

Mittwoch, 16. Dezember 2015, 09:45 - 10:30 Uhr

Mario Briana:

Wann machen NoSQL-Datenbanken Sinn?

Übersicht

Die Informationsflut ist ununterbrochen. Täglich entstehen in Unternehmen riesige Mengen an Daten - aus Dokumenten, Kommunikation, Social Media, der Produktion und vielen weiteren Quellen. Das Problem ist offensichtlich: Allein schon das Datenmanagement ist aufwändig und teuer. Was aber, wenn die Informationen auch schnell bereitgestellt werden müssen, in vielen unterschiedlichen Formaten vorliegen und auf den ersten Blick nicht in das klassische Raster der relationalen Datenbanken passen?

Über einen Einsatz einer NoSQL-Datenbank sollte überall dort nachgedacht werden, wo eine SQL-Datenbank an ihre Grenzen stößt oder zur Erfüllung der Aufgabe aufwändige architektonische Anpassungen notwendig wären. In diesem Vortrag wird auf die spezifischen Unterschiede von NoSQL-Datenbanken zu relationalen Datenbanken eingegangen. Darüber hinaus wird vermittelt, bei welcher Art von Daten und Anwendungsszenarien NoSQL-Datenbanken Vorteile bieten. Dem Vortrag liegen die Funktionen der NoSQL-Datenbank von MarkLogic zugrunde.

Der Sprecher: Mario Briana



Mario Briana arbeitet seit Mai 2015 bei der MarkLogic GmbH als Solution Architekt. Seine Aufgabe ist, Software-Architekturen und -Lösungen basierend auf der MarkLogic Enterprise NoSQL-Datenbank mit Kunden und Partnern zu konzipieren und zu entwickeln. Dies reicht von der Positionierung von NoSQL in Architekturen über alle Themen für Architekten und Entwickler wie z. B. Sicherheit, Datenmodellierung und den verschiedenen Entwicklungsoptionen und deren Einsatzmöglichkeiten. Vor seiner Zeit bei MarkLogic war Mario Briana in verschiedenen Architekten-Rollen tätig, u. a. war er bei TrustedSafe, Brainloop und sammelte 15 Jahre Erfahrung bei Microsoft.

Kontakt: Mario.Briana@marklogic.com

Mittwoch, 16. Dezember 2015, 10:45 - 11:45 Uhr

Jörg Kleinz:

NoSQL-Graphdatenbanken – nicht nur für Big Data

Data is the center of the universe; applications are ephemeral.

Diese Kernaussage des <http://datacentricmanifesto.org/> fordert radikales Umdenken im Umgang mit Daten. Daten sind das neue „Zentrum des Universums“ in Unternehmen – Anwendungen sind „flüchtig, gar vergänglich“.

Nicht jeder wird sich gleich dem data centric manifesto anschließen. Aber Daten so zu strukturieren, dass sie nicht nur für eine Anwendung taugen sondern wiederverwendet werden können, so dass sie auch die Fragen, die der Fachbereich morgen vielleicht haben wird, beantworten können – das Herausforderungen alle Unternehmen in unserer „Data economy“. Egal unter welchem Schlagwort das läuft: Datenintegration, strategische Informationsarchitektur, Master Data Management etc.

Dabei müssen wir vernetzt denken: Wenn wir Daten, die in verschiedenen Quellen nach unterschiedlichen Aspekten strukturiert wurden, zusammenführen und dabei die vorhandenen Redundanzen eliminieren, entsteht automatisch eine Netzstruktur. Die Knoten in einem solchen Netz sind Informationsobjekte, welche Ihre Eigenschaften aus den unterschiedlichen Datenquellen beziehen, aber dann auch mit weiteren, z.B. externen Quellen in Beziehung gesetzt werden können. Das Informationsobjekt sollte der single-point-of-contact für anfragende Applikationen sein.

Die meisten Unternehmen haben hier auch schon Versuche hinter sich und daraus gelernt: so etwas geht nicht als big bang, wir müssen Wege finden, klein anzufangen, und Teilerfolge zu erzielen. Es ist unrealistisch anzunehmen, dass in einem einzigen Schritt das „strategische Datenmodell“ eines Unternehmens eruiert und dann zudem noch von einem auf den anderen Tag implementiert werden kann. Vielmehr zeigt die Erfahrung, dass man sich diesem Ziel in überschaubaren Schritten und iterativ nähern muss.

Eine weitere große Herausforderung ist dabei das Phänomen Schatten-IT - Nutzer warten nicht mehr darauf, dass die IT ihren Anforderungen, Bedarfen und Wünschen nachkommt. Heute sind Nutzer soweit, sich aus den Fachabteilungen heraus nach Lösungen umzusehen und sie „heimlich“ zu betreiben, um ihre Arbeit sach- und zielgerichtet erledigen zu können. So entsteht eine gefährliche Schatten-IT, die diese „Lösungen“ werden IT-seitig nicht unterstützt, niemand garantiert für Datenintegrität und das sichere Vorhalten der Daten. Um diesem Trend zu begegnen darf sich die IT nicht länger als alleiniger Herrscher und Verwalter von Daten verstehen; die Fachbereiche wollen und müssen sich an der Arbeit mit den Daten beteiligen.

Welche Art von Datenbank brauchen wir dafür?

Wir haben gesehen, dass der Umgang mit vernetzten Daten eine wichtige Voraussetzung ist - Dabei geht es nicht darum die Gesamtheit der Datenbankinhalte in einer Datenbank zu vereinigen, sondern darum, die unternehmensrelevanten Informationsobjekte explizit zu materialisieren, allgemein benötigte Informationen dort bereitzustellen und als „Wegweiser“ zu spezifischen Daten zu dienen.

Wir haben auch gesehen, dass wir Agilität und Flexibilität brauchen. Oft wollen wir uns einem strategischen Datenmodell in kleinen Schritten annähern, möglicherweise im Rahmen der Implementierung konkreter Anwendungen, oder auch im Rahmen der Auflösung von Schatten-IT Systemen, welche i.d.R. wertvolle Daten-Assets enthalten, die aber noch unzugänglicher sind als die Bestände der traditionell relationalen Systeme.

Das bedeutet aber nicht, dass wir gleich dem schemalosen Ansatz vieler NoSQL-Datenbanken folgen sollten. Im Gegenteil: Das Daten- und Relationschema der entstehenden Informationsobjekte sollte verbindlich und qualitätsgesichert sein, da es die strategische Sicht des Unternehmens auf die Ressource Daten widerspiegelt. Ein verbindliches, nachvollziehbares Schema erleichtert u.a. die Governance der Strukturen über einen längeren Zeitraum, die Definition gezielter Analysen und Abfragen, die Steuerung von Zugriffsrechten, etc. Idealerweise enthält das Schema auf einer Metaebene die Möglichkeit Entscheidungsprozesse zu dokumentieren, die zur Erstellung des konkreten Schemas geführt haben.

Wir wollen also die Agilität von NoSQL-Datenbanken mit der information governance relationaler Systeme kombinieren.

Schließlich haben wir das Thema Schatten-IT angesprochen: Die Erschaffung einer eigenständigen Unternehmensdaten-Ressource ist nur teilweise eine technische Herausforderung. Insbesondere benötigt es fachliches Verständnis die Semantik vorhandener Quellen zu erkennen und zu erklären, sinnvolle Zusammenhänge zu etablieren und Anforderungen der Nutzer auf das semantische Modell abzubilden. Häufig sind die eingesetzten Technologien zur Strukturierung und Speicherung der Daten dem Fachanwender völlig unzugänglich, während sich dem technisch orientierten IT Experten die Anforderungen der Fachanwender oftmals nur schwer erschließen. Die eingesetzten Werkzeuge sollten aber zusätzlich, z.B. durch Visualisierung von Schemata und Daten eine Partizipation der Fachanwender bei der Strukturierung der Informationsobjekte ermöglichen und auch die Schemaentwicklung direkt anhand von Beispielen unterstützen. Im Idealfall bietet die eingesetzte Technologie eine Kommunikationsplattform zur Informationsstrukturierung, welche die Kluft zwischen IT und Fachbereich auf ein Minimum reduziert.

Schlüsseltechnologie Semantische Graphdatenbanken

Die Verbindung von Agilität und schema-basierter Governance, die Zugänglichkeit für Nutzer aus den Fachbereichen und der angemessene Umgang mit vernetzten Daten sind die

charakteristischen Merkmale einer semantischen Graphdatenbank. Die Organisation der Daten in Graphen statt in Tabellen ist in vielen Anwendungsszenarien dem Fachanwender eingängiger, weil es eher mit der eigenen Denkweise über Zusammenhänge und Kontextinformation korrespondiert. In der semantischen Spielart der Graphdatenbanken kommt das verbindliche, aber jederzeit flexibel änderbare Schema dazu. Die Analysten von Gartner sehen diese Kombination als einen der Top 10 Technologietrends 2016 an: „Information has always existed everywhere but has often been isolated, incomplete, unavailable or unintelligible. Advances in semantic tools such as graph databases as well as other emerging data classification and information analysis techniques will bring meaning to the often chaotic deluge of information“ (Quelle: <http://www.gartner.com/newsroom/id/3143521>)

Der Vortrag stellt die semantische Graphdatenbank k-infinity und ihre Anwendung in der Datenintegration und dem Aufbau von Informationsarchitekturen anhand von Kundenprojekten vor.

Der Sprecher: Jörg Kleinz



Dipl.-Informatiker Jörg Kleinz ist Mitgründer und Geschäftsführer der intelligent views gmbh. Er leitet den Software-Entwicklungsbereich und ist darüber hinaus verantwortlich für den Bereich Forschung und technologische Strategie. Zusätzlich führt Jörg Kleinz das Projekt- und Finanzcontrolling bei intelligent views und verantwortet vertrieblich den Geschäftsbereich Public Security.

Schwerpunkt seiner Tätigkeit ist die Beratung, Konzeption und Durchführung erfolgreicher Wissensmanagement-Projekte, insbesondere im öffentlichen Sektor, Schwerpunkt Sicherheit. Während seiner Tätigkeit am IPSI (Institut für Integrierte Publikations- und Informationssysteme (vormals GMD, heute Fraunhofer-Gesellschaft)) und der TU Darmstadt entwickelte er zusammen mit seinen Kollegen prototypische Werkzeuge zur Modellierung und Nutzung semantischer Netze. Seine Forschungsschwerpunkte lagen dabei in den Bereichen "wissensbasierte Publikations- und Visualisierungssysteme".

Mittwoch, 16. Dezember 2015, 11:45 - 12:30 Uhr

Michael Wittig:

Datenbank am Limit. Spaltenorientiert als Ausweg

Wir alle kennen und schätzen SQL- und NoSQL-Datenbanken. Doch es gibt Anwendungsfälle, in denen diese Datenbanken an ihre Grenzen stoßen. Zum Beispiel bei der Analyse von Finanzmarktdaten. Dieser Vortrag zeigt eine Lösung, bei der die natürliche Sortierung der Daten von größter Bedeutung ist und bei der es keine Trennung zwischen Datenbank, Daten und Abfragesprache gibt.

Bei genauer Betrachtung von Finanzmarktdaten fallen drei Eigenschaften auf: Eine natürliche Sortierung nach Zeit, eine feste Struktur sowie "Nur-Hinzufügen" (append-only) Schreibvorgängen.

Zeit	Käufer	Verkäufer
11:00:00.000	5000 @ 84,81	500 @ 84,82
11:00:00.009	4700 @ 84,81	200 @ 84,82
11:00:00.025	1000 @ 84,82	900 @ 84,83
11:00:00.207	7000 @ 84,81	800 @ 84,82
11:00:00.555	2000 @ 84,81	200 @ 84,82
11:00:00.873	1500 @ 84,81	100 @ 84,82

Abbildung 1: Finanzmarktdaten sind zeitlich sortiert und strukturiert

Desweiteren fallen die Daten in Echtzeit an, müssen aber sowohl in Echtzeit als auch historisch analysiert werden können. Am Ende gewinnt wer die Daten am schnellsten in Handlungsanweisungen an der Börse umwandeln kann. Der Vortrag zeigt in zwei Teilen eine Lösung für historische sowie Echtzeit-Daten-Analyse auf und endet in der Vereinigung beider Lösungen.

Die Sortierung in Kombination mit einer festen Struktur ist ein idealer Anwendungsfall für spaltenorientierte Datenbanken. Die spaltenorientierte Speicherung und Verarbeitung hat

im wesentlichen zwei Vorteile gegenüber zeilen- oder dokumentenorientierten Datenbanken:

- Es müssen nur die benötigten Spalten gelesen und verarbeitet werden
- Die Lokalität der Daten ermöglicht effiziente Verarbeitung von der Festplatte in den Speicher bis in die CPU

Der Vortrag geht an Hand einer "Do It Yourself"-Datenbank auf den Unterschied zwischen dem zeilen- und dem spaltenorientierten Ansatz ein. Zum Abschluss des ersten Teils gibt es eine Einführung (incl. Demo) der spaltenorientierten Datenbank kdb+ von KX Systems. kdb+ ermöglicht die effiziente Speicherung und Verarbeitung von historischen Zeitreihen.

Als nächstes widmen wir uns der Frage wie Echtzeit Daten effizient analysiert und historisiert werden können. Dazu betrachten wir die Architektur des Tickdata-Systems kdb+tick von KX Systems.

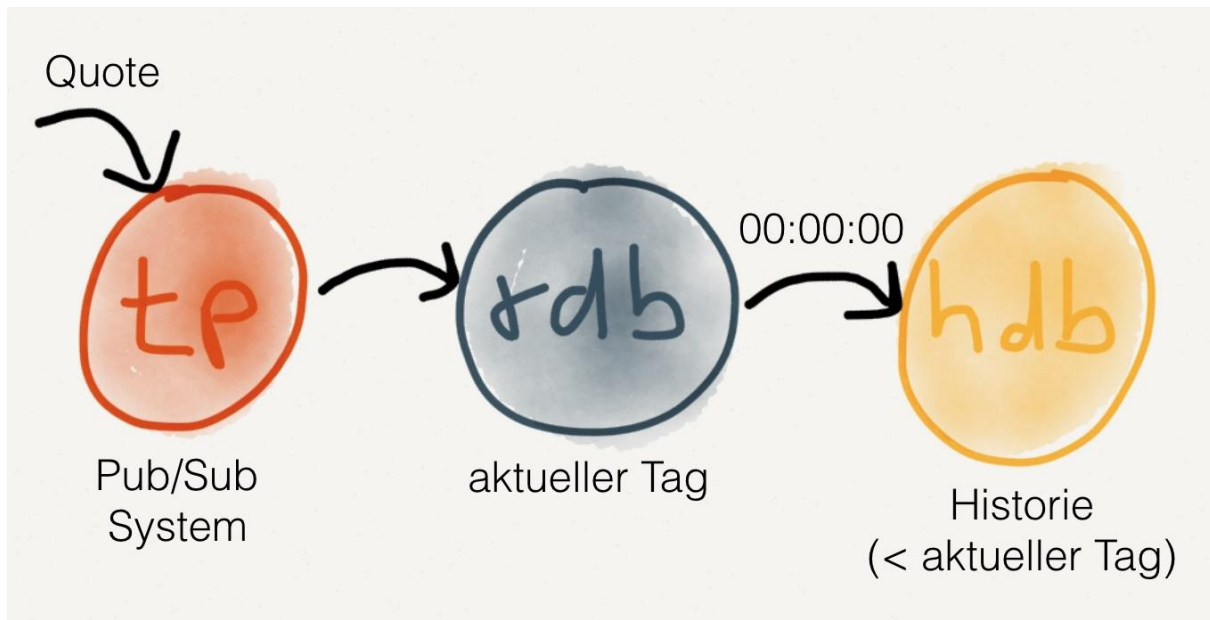


Abbildung 2: Ein Tickdata System verarbeitet Daten in Echtzeit

kdb+tick besteht aus drei Komponenten:

- Tickerplant (tp)
- kdb+
 - Echtzeit In-Memory-Datenbank (rdb)
 - Historische Datenbank (hdb)

Die Tickerplant stellt einen Publish-/Subscribe-Mechanismus zur Verfügung, mit dem Finanzmarktdaten in Echtzeit analysiert werden können. Durch geschicktes Verschalten mehrerer Tickerplants können unterschiedliche Anforderungen an Änderungshäufigkeit

effizient realisiert werden.

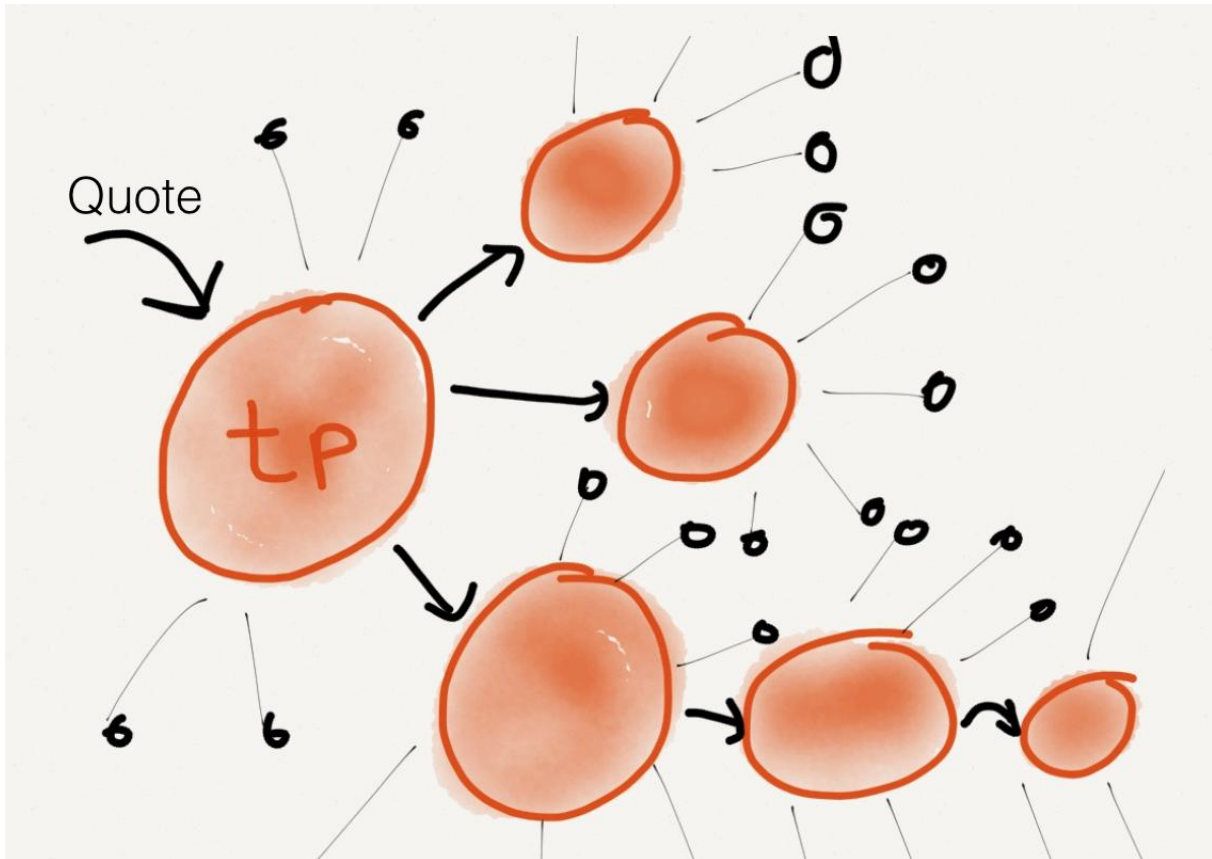


Abbildung 3: Eine Verkettung von Tickerplants ermöglicht eine effiziente Analyse

Zum Ende gibt es eine Demo, die die Verarbeitung von Finanzmarktdaten in Echtzeit demonstriert. Beide Teile werden mit einer kurzen Einführung in die Funktionale Sprache q vereint. q ist sowohl Datenbank als auch Abfragesprache um eine Abfrage so nahe wie möglich an den Daten beantworten zu können.

Der Sprecher: Michael Wittig



Michael Wittig migrierte in einem kleinen Team die IT der ersten Bank in Deutschland komplett in die AWS Cloud. Er hat einen Schwerpunkt im algorithmischen Handel, wo er tausende Gigabytes an Finanzmarkt-Zeitreihen analysierte. So benutzt er die AWS Cloud für die historische und Echtzeit-Analyse von Finanzmarktdaten mit einer Vielzahl an Technologien und Programmiersprachen. Heute berät er mit seinem Bruder andere Unternehmen bei der Nutzung der AWS Cloud, um ihren Datenschatz zu bergen, und ist Entwickler der SaaS Zeitreihen-Datenbank TimeSeries.Guru.

Kontakt: michael@widdix.de

Mittwoch, 16. Dezember 2015, 13:30 - 14:15 Uhr

Stephan Kaps:

Flyway vs. LiquiBase – Battle der Datenbankmigrationstools

Battle der Datenbankmigrationstools

Wenn es das Ziel ist, Software regelmäßig auszuliefern, gegebenenfalls mehrmals am Tag, darf bei all den derzeitigen Überlegungen zu Automatisierung von Tests, Deployments und Infrastruktur die Datenbank nicht vergessen werden. Die Automatisierung von Änderungen (Migrationen) ist auch hier unverzichtbar. Inzwischen existieren Tools für diese Aufgabe, die sich sehr gut in den Entwicklungsprozess integrieren lassen. Die zwei bekanntesten Tools werden in diesem Vortrag vorgestellt und miteinander verglichen.

Einleitung

Im Umfeld agiler Vorgehensmodelle und Initiativen wie Continuous Delivery ist es wichtig, Datenbankänderungen effizient vorzubereiten und durchführen zu können. Dabei ist es wichtig, dass für das evolutionäre Datenbankdesign unter anderem die DDL-Operationen, Migrationsskripte und Referenz- bzw. Stammdaten verwaltet und bereitgestellt werden. Sie müssen unter Versionskontrolle stehen und verlässlich reproduzierbar sein, z.B. auf diversen Testumgebungen. Ziel ist es, der Datenbank ansehen zu können, in welchem Zustand sie sich befindet, um sie im Anschluss automatisiert auf einen anderen Versionsstand migrieren zu können.

Die **aktuelle Situation** in den meisten Softwareentwicklungs-Projekten stellt sich wie folgt dar: Es existiert eine Anwendung, die meist relativ komplex ist und ein großes Datenbankmodell besitzt. Diese Anwendung wird agil entwickelt und eventuell sogar kontinuierlich deployed und ausgeliefert. Dafür existieren mehrere Umgebungen, in der die Anwendung ausgeführt wird (Entwicklung, Test, Staging, Produktion usw.). Auf jeder dieser Umgebungen muss auch die entsprechende Datenbank für die Anwendung bereit stehen, und zwar in der richtigen Version, d.h. passend zum Software-Entwicklungsstand. Denn ebenso wie die Anwendung entwickelt sich auch die Datenbank und deren Tabellen und Inhalte stetig weiter. Hier sind die Datenbankadministratoren gefragt. Sie erhalten von den Entwicklern die relevanten SQL-Skripte. Diese sollten auch gleich mit dem Hinweis versehen sein, in welcher Reihenfolge sie auszuführen sind, falls dies für die Funktionsfähigkeit entscheidend ist.

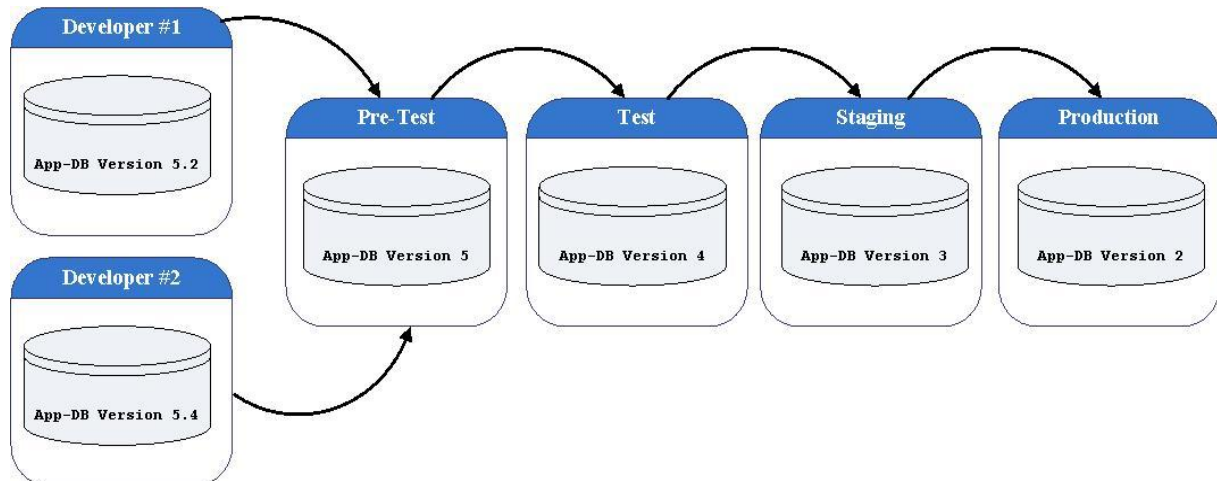


Abb. 1 - Datenbanken in div. Umgebungen

Bei manueller Pflege bedeutet dies einen hohen Aufwand, für den entsprechende Ressourcen zur Verfügung stehen müssen. Außerdem ist diese Vorgehensweise fehleranfällig, weil möglicherweise vergessen wird, ein Skript auf einer Umgebung einzuspielen oder Skripte in der falschen Reihenfolge ausgeführt werden. Vielleicht wurden aber auch Quick-Fixes für die Produktion anschließend nicht in allen Testumgebungen bereit gestellt. Des Weiteren muss nachvollziehbar und reproduzierbar sein, welche Datenbankänderungen für welchen Versionsstand der Software notwendig sind bzw. waren. Oder es stellen sich folgende, viel trivialere Fragen: Wie setzt man eine neue Instanz der Datenbank auf? Welche Skripte müssen dafür eingespielt werden, in welcher Reihenfolge und wo findet man diese überhaupt?

Aus diesen Gründen empfiehlt sich der Einsatz von Datenbankmigrationstools. Zudem ist allgemein anerkannt, dass Quellcode zu versionieren ist. Für Datenbankänderungen gilt dies auch!

Es existieren inzwischen einige Tools auf dem Markt, und zwar sowohl Open Source als auch kommerziell.

In diesem Vortrag werden die wohl bekanntesten Open Source-Tools aus der Java-Welt Flyway [<http://www.flywaydb.org>] und LiquiBase [<http://www.liquibase.org>] miteinander verglichen. Unabhängig davon existieren noch weitere Alternativen wie DbDeploy, mybatis, MIGRATEdb, DBMaintain u.v.m., die jedoch vom Funktionsumfang oder der Größe der Community nicht mit den Platzhirschen mithalten können.

Um sich dabei aber nicht auf Java zu beschränken, wird der Vergleich anhand der Kommandozeilen-Variante durchgeführt, die standalone aus einer Konsole oder aus einem Build-Skript heraus benutzt werden kann. Zunächst wird jedoch die allgemeine Funktionsweise der Tools und die jeweilige Art der Verwaltung der Datenbankänderungen erläutert.

Funktionsweise

Die vorgestellten Tools bieten eine direkte Integration in die eigene Anwendung, so dass beim Start einer neuen Version der Anwendung, die entsprechenden Datenbankänderungen durchgeführt werden (Auto-migration on startup). Eine weitere Möglichkeit ist die Integration in den Build-Prozess.

Beide Tools merken sich in einer von ihnen selbst verwalteten Metatabelle für jedes Schema den aktuellen Versionsstand der jeweiligen Datenbank. In diesen Metatabellen ist hinterlegt, welche Skripte, in welcher Reihenfolge, wann eingespielt wurden. Abbildung 2 zeigt die Metatabelle von LiquiBase (databasechangelog), Abbildung 3 das Pendant bei Flyway (genannt schema_version).

id	author	filename	dateexecuted	orderexecute	executetype	md5sum	description	comments	tag	liquibase
char	character varying(255)	character varying(255)	timestamp w	integer	character vari	character vari	character varying	character varying(255)	character varyin	character vari
1	TimTester	changelog.xml	2014-12-04	1	EXECUTED	7:a49f5873f	createTable	Tabelle liquibaseTest erstellt	Version 2.0	3.2.2
2	System	changelog.xml	2014-12-04	2	EXECUTED	7:3bb63b6af	tagDatabase	''	Version 2.0	3.2.2
3	TonyMontana	changelog.xml	2014-12-04	3	EXECUTED	7:1bb937033	createTable	Tabelle person erstellt		3.2.2
4	TonyMontana	changelog.xml	2014-12-04	4	EXECUTED	7:77a74212f	sql	''		3.2.2
5	TonyMontana	changelog.xml	2014-12-04	5	EXECUTED	7:5e8924b83	addColumn	Spalte web hinzugefügt	Version 3.0	3.2.2
6	System	changelog.xml	2014-12-04	6	EXECUTED	7:92e5fe41e	tagDatabase	''	Version 3.0	3.2.2
7	RainerZufall	create_sqltest.sql	2014-12-04	7	EXECUTED	7:fa143db4e	sql	''		3.2.2
8	RainerZufall	create_sqltest.sql	2014-12-04	8	EXECUTED	7:9a3dd5331	sql	''		3.2.2
9	RainerZufall	insert_sqltest.sql	2014-12-04	9	EXECUTED	7:e8099ba25	sql	''	Version 3.1	3.2.2
10	System	changelog.xml	2014-12-04	10	EXECUTED	7:552ff2116	tagDatabase	''	Version 3.1	3.2.2

Abb. 2 - LiquiBase Metatabelle (databasechangelog)

version_rank	installed_rank	version	description	type	script	checksum	installed_by	installed_on	execution_time	success
integer	integer	[PK] characte	character varying(200)	character vari	character varying(1000)	integer	character vari	timestamp w	integer	boolean
1	1	2	Create test table	SQL	2__Create_test_table.sql	1814441205	postgres	2014-09-10	14	TRUE
2	2	3	add testdata	SQL	3__add_testdata.sql	-1342871217	postgres	2014-09-10	7	TRUE
3	3	3.1	add testdata	SQL	3_1__add_testdata.sql	916155695	postgres	2014-09-10	7	TRUE
*										

Abb. 3 - Flyway Metatabelle (schema_version)

Die Entwickler definieren die Änderungen an der Datenbank, die für die nächste Version ihrer Software notwendig sind und stellen die Dateien unter Versionskontrolle. Beim nächsten Build oder Start der Anwendung wird zusätzlich das Hauptkommando der Tools ausgeführt (*migrate* bei Flyway bzw. *update* bei LiquiBase). Dies bewirkt, dass der aktuelle Stand der Datenbank mit den vorliegenden Migrationskripten verglichen wird. Sollten Skripte existieren, die noch nicht in die Datenbank eingespielt wurden, wird dies nun umgehend durchgeführt und in der Metatabelle vermerkt. Kein Datenbankadministrator muss manuell tätig werden.

Datenbankänderungen bei Flyway

Migrationen beim Einsatz von Flyway können in reinem (Plain-)SQL oder in JAVA definiert werden. Diese Dateien sind per Namenskonvention mit einer Versionsnummer im Präfix zu versehen, anhand derer die Reihenfolge festgelegt wird (z. B. *V4__Create_table_person.sql*).

Für die Nutzung von Flyway ist es sinnvoll, die Versionsnummern der Skripte mit den Versionsnummern der Software gleichzusetzen. Dadurch ist nachvollziehbar, welche Datenbankänderungen zu welcher Version der Software gehören.

Datenbankänderungen bei LiquiBase

Für Datenbankänderungen mit LiquiBase werden die auszuführenden Befehle in einer Datei gesammelt, der sogenannten Changelog-Datei. Diese kann in den Formaten XML, YAML, JSON oder SQL erstellt werden. Migrationen in JAVA zu definieren ist ebenso möglich.

LiquiBase ist in der Lage, die Befehle innerhalb einer Changelog-Datei auf diverse Datenbank-Management-Systeme anzuwenden, ohne auf die Unterschiede (z. B. bei den Feldtypen) explizit eingehen zu müssen.

Beim Vergleich – dem "Battle" der Datenbankmigrationstools – treten gegeneinander an: Flyway in der Version 3.2.1 gegen LiquiBase Version 3.4.1. Es werden sechs Runden ausgetragen in den Kategorien Funktionen und unterstützte Datenbanken (1), Einbindung in bestehende Projekte (2), Integrationsmöglichkeiten (3), Inbetriebnahme und Handhabung (4), verfügbare Plugins (5) und Rahmenbedingungen (6).

Unabhängig vom Ergebnis des Vergleichs hängt die Entscheidung für Flyway oder LiquiBase von individuellen Faktoren ab:

- Wird im Projekt eine spezielle Datenbank verwendet, die nur ein Tool unterstützt?
- Will man seine bisherigen SQL-Skripte 1:1 weiterverwenden?
 - dann eher Flyway
- Kann man auf Rollbacks verzichten? (z. B. durch geschickte Forward-Migrationen)
 - wenn nicht, dann LiquiBase
- Benötigt man deutschen Support?
 - dann eher Flyway
- Will man seine Testdaten mit verwalten lassen?
 - dann LiquiBase
- Will man diverse Datenbankmanagementsysteme unterstützen?
 - dann eher LiquiBase

Wahrscheinlich entscheiden eher solche Fragen, welches das geeignete Tool ist. Offen bleibt aktuell allerdings noch die Frage der Unterstützung von NoSQL Datenbanken, die bisher von keinem der beiden Tools beantwortet wird.

Insgesamt empfiehlt es sich, ein solches Tool zunächst an einem kleinen neuen Projekt auszuprobieren, bevor es in bestehende Projekte integriert wird. Es sind nicht nur technische Hürden zu überwinden, sondern vor allem kulturelle und organisatorische. Hier kann großer Widerstand entstehen. Machen Sie daher behutsam auf die Vorteile der Verwendung eines solchen Tools aufmerksam.

Bei Fragen zu den Tools oder dem Wunsch, Sie bei der Einführung zu unterstützen, können Sie mich gerne per Mail, über XING oder über Twitter (@kitenco1) kontaktieren.

Der Sprecher: Stephan Kaps



Stephan Kaps arbeitet im IT-Management beim Bundesversicherungsamt. Schwerpunkte sind Software-Architektur, Continuous Delivery, BizDevOps und Application Lifecycle Management. Zudem hat er langjährige Erfahrung als Java- und Host-Entwickler. Seine Leidenschaft sind neue Technologien und Methoden. Zudem ist er Autor diverser Fachartikel und Speaker auf Konferenzen (JAX, Continuous Lifecycle, Developer Week).

Kontakt: Kaps.Stephan@gmx.de

Mittwoch, 16. Dezember 2015, 14:30 - 15:15 Uhr

Wolfgang Epting:

Testdaten - versteckte Geschäftschance oder immanentes Sicherheitsrisiko?

Qualitative hochwertige Testdaten in der richtigen Größe und Zusammensetzung, zur richtigen Zeit am richtigen Ort steigern nachweislich die Anwendungsqualität, reduzieren die Fehlerrate im produktiven Betrieb, erhöhen die Agilität der Anwendungsentwicklung und sparen somit erhebliche Kosten. Aber welcher Entwickler oder Tester möchte in der Ausübung seines Berufes etwas eigentlich Verbotenes tun, wenn er mit personenbezogenen Daten in Berührung kommt? Daher bedarf es klarer Richtlinien und Standards in Kombination mit geeigneten Werkzeugen, um möglichen Verletzungen des Bundesdatenschutzgesetzes vorzubeugen.

In diesem Vortrag erfahren Sie, welche Chancen eine Flexibilisierung des Testdatenmanagements für Ihr gesamtes Unternehmen bringt. Testdatenmanagement kann nur dann zielgerichtet eingesetzt werden, wenn alle Möglichkeiten adressiert und alle Facetten abgedeckt werden. Diese umfassen sowohl die Bereitstellung von kopierten Produktionsdaten, deren vertrauliche Bestandteile anonymisiert werden müssen, als auch die Synthetisierung von komplett neuen Daten.

Sie erfahren darüber hinaus in diesem Vortrag, wie Testdatenmanagement sinnvoll mit Testautomation kombiniert werden kann. Automatische Tests bringen nur dann ihren vollen Nutzen, wenn Sie iterativ mit gleichen Ausgangsdaten durchgeführt werden können, um vergleichbare Testserien zu erzeugen. Das Zurücksetzen von verbrauchten Testdaten ist daher ein integraler Bestandteil eines modernen Testdatenmanagement. Apropos modern – natürlich werden in diesem Vortrag auch Themen wie Big, Data, Hadoop, NoSQL im Kontext des Testdatenmanagements vorgestellt.

Der Sprecher: Wolfgang Epting



Wolfgang Epting beschäftigt sich seit über 20 Jahren mit Datenbanktechnologien. Nach dem Studium arbeitete er als Datenbankadministrator in einem mittelständischen Unternehmen und übernahm sukzessive die Gesamtverantwortung für die gesamte Datenhaltung. Als technischer Vertriebsrepräsentant war er zunächst bei IBM sechs Jahre deutschlandweit und ist nun seit einem Jahr bei Informatica europaweit für Lösungen im Bereich Information Governance und Life Cycle Management zuständig.

Kontakt: wepting@informatica.com

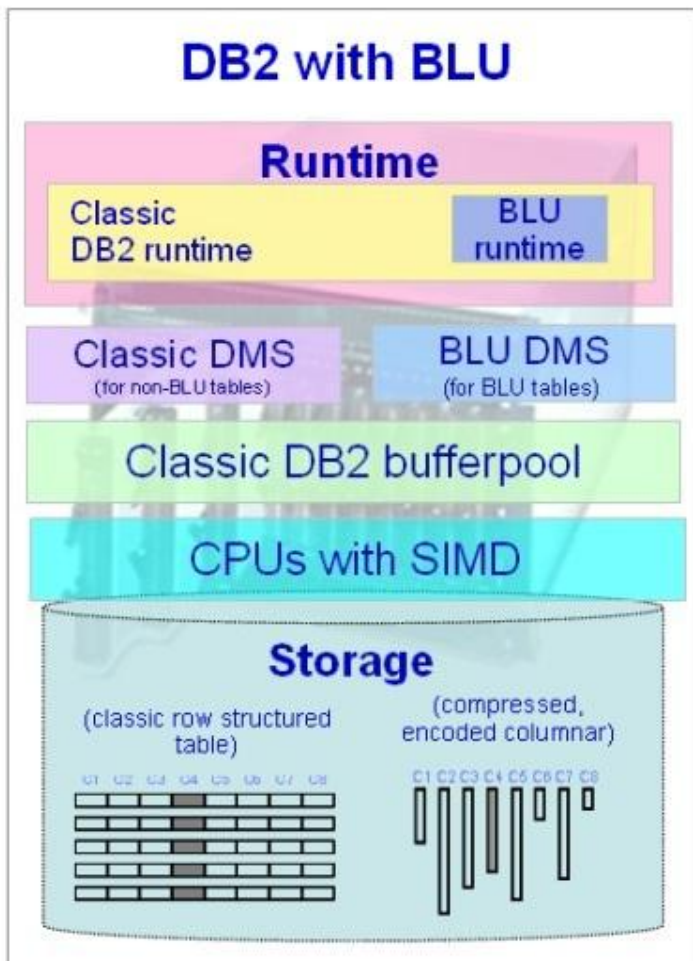
Mittwoch, 16. Dezember 2015, 15:30 - 16:15 Uhr

Stefan Hummel:

DB2 In-Memory - Eine Technologie nicht nur für typisches OLAP

DB2 with BLU Acceleration ist die nächste Generation der Datenverarbeitung. Durch die Kombination von unterschiedlichen Technologien wird die Verarbeitung von großen Datenmengen um ein Vielfaches beschleunigt. Dadurch ist es möglich, Auswertungen von historischen Daten und Echtzeitdaten innerhalb kürzester Zeit durchzuführen.

BLU Acceleration kombiniert die Technologien In-Memory-Computing, spaltenorientierte Tabellen, Komprimierung, Data Skipping sowie SIMD (Single Instruction Multiple Data) miteinander. Darüber hinaus werden hardwarenahe Funktionen genutzt, um die Verarbeitung der Daten soweit wie möglich in den CPU Kernen durchzuführen.



Wenn DB2 with BLU Acceleration mit Big Data Technologien kombiniert wird, lassen sich weitere Use-Cases besser unterstützen wie

- Verknüpfung von strukturierten mit unstrukturierten Daten über BigSQL
- Sandboxing
- Advanced Analytics
- ETL Processing

Mittels Schattentabellen ist BLU Acceleration aber auch für analytisches SQL in transaktionalen Systemen (OLTP) geeignet. Dadurch ist eine Auswertung von Echtzeitdaten möglich.

BLU Acceleration ist in DB2 10.5 integriert. Änderungen an SQL oder Datenbankschema sind nicht notwendig. Die Fachbereiche können wie bisher gewohnt auf ihre Daten zugreifen und von der Performancesteigerung unmittelbar

profitieren. DB2 kann sowohl im eigenen Rechenzentrum als auch in der Cloud genutzt werden.

Der Sprecher: Stefan Hummel



Stefan Hummel ist seit über 20 Jahren im Datenbankumfeld tätig. Bei IBM ist er im Technischen Vertrieb verantwortlich für DB2 unter Linux, Unix und Windows. Seine Schwerpunkte sind die Migration von Wettbewerbsdatenbanken zu DB2, DB2 für SAP und DB2 in der Cloud. Vor seiner Beschäftigung bei IBM war er zuständig für Entwicklung und Performancetuning von Oracle-Anwendungen.

Kontakt: stefan.hummel@de.ibm.com

Mittwoch, 16. Dezember 2015, 16:30 - 17:15 Uhr

Philipp Krenn:

Lasst uns die Datenbank in der Cloud hosten...

Einführung

Der Start eines Projekts in der Cloud ist denkbar einfach: Man zückt seine Kreditkarte und kann sofort starten. Vorbei sind die Zeiten, in denen man bei einem Menschen bestellen musste, danach stunden- oder tagelang auf das Produkt gewartet hat und schlussendlich zumindest ein Jahr lang daran gebunden war.

Natürlich kann man sämtliche Services auch selbst betreiben. Nicht umsonst heißt es so treffend: "Es gibt keine Cloud, nur anderer Leute Computer." Aber möchte man wirklich alles selbst betreiben? In aktuellen Projekten werden häufig mehrere Datenbanken eingesetzt. Wer hat schon einen Spezialisten, der sich ausreichend auf eine dieser Datenbanken konzentrieren kann und das nötige Wissen und Tooling (Logging, Monitoring, Backups, etc) bereitstellt? Anstatt das Thema einem Mitarbeiter als zwanzigste Aufgabe anzuhängen, würde es sich doch anbieten ein entsprechendes Service in der Cloud zu nutzen. Der Anbieter davon kann sich voll und ganz auf den Betrieb davon konzentrieren und sollte dadurch wesentlich mehr Erfahrung und Ressourcen dafür zur Verfügung haben, als jede In-House Lösung es könnte.

Soweit klingt die Argumentations-Kette schlüssig und für viele Firmen ist es wohl tatsächlich der sinnvollste Ansatz. Nur wie nahe liegen Theorie und Praxis in diesem Fall zusammen? Dazu wollen wir uns zuerst allgemein relevanten Punkte ansehen und abschließend einen konkreten Anwendungsfall: MongoDB as a Service.

Generelle Überlegungen

Skalierbarkeit

Skalierbarkeit ist eines der zentralen Versprechen der Cloud. Wenn ich morgen die zehnfache Anzahl an Anfragen bewältigen muss, wird mich das mehr kosten, aber die Anpassung dafür sollte transparent im Hintergrund möglich sein.

Security

Selbstkonfiguration, sofortige Verfügbarkeit und transparente Skalierbarkeit machen das Thema Sicherheit schwieriger als bei klassischen Installationen. Ein wichtiger Punkt ist, ob man zusätzlich zur Authentifizierung noch den Zugriff über Firewall-Regeln einschränken kann. Ein weiterer wichtiger Schritt ist die Absicherung aller Accounts mit Zwei-Faktor-Authentifizierung (2FA), was mittlerweile von jedem Anbieter unterstützt werden sollte.

Ansonsten muss man dem Provider vertrauen, dass er seine Hausaufgaben macht und neben dem eigentlichen Service auch die Mitarbeiter-Zugriffe ausreichend absichert.

Monitoring und Logging

Idealerweise wird hier alles zur Verfügung gestellt und man muss nur die Benachrichtigung konfigurieren – sei es über Email oder als Integration in externe Systeme wie PagerDuty.

Erfahrungsgemäß ist das Logging aber wesentlich weniger feingranular, als wenn man es selbst betreiben würde.

Backups

Mit Backups verhält es sich wie mit Schrödingers Katze: So lange man sie nicht versucht hat wieder herzustellen, weiß man nicht, ob sie existieren oder nicht.

Wie leicht kann man auf seine Backups zugreifen? Ist es möglich sie einfach herunterzuladen und lokal zu testen (was auch die Migration zu einem anderen Betreiber oder zum Eigenhosting wesentlich erleichtern würde)? Oder muss man sich auf einen Eintrag im Webinterface des Providers verlassen, dass sie zu einem gewissen Zeitpunkt gemacht wurden und per Klick (hoffentlich nicht ausschließlich über den Support) wiederherstellbar sind?

Support

Falls doch einmal etwas schief läuft: Wie lange dauert es, bis der Support das Problem analysiert und danach beheben kann? Möglicherweise ist die Reaktionszeit vom Provider wesentlich besser als die eigene. Andererseits ist es äußerst unangenehm, wenn es Probleme gibt und man eine ungewisse Zeit auf den Support warten muss, in der man selbst nichts sinnvolles tun kann.

Testbarkeit

Ein unerwarteter Effekt von der Migration in die Cloud ist, dass das Testen von Fehlerszenarien plötzlich schwieriger wird. Man kann nicht mehr zu einem Server gehen und Strom- oder Netzkabel herausziehen, das Betriebssystem neu starten, den Prozess stoppen, etc. Die verbliebenen Testmöglichkeiten beschränken sich meistens auf Datenbankupdates (Prozess muss neu gestartet werden oder es findet ein Failover statt) oder lokale Firewall-Regeln um einen Ausfall zu simulieren.

Eigene Erfahrungen

Wir hatten das "Glück" in fast allen Bereichen Probleme zu haben. Wobei man festhalten muss, dass wir nie Daten verloren haben und nie eine, für Kunden sichtbare, Service-Unterbrechung hatten.

Allerdings haben wir folgende Erfahrungen gemacht:

- **Gestoppte Test-Instanz:** Der Support wollte eine Instanz eines anderen Kunden stoppen, hat allerdings versehentlich den falschen Prozess gestoppt.
- **Backups:** Wenn der Prozess ausgeführt wurde, waren sie korrekt, aber nur geschätzt eines von drei Backups wurde fertig gestellt.
- **Security:** Einem Mitarbeiter wurde per gezieltem Phishing auf sein Facebook-Konto sein Firmen-Passwort abgenommen. Damit konnte von überall auf jede Datenbank als Administrator zugegriffen werden.
- **Leere Backups:** Nach einer Änderung wurden die Backups korrekt erstellt, allerdings von einer falschen Datenbank wodurch die Sicherungsdatei keine Daten enthielt.
- **Billing:** Rechnungen zum fünffachen Preis der verwendeten Services, die auch sofort abgebucht wurden.
- **Support:** Meistens funktionierte der Support gut, aber in manchen unvorhersehbaren Fällen dauerte es tagelang bis wir eine Antwort bekamen.

Für den Start war die Cloud-Lösung richtig. Nachdem wir aber mittlerweile über jahrelange Erfahrung mit MongoDB verfügen, betreiben wir es jetzt selbst (in der Cloud). Und können hoffentlich die Fehler unseres ehemaligen Service-Providers vermeiden.

Der Sprecher: Philipp Krenn



Philipp Krenn betreut Datenbanken, Deployments und die gesamte Cloud-Infrastruktur beim B2B-Startup ecosio. Wenn er nicht gerade mit MongoDB, MySQL, Jenkins oder AWS kämpft, hält er Schulungen oder organisiert ein ViennaDB oder Papers We Love Vienna Meetup.

Kontakt: pk@xeraa.net

Donnerstag, 17. Dezember 2015, 09:30 - 10:15 Uhr

Martin Klier:

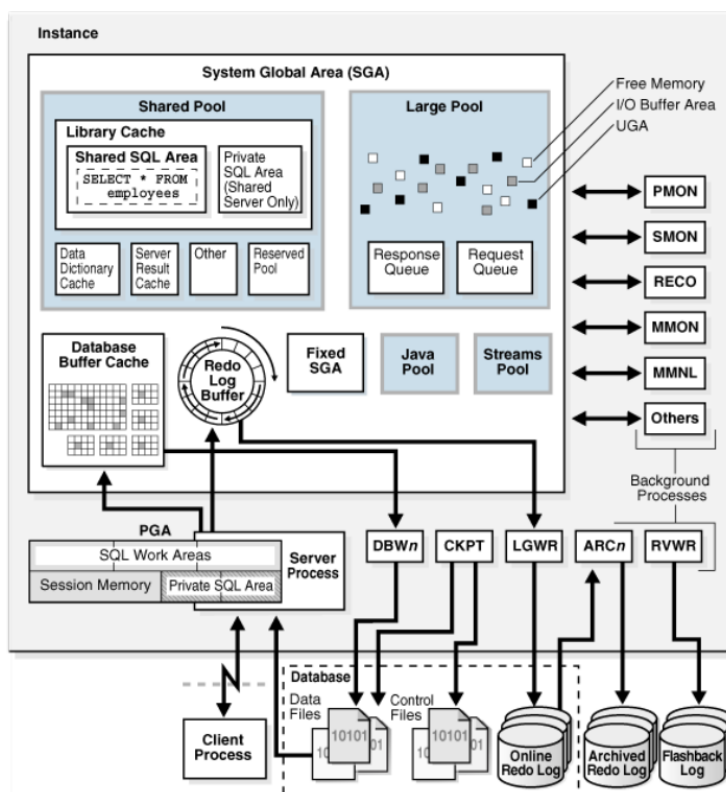
Oracle InMemory Column Store

Einleitung

Die neue InMemory Funktionalität der Oracle Database ist derzeit ein wichtiges Thema und wird vielerorts erörtert. Dieses Dokument möchte vorrangig in Form eines Überblicks das Konzept erläutern, und Grundwissen schaffen. Daher ist die Konfiguration und die beim Hersteller dokumentierte Syntax hier nicht behandelt.

Durch begrenzten Raum und die anvisierte Zielgruppe werden einige Vorgänge deutlich vereinfacht dargestellt. Detailfragen, Kritik und Verbesserungsvorschläge sind jederzeit willkommen: Im persönlichen Gespräch, oder unter martin.klier@performing-db.com.

Basiswissen - Aufbau des RDBMS



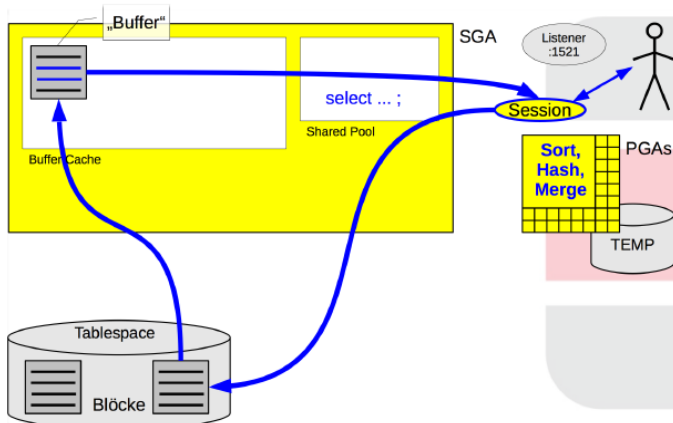
Das Relationale Datenbank-Management-System (RDBMS) von Oracle besteht im Wesentlichen aus zwei Teilen: Während alle Komponenten im RAM, wie Speicherbereiche und Prozesse, zur "Instanz" zählen, bilden die Datafiles, Online Redo Logs und Control Files die eigentliche "Datenbank". Diese Unterscheidung ist sinnvoll: Denn die Datenbank stellt das Ergebnis aller Bemühungen des RDBMS dar. Die Instanz ist "nur" das Werkzeug, das zum Ziel führt.

Diese Graphik aus der Oracle-Dokumentation 12cR1 zeigt das Zusammenwirken von Instanz

und Datenbank.

Arbeiten mit dem Massenspeicher

Zur Wiederholung sei kurz und vereinfacht die Handhabung von Daten im Oracle RDBMS dargestellt.

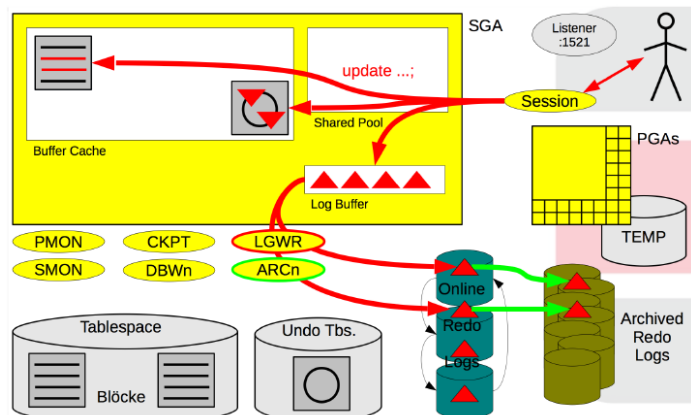


Lesen

Reguläre Lesevorgänge erfolgen durch den jeweiligen Serverprozess ("Session") und bringen relevante Blöcke im Ganzen in den Buffer Cache. Ab hier spricht man nicht mehr vom Block, sondern vom Buffer. Von dort aus grenzt der Server auf Teilmengen/Rows ein.

Schreiben

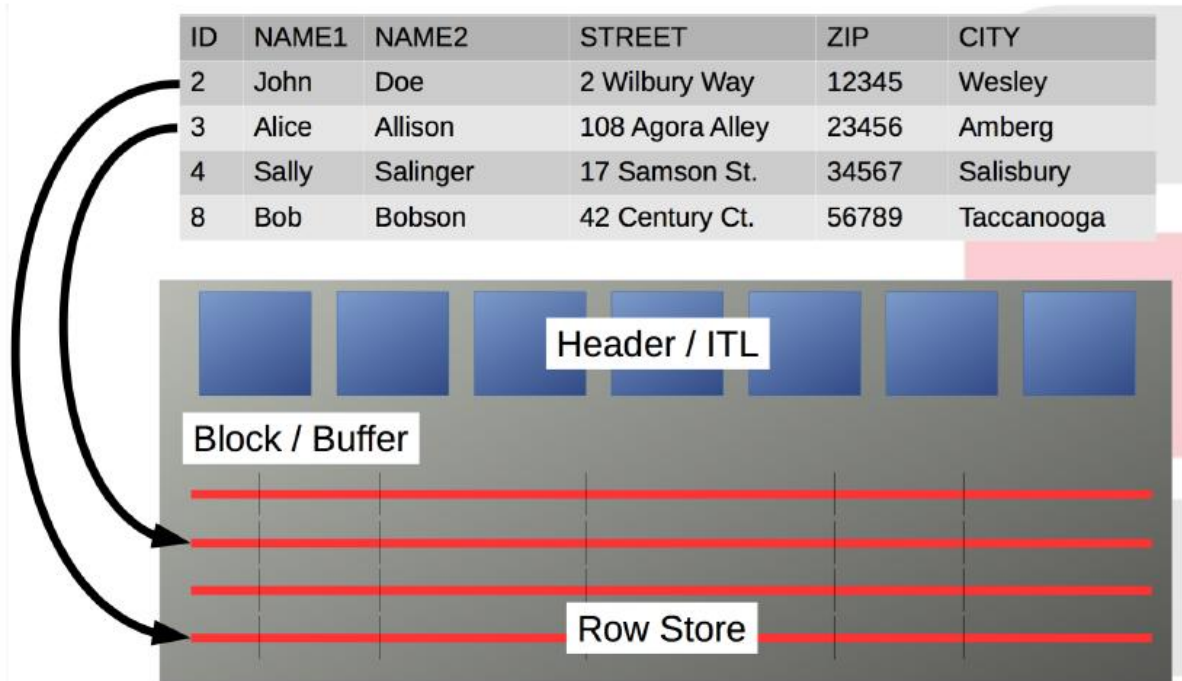
- Verändernde Transaktionen schreiben zunächst auf den Buffer im Cache, erzeugen dort ebenso Undo-Informationen in einem weiteren Buffer.
- Für beide Vorgänge werden Redo-Daten im Redo Log Buffer eingetragen.
- Alle Buffer des Cache werden nach und nach durch den Komplex aus den Checkpoint- (CKPT) und Database Writer (DBWn) Prozesse in die Tablespace Datafiles synchronisiert.
- Der Log Writer Prozess (LGWR) schreibt die Inhalte des Redo Log Buffer in die Online Redo Logs. Sind alle Redo Logs voll und alle referenzierten Buffer synchronisiert, so werden die ältesten Einträge im Redo Log wieder überschrieben.
- Die aktuelle System Change Number wird im Control File aktualisiert, ebenso ggf. neue Datafile-Größen.
- Optional, aber sehr üblich für "Archivelog-Mode": Die Archiver-Prozesse (ARCn) sorgen für die Duplizierung aller vollständig gefüllten Online Redo Logs in die sogenannten Archived Redo Logs (kurz: Archivelogs), die für verschiedene Recovery Szenarien benötigt werden, z.B. Vollständiges Recovery.
- Optional für "Flashback Database", nur in Enterprise Edition: Der Recovery Writer Prozess (RVWR) kopiert angefallene Undo-Informationen in die Flashback Logs. Diese finden Verwendung um z.B. die ganze Datenbank in der Zeit zurückrollen zu können, ohne ein Backup einzuspielen.



Zeilenorientierte Datenhaltung

Herkömmlicherweise ist die relationale Oracle Datenbank für die Speicherung von Nutzdaten in Zeilen ("rows") angelegt. Dies ist nicht nur logisch der Fall, auch auf Blockebene werden

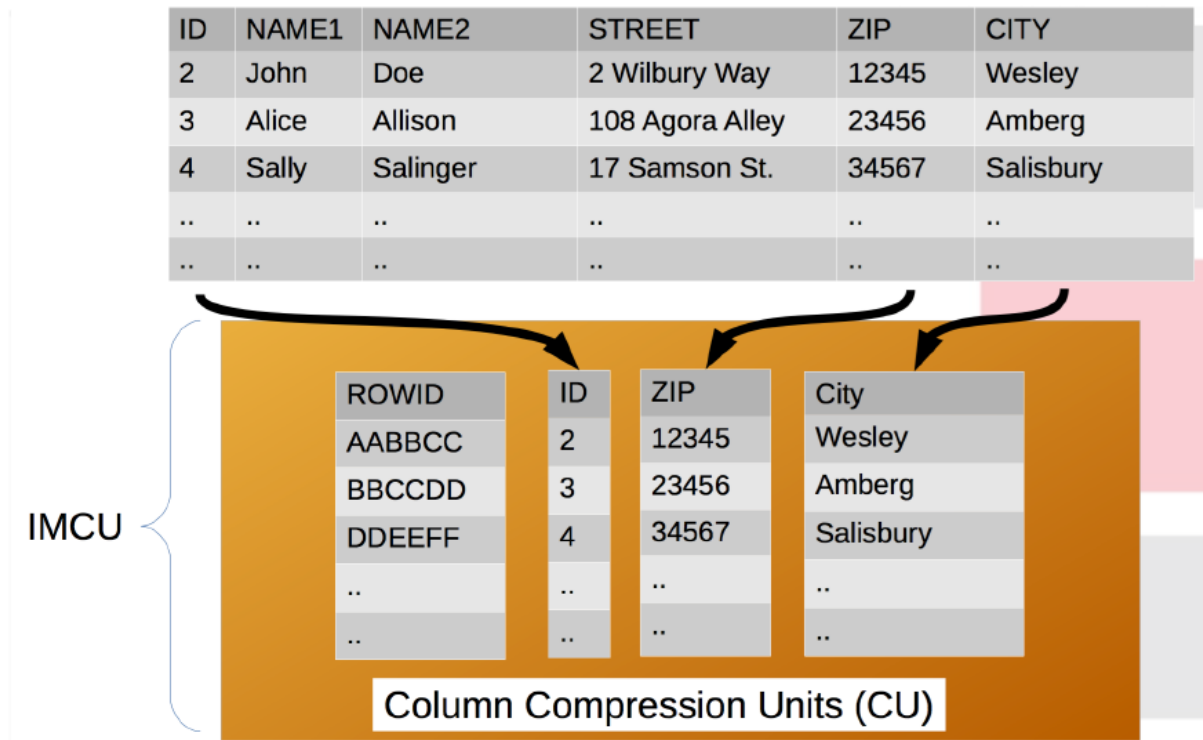
die entsprechenden Strukturen zeilenweise im "Row Store" angelegt. Ebenfalls danach ausgelegt ist die Behandlung von Schreibsperrern: Im Block Header wird mit der Interested Transaction List (ITL) zeilenweise verwaltet, welche Operation bzw. Transaktion gerade welche Zeile mit welcher Sperre belegt.



Bedingt durch dieses Konzept liest das System stets alle Spalten einer Zeile als logische, integrale Einheit. Damit muss es auch ggf. für einen speziellen Fall uninteressante Spaltenwerte entweder mitlesen und später ausfiltern, oder durch aufwändiges Überspringen auslassen. Das ist zwar grundsätzlich kein Problem, erzeugt jedoch bei im Wesentlichen auf eine Spalte konzentrierten Vorgängen (z. B. Aggregatfunktionen oder Filter) und entsprechend hoher Datenmenge einen signifikanten Overhead.

Spaltenorientierte Datenhaltung

Den im letzten Abschnitt genannten Overhead versucht der Column Store der Oracle InMemory Komponente zu vermeiden. Da das Konzept - wie es der Name andeutet - darauf ausgelegt ist, die Daten im knappen und teuren Arbeitsspeicher zu halten, ist auch eine mehrstufige Kompression vorgesehen.



Die Oberstruktur ist die sogenannte "In Memory Compression Unit" (IMCU), die man mit einem Buffer des herkömmlichen Buffer Cache vergleichen könnte. Allerdings gibt es darin keinen Row Store, sondern "Column Compression Units" (CU). Dort werden Teilmengen der Spalten unter Referenz auf ihre jeweilige Row ID gehalten. Vielerlei Metadaten pro CU (z.B. Beginn- und Endwert) eröffnen den Weg zu interessanten, später noch beschriebenen Optimierungen beim Zugriff.

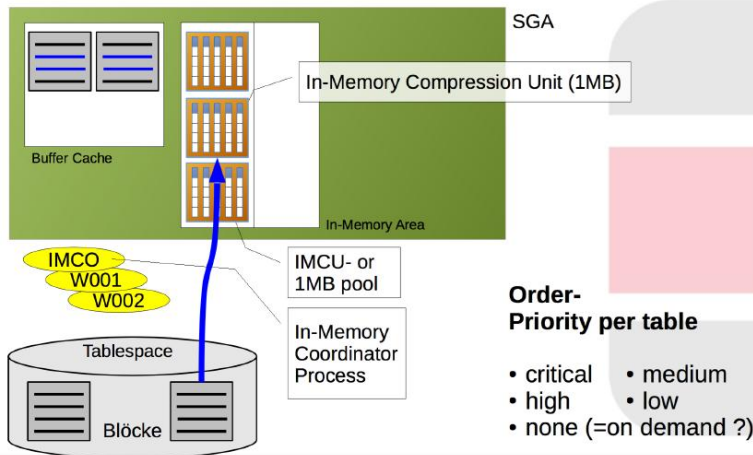
InMemory Architektur

Die Einbettung in die herkömmliche Architektur der Datenbankinstanz geschieht durch die Schaffung einer "In-Memory Area" innerhalb der SGA. In deren "1-MB-Pool" werden die zuvor erläuterten IMCUs gehalten und verwaltet.

Population

Der "In-Memory Coordinator Process" (IMCO) sorgt für die sogenannte "Population" von Tabellen in den Column Store. Dafür lassen sich zwischen den Tabellen gemeinsam oder einzeln Prioritäten von "critical" über "high", "medium" bis hin zu "low" vergeben. Auch "none" ist möglich, was eine Population bei der ersten Nutzung konfiguriert.

Die Population umfasst auch die Kompression der Daten in die IMCUs, was kurz nach dem Start der DB Instanz eine erhebliche CPU-Last verursachen kann.

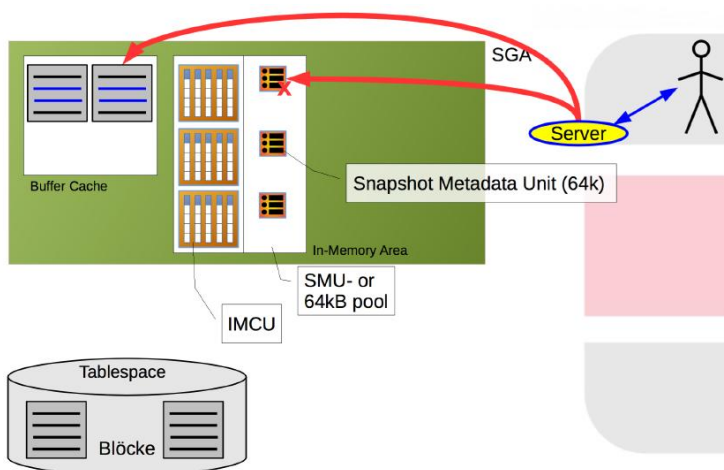


Anzumerken ist noch, dass es keine Form eines Least Recently Used (LRU) Algorithmus gibt: Ist die In-Memory Area aufgebraucht, ist weiteren Objekten (bzw. genauer: IMCUs) die Aufnahme verwehrt. Hier ähnelt der Column Store bei weitem mehr einem Tablespace als einem Cache, der er auch nicht ist.

Diese Charakteristik erfordert allerdings eine darauf abgestimmte Planung durch den DBA: Die In-Memory Area muss auf die (ggf. wachsende?) Datenmenge der populierten Tabellen abgestimmt werden. An dieser Stelle ist entsprechend langfristige Planung und sicher auch passendes Monitoring dringend angeraten.

Schreiben

Beim Schreibvorgang auf Daten, die im In-Memory Column Store liegen, stellt sich die Problematik, dass diese Inhalte konsistent zum Buffer Cache bleiben müssen. Andernfalls würde man die später beschriebene Optimizer-Funktionalität für den Zugriffspfad immer noch zusätzlich mit Information über die Aktualität der Daten überlagern müssen.

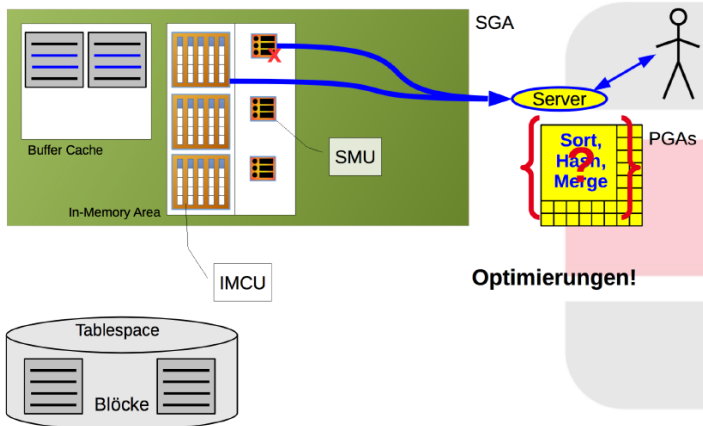


Allerdings kann es keine Lösung sein, die vorkomprimierten IMCUs bei jedem Schreibvorgang aufzubrechen und neu zu konstruieren. Um das Problem zu vermeiden, führt Oracle innerhalb der In-memory Area einen 64kB-Pool ein, in dem sogenannte Snapshot Metadata Units (SMU) abgelegt werden. Zu jeder IMCU gibt es genau einen

solchen Bereich, der Differenzinformationen zwischen IMCU-Inhalt und der zugehörigen Datenmenge im Buffer Cache speichert. Das erfordert zusätzliche Schreibvorgänge auf dem Shared Memory, hat aber verglichen mit der Kompression von der Menge Daten für ein 1MB großes IMCU einen relativ kleinen Footprint.

Lesen

Beim Lesen von zwischenzeitlich veränderten Daten aus der IMCU entstehen durch Verschmelzung mit Information aus der SMU die korrekten Rückgabewerte.



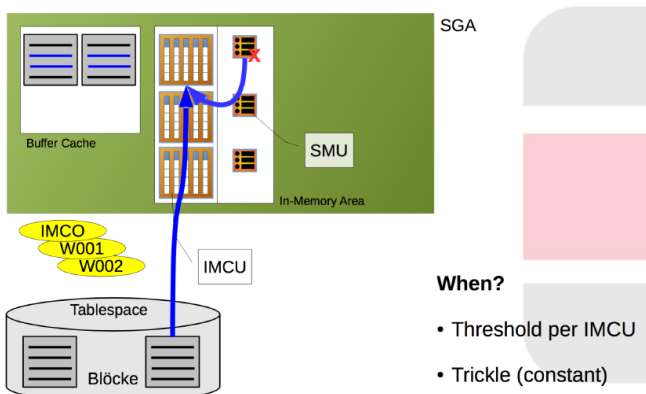
Auch hier entsteht Overhead, der jedoch auch mit der oben erwähnten, nicht praktikablen Lösung der (Neu-)Erstellung von IMCUs innerhalb einer Transaktion relativiert werden muss.

Allerdings ist der konkret an der Stelle auftretende Zeitverlust durchaus zu spüren - in entsprechend angelegten Praxisversuchen konnte ich im

Vergleich zu einem perfekt ausbalancierten In-Memory Store durchaus eine Verzehnfachung der Antwortzeiten "erreichen".

Re-Population

Das Konzept, manipulierte Daten über eine Art inkrementellen Speicher aufzufangen, ist nur endlich skalierbar. An einem gewissen Punkt muss eine Rückführung stattfinden. Diese nennt sich Re-Population, und wird von den Workerprozessen des IMCO ausgeführt. Sie wird entweder durch Schwellenwert nach prozentualer Veränderung pro IMCU ausgelöst, das ist die "harte" Variante.



Als "Soft"-Lösung wird in Ruhe das "Trickle"-Verfahren, zu Deutsch "Tröpfeln", angewendet. Der unbeschäftigte IMCO Prozess beginnt, auch IMCUs unterhalb ihres Schwellenwertes zu populieren, um für ggf. später auftretende Lastspitzen optimal gerüstet zu sein.

Mit der Re-Population nähert sich das System wieder an den perfekten Zustand des In-Memory Column Store an.

Transparente Nutzung

Der Cost Based Optimizer nutzt den In-Memory Column Store automatisch, wenn dieser einen effektiven Zugriffspfad auf die Daten darstellt. Damit ist es nicht notwendig, auf die In-Memory Funktion abgestimmtes SQL zu verwenden - weder hinsichtlich Syntax, noch hinsichtlich Semantik.

```
SQL_ID a202cw0jn1trn, child number 0
select /** noparallel */ object_id, count(*) x from imtable group by
object_id
Plan hash value: 186096447
```

Id	Operation	Name	E-Rows	E-Bytes	E-Temp	Cost (%CPU)	E-Time
0	SELECT STATEMENT					7741 (100)	
1	HASH GROUP BY		90895	443K	66M	7741 (4)	00:00:01
2	TABLE ACCESS INMEMORY FULL	IMTABLE	5810K	27M		1174 (5)	00:00:01


```
SQL_ID 9n0h8a6c4jn5d, child number 0
select /** noparallel */ object_id, count(*) x from imtable where
object_id=82345 group by object_id
Plan hash value: 1486712282
```

Id	Operation	Name	E-Rows	E-Bytes	Cost (%CPU)	E-Time	
0	SELECT STATEMENT					3 (100)	
1	SORT GROUP BY NOSORT		64	320	3	(0)	00:00:01
* 2	INDEX RANGE SCAN	I_OID_1	64	320	3	(0)	00:00:01

Ein kleines Beispiel zeigt schon mit wenig Aufwand, dass das System fallweise automatisch auf einen INMEMORY Zugriff umstellt: Die COUNT Funktion auf die große Tabelle (oben) ist ein Paradebeispiel, da die Anzahl der Datensätze pro IMCU schon ohne Dekompression sichtbar

ist, die Daten also als In-Memory Index vorliegen. Dagegen wäre es in höchstem Maße ineffektiv, für den Zugriff auf eine einzige Row eine 1MB große IMCU zu entpacken - der Zugriff über den herkömmlichen B*Tree Index auf die Spalte OBJECT_ID zur Prüfung des Prädikates ist hier bei weitem kostengünstiger.

Zugriffsoptimierungen

Oracle hat in das Lesen aus dem In-memory Column Store einige Verhaltensweisen und Optimierungen eingebaut, die besondere Erwähnung verdienen, und die dieses Feature bei entsprechendem Anwendungsfall attraktiv machen. Ich möchte nur kurz auf die wichtigsten Punkte eingehen:

- Vector Processing für SIMD-fähige Prozessoren, z.B. Intel AVX Extension mit 256bit Register (verändert mehrere vorgeladene Daten in einem Verarbeitungsschritt)
- Operation Pushdown In-Memory Scan (ähnlich Exadata Smart Scan, Prädikatprüfung auf Speicherebene)
 - Prädikat-Prüfung im Column Store (min/max von CUs von außen sichtbar)
 - Aggregation im Scan (diverse Aggregate des Inhalts einer CU von außen abgreifbar)
 - Bloom Filter im IMCU Scan (Kombination der beiden letzten Punkte für effektive Joins)
- In-Memory Index (Auslassen unnötiger IMCUs)
- Predicate Optimization (Weglassen der Prädikatprüfung bei ohnehin vollständig benötigter CU)

Abwägung

Vorteile

- Schneller bei spaltenbasierten Auswertungen
- Optimizer nutzt „Shortcuts“ der Architektur

- Voll transparent, ideal für Mischbetrieb auf selbem Datenbestand

Nachteile

- Ressourcenbedarf
 - RAM
 - CPU für Compression
- Inhalt muss gepflegt / verwaltet werden
- Eingeschränkt deterministisch: Nutzen stark abhängig von
 - Workload
 - Inhalt
 - Zustand
- Skalierung im RAC ohne Ausfallsicherheit (nur auf Engineered Systems verfügbar)

Schlussfolgerung

InMemory ist cool. Aber Cool reicht nicht. Der UseCase muss passen.

Abschluss

Das Thema wie auch dieses Dokument kennen kein Ende, nur die Aufforderung an alle Leser, sich weiterzubilden, und die "losen Enden" der Geschichte aufzugreifen, als Ansatzpunkt für das nächste Projekt, oder die eigene Fortbildung aus Interesse.

Für jeden, der mehr über das Oracle RDBMS erfahren möchte, sei dessen vorzügliche Onlinedokumentation empfohlen, für die 12c derzeit hier zu finden:

<http://docs.oracle.com/database/121/index.htm>

Der Sprecher: Martin Klier



Martin Klier ist Oracle ACE und Senior Datenbank-administrator mit Schwerpunkt auf Oracle Performance und Hochverfügbarkeit. Sein Hintergrund im Linux/Unix-Umfeld legte den Grundstein für zahlreiche erfolgreiche Projekte mit Oracle RAC und Dataguard. Dabei entdeckte er seine Passion für Leistungsanalyse und -optimierung, die nun seit zehn Jahren sein Arbeitsleben bestimmt. Skeptische Bedarfsanalysen, gründliche Architekturplanung, ordentliches Systemsizing, kritisches Benchmarken und eine auf jahrelanger Erfahrung basierende Systemanalyse sind Martins Rezept, um Ausfälle in Produktionsumgebungen zu verhüten.

Seit Jahren ist Martin als Sprecher zu Themen aus dem Bereich "Oracle Core und Performance" auf diversen nationalen und internationalen Veranstaltungen zu sehen,

unter anderem COLLABORATE (IOUG), Oracle Open World oder der DOAG Konferenz.

Martin Klier ist einer der Geschäftsführenden Gesellschafter der Performing Databases GmbH und lebt mit seiner Familie in Nordbayern.

Kontakt: martin.klier@performing-db.com

Donnerstag, 17. Dezember 2015, 10:30 - 11:15 Uhr

Marek Adar:

Oracle 12c Multitenant: Architektur und Anwendung

Übersicht

Oracle Database 12c bringt eine grundlegende Architektur-Änderung mit sich: Die Multitenant Database. Marek Adar zeigt Ihnen in dieser Session die wichtigen Features und deren Verwendung.

Die Themen dieser Session sind:

- Multitenant: Architektur
- Containerdatenbank und Pluggable Datenbanken
- Architektur einer Containerdatenbank
- Container in einer Containerdatenbank
- Erstellung einer Multitenant Datenbank mit dem DBCA
- Anmelden an einer CDB und PDB
- Informationen über Dictionary-Views abfragen
- Starten und Stoppen von Pluggable Datenbank
- Standard permanent und temporären Tablespace ändern
- Benutzerverwaltung in Multitenant
- Erstellung von allgemeinen Benutzern (Common Users)
- Erstellung von lokalen Benutzern (Local Users)
- Erstellung von allgemeinen Rollen (Common Roles)
- Erstellung von lokalen Rollen (Local Roles)
- Erstellen von Pluggable Datenbanken
- Erstellen einer Pluggable Datenbank von der Seed Datenbank
- Erstellen einer Pluggable Database aus einer vorhandenen Pluggable Database
- An- und Abhängen von Pluggable Datenbanken
- Abhängen einer Pluggable Datenbank
- Anhängen einer Pluggable Database

Nach dieser Session kennen Sie die Architektur der Oracle Multitenant Database in 12c. Sie verstehen die Features und können Sie zuverlässig bewerten und einsetzen.

Der Sprecher: Marek Adar



Diplom-Physik-Ingenieur Marek Adar ist Oracle Certified Professional und arbeitet seit vielen Jahren als freiberuflicher Oracle Referent und Consultant. Seine Kernbereiche im Oracle-Umfeld liegen in der SQL- und PL/SQL-Programmierung, Datenbankadministration, Performance Tuning, Backup und Recovery, Grid Control, ASM, RAC und Data Guard.

Er ist Autor und Co-Autor mehrerer Oracle-Bücher, darunter "Der Oracle DBA" (zweite überarbeitete Auflage, Hanser-Verlag 2015).

Kontakt: marek.adar@de.luxaviation.com

Donnerstag, 17. Dezember 2015, 11:30 - 12:15 Uhr

Marek Adar:

Oracle 12c: Oracle Recovery Manager New Features

Übersicht

Marek Adar zeigt Ihnen die New Features rund um Oracle Backup und Recovery mit den Oracle Release 12c.

- Backup & Recovery in Oracle 12c
- Fein granuliertes Recovery für Tabellen und Partitionen
- Container und Pluggable Databases Backup und Recovery
- Erweiterungen der Cloning-Fähigkeiten
- Schnelleres Recovery über das Netzwerk
- Multi-Section Support für inkrementelle und Image-Copies
- SQL Interface in RMAN

Nach dieser Session kennen Sie die neuen Features zu Oracle 12c Backup und Recovery. Sie können die Features bewerten und zuverlässig einsetzen.

Der Sprecher: Marek Adar



Diplom-Physik-Ingenieur Marek Adar ist Oracle Certified Professional und arbeitet seit vielen Jahren als freiberuflicher Oracle Referent und Consultant. Seine Kernbereiche im Oracle-Umfeld liegen in der SQL- und PL/SQL-Programmierung, Datenbankadministration, Performance Tuning, Backup und Recovery, Grid Control, ASM, RAC und Data Guard.

Er ist Autor und Co-Autor mehrerer Oracle-Bücher, darunter "Der Oracle DBA" (zweite überarbeitete Auflage, Hanser-Verlag 2015).

Kontakt: marek.adar@de.luxaviation.com

Donnerstag, 17. Dezember 2015, 13:30 - 14:15 Uhr

Udo Fohrmann:

Adaptive Dynamic Sampling in Oracle 12c

Dynamic Statistics vs. (Adaptive) Dynamic Sampling

- Dynamic statistics and dynamic sampling are synonyms
- In 12c a new implementation called adaptive dynamic sampling (ADS) is available
- The former implementation, which wasn't improved in 12c, is called old style dynamic sampling
- This presentation focuses on adaptive dynamic sampling

Controlling Adaptive Dynamic Sampling

- It's enabled when `OPTIMIZER_DYNAMIC_SAMPLING` is set to 11
- Level 11 is available as of 11.2.0.4
 - It's a 12c feature that was back ported
- Even though it's considered an adaptive query optimization technique, it's not controlled by `OPTIMIZER_ADAPTIVE_FEATURES`

When does Adaptive Dynamic Sampling Take Place?

- It isn't limited (as the old style dynamic sampling) to single-table cardinality estimations involving
 - missing statistics,
 - expressions, or
 - predicates referencing several columns of the same table
- The query optimizer can use it for most SQL statements
- It's used for the estimation of single-table cardinalities as well
- as join cardinalities and group-by cardinalities

Time Limit

- The dynamic sampling phase can't exceed a given amount of time called time limit
 - There's no specific number of blocks to sample like in old style dynamic sampling
- The time limit depends on whether the SQL statement is present in the cursor cache or AWR
 - For AWR query, only 12.1.0.2 considers `CONTROL_MANAGEMENT_PACK_ACCESS`
- If it's not present, a time limit of 10 seconds is used
- If it's present, the time limit depends on the CPU consumption and the number of executions

Sampling Queries with old style vs adaptive dynamic sampling

- With old style dynamic sampling the optimizer runs a single sampling query
 - The sampling percentage is determined by the dynamic sampling level
- With adaptive dynamic sampling the optimizer runs several/many sampling query
 - The number depends on the complexity of the query and the number of available indexes
 - Even for queries accessing few tables, it's not unusual at all to see 10-30 sampling query for a single parse
 - The number also depends on the database version
- Newer releases usually executes more sampling queries

Basic Strategy

- General methodology with ODS=11:
 - Execute the SELECT statement with sample block (x,8) seed(1) for reducing number of scanned rows
 - Depending of estimation accuracy (difference E-Rows to A-Rows) generate more statements (*)
 - Use index for selectivity estimation for every single column in WHERE clause
- Consequences:
 - Depending on table size, one or several «from table sample block» will be executed
 - If estimation is always wrong, for every index defined on column in WHERE clause, a query will be executed 1 or 2 times using «from table sample block» or any visible index

Summary

- Adaptive dynamic sampling can provide very useful information to the optimizer
- More (in some situations, much more!) parse time is required
- Three main differences to «old style dynamic sampling» can be notified:
 - Time Limit:
- The dynamic sampling phase can't exceed a given amount of time called time limit
 - Dynamic Sampling takes not only place for single-table cardinalities it's activated also for join cardinalities and group-by cardinalities
 - SQL plan directives can instruct the database to use dynamic sampling
- The feature needs to be carefully tested!

Der Sprecher: Udo Fohrmann



Nach dem Studium der Mathematik arbeitete Udo Fohrmann zunächst in der Anwendungsprogrammierung auf Basis von Oracle-Datenbanken meist im Unix-Umfeld und bezog dann Administrationsaufgaben von Oracle-Datenbanken in verschiedenen Branchen.

Nach Datenmodellierung sind dann Performance-Tuning und Datenbankadministrationsaufgaben zu den Schwerpunkten geworden.

Seit 2008 ist er bei Trivadis Oracle DBA mit Schwerpunkten Performance-Tuning, Workshops und Seminare mit den Themen Performance-Tuning, Datenmodellierung und SQL-Tuning wurden in den letzten 4-5 Jahren regelmäßig doziert.

Kontakt: udo.fohrmann@trivadis.com

Donnerstag, 17. Dezember 2015, 14:30 - 15:15 Uhr

Jérôme Witt:

Oracle RDBMS – Grid Infrastructure 12c: failover cluster for dummies

Introduction

Cold failover clusters are a very popular high availability solution where the database is only available one node at a time. Several vendors offer solutions with more or less all the same features. However none of them provide an “out-of-the box” solution to install, deploy and operate.

Even, the Oracle Grid Infrastructure suite does not provide failure protected single instance database by default. In fact Oracle provides several types of database:

- Real Application Cluster (RAC)
- RAC One Node
- Single

Grid Infrastructure supplied database types

This paragraph introduces shortly the Oracle supplied database type and their limits.

Real Application Cluster (RAC)	
Short description	This type of database is out of scope because it breaks the principle of a “failover cluster”.
Limits	Extra cost (Oracle EE and SE2 option)
RAC One Node	
Short description	Single instance database type with RAC capabilities. One of the main advantages of this database type is that the database relocation is performed without service disruption
Limits	Extra cost and only available as an option for Oracle EE
Single	
Short description	Restricted to Oracle Restart installations.
Limits	No relocate/failover capabilities

Last one but not least, the integration of those database types in the cluster are of course out of the box, follows a short example how to add a RAC One Node database to the cluster:

```
# srvctl add database -db $ORACLE_SID
\ -oraclehome $ORACLE_HOME -dbtype RACONENODE
```

Alternatives

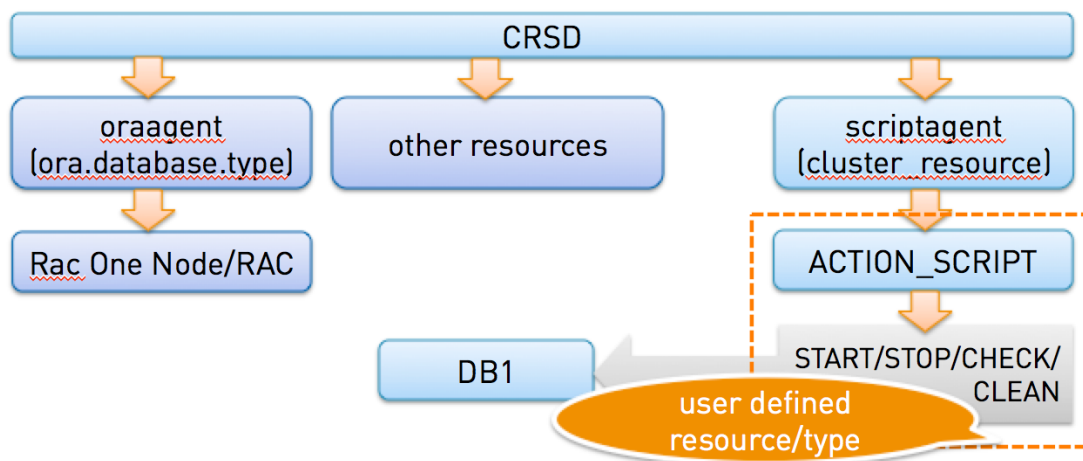
The RAC One database type fits to the use case of an Active/Passive cluster. However, as mentioned it generates additional costs and is reserved to the Oracle Enterprise Edition database.

So what? Oracle Grid Infrastructure allows to start, stop, monitor, restart and relocate applications through “user defined” resources. Those user-defined resources are managed the same way with almost the same parameters as supplied resources. Except, that the whole intelligence behind the management commands (start, stop, restart) are self-engineered.

The Oracle Clusterware (component of the Grid Infrastructure software suite) executes in background the self-engineered action script.

Grid Infrastructure action script integration

The following big picture presents the action script integration with Oracle Grid Infrastructure:



The Cluster Ready Service Daemon (CRSD) is the upper stack, which interacts with the user. The CRSD interacts directly with different kind of agent, which are interacting with the resources like a RAC One Node database.

We do manage the "user-defined" resource exactly the same way, by interacting with the CRSD. The CRSD itself manage then the resource using the action script executed by the scriptagent.

The scriptagent executes the action script with simple arguments; this is protocolled in the scriptagent’s logfile as shown below:

```

2015-09-09 21:28:21.302: [DB1.db] [1749710592] {1:62596:2} [check]
Executing action script:
/u01/app/oracle/local/dmk/bin/db_crs.ksh[check]
  
```

Database Licensing “The 10-days rule”

From a licensing point of view the RAC One node and action scripts are both threatened as “Failover data recovery” methods and documented accordingly.

Reference “Licensing Data Recovery Environments”

Rule:

- Multiple nodes have access to one single storage (SAN)
- Includes the right to run the licensed program on an unlicensed spare computer for up to a total of 10 separate days
- Downtime for maintenance purposes counts towards!
- If failover period exceeds 10 days then the failover node must be licensed

Conclusion

The answer is always “it depends”.

Oracle Grid Infrastructure is a software suite composed of some highly available feature for storage (ie ACFS) and network (Virtual IPs) capabilities. The knowledge gained to manage a single database instance as a cluster resource using action scripts can be easily moved to third party software's. Thus enabling application availability (or whatever else).

However, if you are an Oracle EE customer and you do not have to worry about the Oracle licensing; then, the Oracle RAC One Node solution might be appropriate, as the database relocation will be performed without service disruption.

In case of a failure, the “failover” will for both solutions cause a service disruption.

Der Sprecher: Jérôme Witt



Jérôme startete seine Consulting-Karriere vor 5 Jahren. Fachlich ist er in den Bereichen Datenbank, Datenbankinfrastrukturen, Engineering und Optimierung spezialisiert. Er ist hoch-qualifiziert in Oracle Hochverfügbarkeit, Backup & Recovery und Datenbankoptimierungs-Technologien. Seine Kompetenzen erfassen ebenfalls den Open Source, das Unix/Linux-Umfeld, sowie Perl, Shell, Windows PowerShell-Programmierung und Automatisierungs-Tools wie UC4 und BMC Control-M.

Kontakt: jerome.witt@dbi-services.com

Donnerstag, 17. Dezember 2015, 15:30 - 16:15 Uhr

Dierk Lenz:

Oracle 12c Upgrade – Erste Erfahrungen

Wo stehen wir aktuell?

Die Version 12c der Oracle Database ist aus diversen Gründen außergewöhnlich. Zunächst wurde Release 12.1 erst veröffentlicht, als sich Release 11.2 bereits mit großen Schritten dem Extended Support näherte. Das bedeutete zunächst, dass man die übliche Strategie, auf das ".2er Release" zu warten, überdenken musste. Schließlich möchte man ja – in den meisten Fällen – ein von Oracle supportetes Release betreiben.

Hinzu kommt seit Juli 2014 die Verwirrung um die Standard Edition. Über ein Jahr lang keinerlei Information, wann es das Patchset 12.1.0.2 für die Standard Edition geben wird, dann am 01.09.2015 eine neue "Standard Edition 2" mit neuen Lizenzen, neuen Preisen und technischen Einschränkungen.

In dieser Situation ist es mehr als verständlich, dass viele Standard Edition-Kunden in einer Art Schockstarre verweilt haben, d. h. hier gibt es noch nicht allzu viele Migrationen. Im Bereich Enterprise Edition sieht es nur wenig besser aus; aber immerhin gibt es hier einige laufende Installationen.

Was sind die zentralen *New Features*?

Stellt man diese Frage bei Oracle, dann sind natürlich die Multitenant- und In-Memory-Optionen die wegweisenden neuen Fähigkeiten. Über produktive Datenbanken, die diese Features als zusätzliche Optionen einsetzen, kann an dieser Stelle nicht berichtet werden.

Ohne Multitenant-Option kann eine DB als Container mit genau einer Pluggable DB betrieben werden. Hierzu gab es einige hitzige Diskussionen, da Oracle die bekannte Konfiguration ohne Container-DB als deprecated gekennzeichnet hat. Allerdings mit der Bemerkung, dass dies nicht gilt, wenn man bestimmte Features nutzen möchte. In der Liste taucht dann u. a. das DB Verify Utility auf, mit dem man die Konsistenz von DB-Dateien prüfen kann.

Es gibt aber einige andere Features, über die nicht so viel geredet wird.

Längere VARCHAR2-Inhalte

Es ist möglich, mit 12c das Limit von 4000 Bytes pro VARCHAR2-Inhalt zu überschreiten. Hierzu ist es allerdings notwendig, den Systemparameter `max_string_size` von standard auf extended zu ändern. Damit sind für VARCHAR2 Längen von bis zu 32 KB möglich. Zusätzlich ist das Skript `?/rdbms/admin/utl32k.sql` auszuführen. Die Änderung ist nicht umkehrbar.

Die Implementierung basiert wiederum auf LOBs. Hiermit ist es einerseits möglich, bei UNICODE-Migrationen etwas gelassener mit großen VARCHAR2-Feldern umzugehen. Andererseits ist die Implementierung auch nicht ganz Bug-frei.

Optimizer

Neue Histogrammtypen

Eines der zentralen Probleme für den Optimizer bleibt die Abschätzung der Kardinalitäten. Gibt es hier Fehler, so kann der erzeugte Plan ernsthaft daneben gehen. Um nicht gleichverteilte Wertemengen darzustellen, nutzt Oracle Histogramme für Spalten. Hier gibt es mit 12c neue Histogrammtypen, die in einigen speziellen Fällen extrem gut weiter helfen. Die Auswahl des Histogrammtyps erfolgt automatisch; lediglich die Entscheidung, ob es ein Histogramm für eine Spalte geben sollte, kann und sollte man im Zweifel beeinflussen.

Automatische Extended Statistics

Extended Statistics wurden mit 11g eingeführt. Es geht darum, für logisch voneinander abhängige Spalten (z. B. PLZ und ORT) Statistiken über mehrere Spaltenwerte zu berechnen. Musste man in 11g noch selbst an solche Statistiken denken, so führt der Optimizer in 12c Extended Statistics zumindest in bestimmten Fällen automatisch ein. Das kann in einigen Fällen gut sein, in anderen Fällen aber auch zu Problemen führen.

Planänderung während der Laufzeit

Sollte dann doch mal die Kardinalität falsch berechnet sein, z. B. wegen nicht aktueller Statistiken, dann kann Oracle den Ausführungsplan noch ändern, wenn dieser bereits abgearbeitet wird. Es geht hier um die Überführung von Hash Joins in Nested Loop Joins und umgekehrt – allerdings mit der Einschränkung, dass dies nur für den ersten Join im Ausführungsplan gilt.

Full Database Caching Mode

Die Idee hat es ja schon früher gegeben: Einfach den Buffer Cache so groß machen, dass die gesamte Datenbank hinein passt. Dies wird nun mit einem entsprechenden Befehl unterstützt:

```
ALTER DATABASE FORCE FULL DATABASE CACHING;
```

Dies ist dann umso wirkungsvoller, wenn z.B. LOBs mit NOCACHE-Option ebenfalls im Speicher gehalten werden.

Und die neue Standard Edition 2

Mit dem ersten Patchset von 12c wurde nach einem Jahr Standard Edition-Pause, in der es die Version 12.1.0.2 ausschließlich als Enterprise Edition gab, gleich die neue Standard Edition 2 eingeführt. Konkret heißt das, dass es ab dieser Version keine Standard Edition bzw.

Standard Edition One mehr gibt. Das geht so weit, dass bestehende Lizenzen migriert werden müssen. Im Neukauf sind SE2-Lizenzen bei den Preisen für SE. Dafür gelten die Einschränkungen der SE1, z.B. ein Maximum von zwei CPU-Steckplätzen für das Mainboard.

Zusätzlich überrascht die SE2 mit der Tatsache, dass maximal 16 CPU-Threads pro Instanz beschäftigt werden.

Der Sprecher: Dierk Lenz



Dierk Lenz hat von 1983 bis 1989 an der RWTH Aachen Informatik studiert und ist Diplom-Informatiker. Von 1989 bis 1995 arbeitete er als Senior Systemberater bei der Oracle Deutschland GmbH. Zusammen mit Uwe Herrmann gründete er 1995 das Unternehmen Herrmann & Lenz Services GmbH. Sein Spezialgebiet ist die Oracle Datenbank mit den Schwerpunkten Konfiguration, Optimierung, Hochverfügbarkeit und Migration.

Dierk Lenz ist seit vielen Jahren bei der DOAG u. a. als einer der Regionalgruppenleiter NRW sowie mit Vorträgen und Schulungen bei Regionaltreffen, SIGs und der Konferenz aktiv. Er ist Mitverfasser von vier Oracle Fachbüchern.

Kontakt: dierk.lenz@hl-services.de

Donnerstag, 17. Dezember 2015, 16:30 - 17:15 Uhr

Angelika Gallwitz:

Statistische Auswertungen in Oracle mit Statspack und AWR

Einleitung

Auch in Oracle12c werden Statspack und AWR Reports herangezogen, um Performance-Engpässe zu erkennen und Maßnahmen zu ergreifen. Der ADDM liefert seinen Beitrag. Wie erzeuge und interpretiere ich einen Statistik-Report? Wie verändere ich Snap Level und Threshold? Anhand der Wait Events und Wait Classes wird eine Zuordnung zu den Problembereichen vorgenommen und konkrete Empfehlungen zu Tuning-Maßnahmen gegeben.

Statspack , AWR Überblick

Mit historischen Performance Statistiken können nachträglich Performance-Probleme analysiert werden. Oracle stellt dafür den User Perfstat mit seinem Statspack Repository und das AWR Automatic Workload Repository zur Verfügung. Für die Erzeugung des AWR wird die Option „Diagnostik und Tuning Pack“ benötigt, für Statspack Auswertungen nicht.

AWR ist mächtiger und kann auch im OEM Oracle Enterprise Manager angezeigt werden, ebenso ADDM Automatic Database Diagnostic Monitor.

In Oracle 12c gibt es maßgebliche Neuerungen in der Erzeugung der Snapshots für den AWR Report, der Performance Analyse und Diagnose durch den „Real Time ADDM“ .

In Oracle 12c wurde der ADDM weiterentwickelt. In Abhängigkeit vom Target hat der ADDM unterschiedliche Analysearten

Die Performance Analyse findet Top down statt, wobei die Symptome und Ursachen der Performance-Probleme identifiziert und redefiniert werden mit dem Ziel, der Reduktion des Durchsatzes - die DB time.

Darin enthalten ist:

- Wait Time
- CPU Time
- Active Session Time

Folgende Performance Views können für die Analyse herangezogen werden. Wobei immer nur die Daten des aktuellen Containers, in dem man sich befindet angezeigt werden.

```
SELECT *  
FROM  
V$CON_SYS_TIME_MODEL oder V$SESS_TIME_MODEL;
```

AWR und ADDM mit den dazugehörigen Komponenten

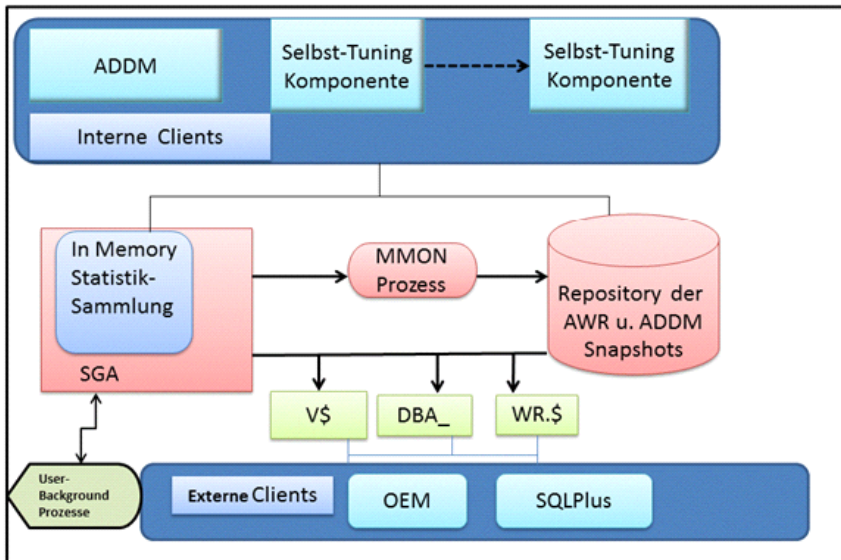


Abb1.: AWR und ADDM Architekturübersicht

Der Prozess MMON Manageability Monitor sammelt Snapshots und speichert sie im SYSAUX Tablespace des Users SYS. Die hier gesammelten Daten sind über DBA-Views ersichtlich. Die Komponente Automatic Database Diagnostic Monitor (ADDM) interpretiert diese Daten zyklisch und macht Vorschläge, die über Views (dba_advisor_...) oder über GRID Control ausgewertet werden können.

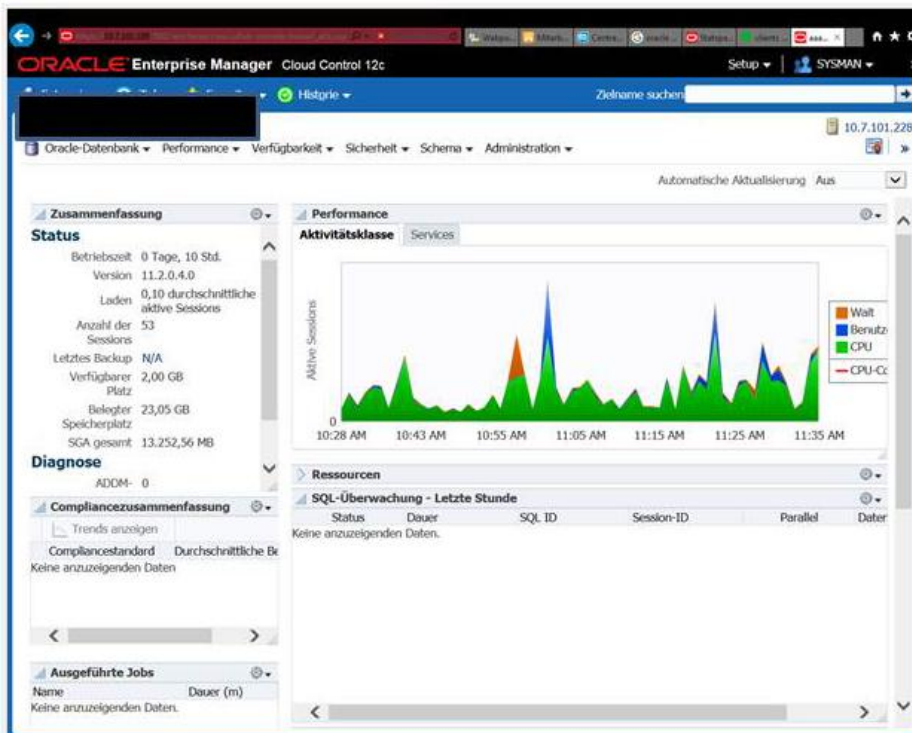


Abb2.: aktuelle Summe der Aktive Sessions mit den Unterscheidungsmerkmalen : Wait Event, Benutzer und CPU.

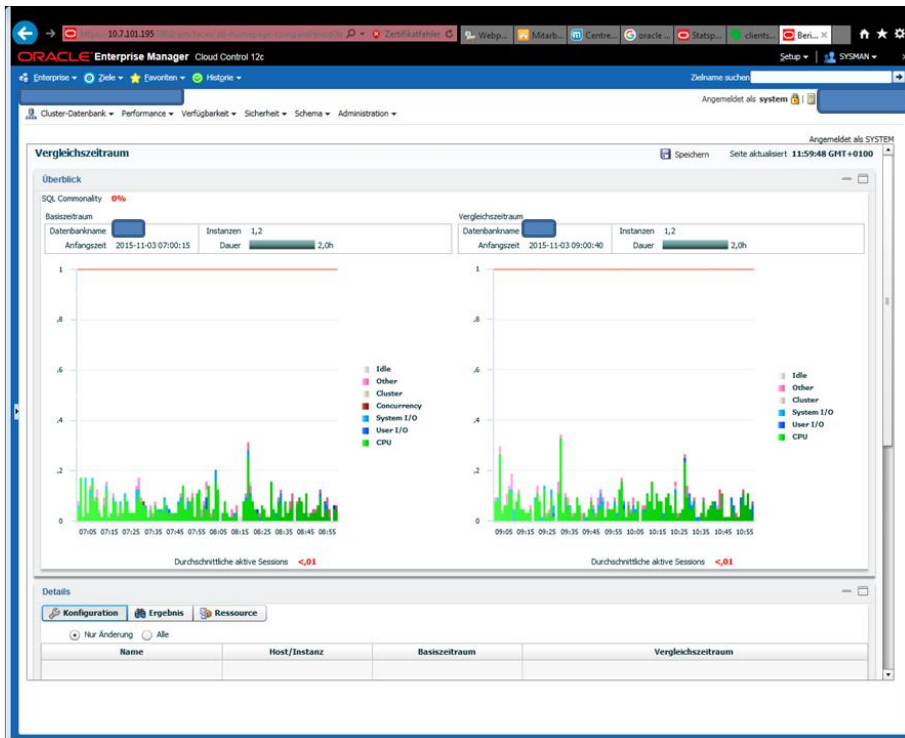


Abb3: Aufruf des Vergleichszeitraum ADDM aus dem OEM Cloud

Es wird der Database Mode verwendet . Hier ist die DB-TIME die Summe aller Instanzen. Die ADDM Analyse findet bei jedem AWR Snapshot statt, also jede Stunde Die Ergebnisse werden in „WRIS_ADDM_FDG“ in der Datenbank abgespeichert Im Enterprise Manager kann man sie ansehen, im ADDM Performance Analyse Button.

Die Sprecherin: Angelika Gallwitz



Angelika Gallwitz (Dipl-Geol.) arbeitet seit fast 30 Jahren (1987) mit der Oracle-Datenbank. Sie begann zunächst als Entwicklerin mit Oracle Forms, Reports und PL/SQL, seit 1990 als Datenbankadministratorin. Ab 1997 als Consultant tätig, 1999 freiberufliche Datenbank-Beraterin mit dem Schwerpunkt sehr großer Datenbanken und Real Applikation Cluster. Ihr Steckenpferd ist Performance Tuning mit Oracle-Mitteln sowie Backup und Recovery. Angelika Gallwitz ist regelmäßig Sprecherin auf Konferenzen und gibt Oracle-Schulungen. Sie veröffentlichte mehrere Artikel in Fachzeitschriften und ist Co-Autorin des Buches "Der Oracle DBA".

Kontakt: angelika@gallwitz-it.de

Freitag, 18. Dezember 2015, 09:30 - 10:15 Uhr

Norbert Rieger:

Performance-Optimierung von Oracle Datenbanken "nur" mit der Standard Edition

Einleitung

In dieser Präsentation sollten die Möglichkeiten zur Optimierung der Performance bei Oracle-Datenbanken mit der Standard Edition dargestellt werden.

Es sollten die Vor- und Nachteile einiger Werkzeuge vorgestellt werden mit Beispielen aus der Praxis.

Werkzeuge

- Enterprise Manager -> aber ohne Diagnostics und Tuning Pack!
- AWR -> nur mit Enterprise Edition
- utlstat/utlstat - Vorgänger vor 8.1.6 für statspack
- statspack - Periodische Snaps - Reports nach Bedarf
 - Voraussetzungen
 - Installation
 - Statistiken sammeln
 - Reports erstellen

Hilfsmittel

- Aktivierung von Traces
 - Instanz-Ebene
 - Session-Ebene
 - SQL-Plus - autotrace
- Systemstatistiken
 - Voraussetzungen
 - Erstellen
 - Auswerten
 - Dictionary-Views
- Waits
 - Wait-Events
 - Wait-Klassen
- Optimizer - CBO - Ausführungsplan erstellen
 - SQLPLUS
 - Package

Komponenten

- Memory
 - SGA

- PGA
- Tablespaces
 - Undo
 - Temp
 - Data
- Indizes
 - Dictionary Views
 - Shrink versus rebuild

Automatisierung - Skripte

- Indizes - rebuild
- Statistiken - Neuaufbau
- Tablespaces – Shrink - Coalesce
- Inaktive Objekte – compile

Report mit Statspack - Beispiel

- Report erstellen
- Aufbau des Reports
- Report auswerten
- Wichtige Kapitel

Der Sprecher: Norbert Rieger



Norbert Rieger ist bei dbaservices GmbH in Neu-Isenburg als Senior Oracle-DBA tätig. Zuvor war er über 15 Jahre bei einer mittelständischen Firma im Bereich Automotiv ebenfalls als Senior Oracle-DBA tätig und maßgeblich an globalen Projekten beteiligt. Hierzu gehören Projekte wie Jahr2000, Plattformwechsel von Datenbanken, globales Monitoring und Einsatz von Unicode in China. Sein besonderer Schwerpunkt ist die Performance der Datenbanken. Er ist zertifiziert als "Oracle Database Performance and Tuning Implementation Specialist".

Kontakt: norbert.rieger@dbaservices.de

Freitag, 18. Dezember 2015, 10:30 - 11:15 Uhr

Markus Winand:

Den Suchraum des Optimizers gestalten

Übersicht

Auf seiner Suche nach dem optimalen Ausführungsplan ist der Optimizer in einen Käfig gesperrt, der von den SQL-Abfragen und dem physischen Datenbank-Design vorgegeben ist. Per Definition kann der Optimizer nur den besten Ausführungsplan innerhalb dieses Käfigs finden – auch wenn außerhalb bessere Ausführungspläne sind. In diesem Vortrag werden wir sehen, wie Indizierung diesen Käfig beeinflusst und lernen, wie man durch kluge Indizierung den Käfig so gestaltet, dass der absolut beste Ausführungsplan in der Reichweite des Optimizers liegt.

Der Sprecher: Markus Winand



Markus Winand hat sich auf effizientes SQL spezialisiert. Dabei deckt er sowohl den Entwicklungs- als auch den Betriebsaspekt ab: Entwicklern zeigt er, wie man Business-Anforderungen mit SQL schneller umsetzen kann, als es mit anderen Sprachen möglich wäre. Darüber hinaus stellt er sicher, dass die in SQL entwickelten Lösungen auch bei wachsenden Datenmengen und Zugriffszahlen die optimale Performance bringen. Dazu hat er auch das Buch "SQL Performance Explained" veröffentlicht, das mittlerweile in vier Sprachen erhältlich ist.

Freitag, 18. Dezember 2015, 11:30 - 12:15 Uhr

Sven Weller:

Materialized Views – Praktischer Einsatz vor und in 12c

Einleitung

Materialized Views (MV) bieten sowohl eine einfache Lösung für eine voraggregierte Form großer Datenmengen als auch für die Vervielfältigung gleicher Daten auf verschiedene Standorte (Replikation). Der Vortrag arbeitet heraus, welche Vorteile Materialized Views gegenüber normalen Tabellen haben. Er geht auf die tollen kleinen Features ein, die Oracle bereitstellt, um eine MV für nahezu jedes Szenario einsatzfähig zu machen. Hierzu zählen unter anderem:

- Performance Optimierung beim Refresh Vorgang (atomicrefresh)
- Inkrementelle Refreshes beim Replizieren großer Datenmengen (fast refresh)
- Performance Optimierung für Queries, die die MV bereits kennt (queryrewrite)
- Change Management für MVs - geringe Downtime (prebuildtable)
- Eine interessante Neuerung in 12c (out of place refresh)

Der Vortrag berichtet Praxisbeispiele (aus dem Pharma-Umfeld) und realistische Zahlen. Das hilft einzuschätzen, wann und weshalb sich der Einsatz von MVs lohnt und welche Fallstricke Sie unbedingt beachten sollten.

Im Vortrag wird nicht behandelt, welchen Einschränkungen das SQL unterliegt (connectby, setoperators, distinct, not exists). Replikationslösungen können außerdem so aufgebaut werden, dass UPDATES auf Seite der Satelliten möglich sind (updateableMVs).

Was sind Materialized Views?

MVs sind zum einen ein Performance Feature, das ohne zusätzliches Tuning für schnellere SQLs sorgt. Zum anderen lassen sich Tabellen ohne großen Aufwand (Trigger, Tracken der Änderungen, Updates...) replizieren. Grundsätzlich kann eine MV wie eine Tabelle in Oracle behandelt werden. DML Operationen können durchgeführt werden, wenn die vorhandenen Rechte eingeräumt wurden und gewöhnliche SQL Abfragen funktionieren reibungslos. Auch ein Index lässt sich auf beliebige Spalten der MV anlegen.

Betrachtet man die nach einem CREATE MATERIALIZED VIEW entstandenen Objekte, wird deutlich, dass Oracle im Hintergrund sogar eine Tabelle angelegt hat:

```
select SEGMENT_TYPE
from user_segments
where SEGMENT_NAME = 'MV_NAME';
SEGMENT_TYPE
```

TABLE

Anders als bei Tabellen wird beim Anlegen einer MV allerdings ein SQL hinterlegt, welches die Daten beschreibt, die in der Datenbank abgelegt werden. Nimmt man den Namen "Materialized View" genauer unter die Lupe, dann steckt diese Eigenschaft schon darin:

- Materialized = "gespeichert/persistiert"
- View = "SQL Definition"

MVs bieten somit eine Möglichkeit, aufwendige SQLs (die unter Umständen auf sehr großen Datenmengen aufsetzen) in einer voraggregierten Form in der Datenbank abzulegen. Ein performancekritisches SQL könnte dann beispielsweise nachts laufen. Analysen, die tagsüber mit geringeren Abfragezeiten gegen die Datenbank abgesetzt werden müssen, können mit einfacheren SQLs darauf aufsetzen. Was aber ist der Vorteil gegenüber einer normalen Tabelle, die mittels

```
CREATE TABLE daily_statsas(
    SELECT avg(col)--...
    FROM BIG_TABLE
    --WHERE...
);
```

das selbe bieten könnte? - Oracle liefert mit der query rewrite Option die passende Antwort. Durch eine Angabe beim Erstellen der MV kennt die Oracle Datenbank das in der MV hinterlegte SQL und erkennt automatisch, wenn Queries, die adhoc in einer Session (OLAP) gegen die Datenbank geschickt werden, auf den voraggregierten Datenbestand umgebogen werden können, weil dieser bereits die notwendigen Ergebnisse enthält. So komplex dieses Performance Feature im Hintergrund auch implementiert sein mag - es ist spielend einfach mittels des folgenden Befehls einzusetzen:

```
CREATE MATERIALIZED VIEW schema.mv_name
ENABLE QUERY REWRITE
AS (SELECT avg(col)--...
FROM BIG_TABLE
--WHERE...
);
```

Ob ihre Query dann einem rewrite unterzogen wurde, zeigt Ihnen der Explain Plan an:

Id	Operation	Name	Rows	Bytes	Cost
----	-----------	------	------	-------	------

0	SELECT STATEMENT		4	123	3
---	------------------	--	---	-----	---

| 1 | MAT_VIEW REWRITE ACCESS FULL| MV_NAME | 4 | 545 | 3

Damit ist der Einsatz im DWH-Umfeld besonders attraktiv. Ein weiterer Vorteil ist die Vervielfältigung gleicher Daten auf verschiedene Standorte (Replikation). Der wesentliche Vorteil gegenüber Tabellen Objekten ist hierbei, dass der Refresh Vorgang bereits durch Oracle-Funktionalitäten erreicht werden kann. Einer Tabelle würde diese zusätzliche Intelligenz fehlen. Dem Entwickler stehen sogar verschiedene Methoden zur Verfügung: Er kann über den Zeitpunkt des Refreshs (bei Commit oder manuell gesteuert) und über die Art des Updates (inkrementell oder komplett) entscheiden. Hierzu müssen die notwendigen Einstellungen ebenfalls im Header definiert werden:

```
CREATE MATERIALIZED VIEW schema.mv_name  
REFRESH FAST ON DEMAND  
--REFRESH COMPLETE ON COMMIT  
--REFRESH FAST ON COMMIT  
--REFRESH COMPLETE ON DEMAND  
AS (select --...
```

Entscheidet man sich für den manuellen Refresh (on demand) so aktualisiert sich der Datenbestand der MV nur dann, wenn der Befehl `exec dbms_mview.refresh('MV_MASTER', 'C');` ausgeführt wird.

Der zweite Parameter bestimmt, ob die MV lediglich inkrementelle Änderungen erhält („F“ wie fast), oder ob der gesamte Datenbestand erneuert wird („C“ wie complete). Die Art der Refreshs, die in der MV Definition mitgegeben wurde, würde vom Parameter dann überschrieben werden. Eine mittels `refresh fast on demand` angelegte MV kann somit ebenfalls mittels `complete refresh` erneuert werden. Das macht auch Sinn: Wenigstens beim Anlegen muss schließlich einmal der gesamte Datenbestand ermittelt werden. Oracle erkennt das automatisch. Die Prozedur könnte z. B. regelmäßig über einen `scheduled job` aufgerufen werden.

Performance-Überlegungen

Unter Umständen lohnt sich eine weitere Performance-Optimierung während des Refresh-Vorgangs. Es handelt sich um einen dritten Parameter in der Prozedur (`atomic_refresh`). Er kann nur beeinflusst werden, wenn man den refresh manuell anstößt (nicht bei `on commit`). Per Default steht er auf „true“, sodass die Frage naheliegt, was ein `atomic_refresh=false` bewirkt.

Wird der gesamte Datenbestand erneuert, so führt Oracle im Hintergrund ein `TRUNCATE` statt einem `DELETE` aus. Alle neu einzutragenden Rows werden mittels dem vom Loader bekannten `/*+ append*/` Hint angehängt, ohne die eventuell durch ein `DELETE` entstandenen Lücken im Datenblock zu suchen. Das sorgt im Hintergrund für wesentlich

weniger UNDO und REDO (lediglich Metadaten, keine Sicherung der eigentlichen Daten) Informationen. Außerdem werden erst nach dem Eintrag aller neuen/ geänderten Rows eventuelle Indizes gepflegt. Auch der Zeitpunkt der Commits kann variieren.

Sie sollten definitiv bedenken, dass das TRUNCATE nicht in allen Fällen eine geeignete Lösung ist. Bis die neuen Daten geladen sind, liefern Abfragen gegen die MV keine Daten. Durch einen directpathload, wenn kein TRUNCATE gemacht wurde, entstehen Lücken unterhalb der high watermark, was sich negativ auf die Performance auswirken kann. Auf diese Besonderheiten wird im Vortrag weiter eingegangen. Es ist Aufgabe des Entwicklers, die richtige Kombination zu finden. Nur weil der ein refresh FAST genannt wird, muss er z. B. nicht immer der schnellste sein.

Change Management

Eine weitere Einstellungsmöglichkeit kann die Arbeit in der Praxis extrem erleichtern:

```
CREATE MATERIALIZED VIEW schema.mv_name  
ON PREBUILD TABLE  
AS (select
```

Haben Sie schon einmal versucht, vorhandene Spaltennamen abzuändern oder neue Spalten zur MV hinzuzufügen? Ein einfaches ALTER MATERIALIZED VIEW oder CREATE OR REPLACE werden Sie nicht finden. Das erschwert Änderungen an Materialized Views – gerade wenn eine geringe Downtime notwendig ist. Bedenken Sie, dass Ihr hinterlegtes SQL unter Umständen mehrere Tage laufen könnte bis es alle Daten ermittelt hat. Während dieser Zeit wäre das Objekt für den Livebetrieb nicht erreichbar! Im Vortrag wird die prebuildtable Option (und auch build deferred) erläutert. Es handelt sich um die Möglichkeit, eine bereits vorhandene Tabelle in eine MV zu überführen. Somit können die Daten zunächst in eine Tabelle geladen werden. Nebenbei setzen Abfragen weiterhin auf der alten MV auf. Ist die Tabelle fertig, wird sie zu einer MV umgewandelt und um die nützlichen Features für den Refresh etc. ergänzt.

Fallstricke könnten sich ergeben, sobald Sie die eigentliche MV wieder löschen. Im Gegensatz zu direkt erzeugten MVs bleibt das eigentliche Tabellen Objekt erhalten. In der Praxis sorgte folgendes Szenario, welches in Abbildung 1 gezeigt ist in der Oracle Version 10.2.0.5.0 - 64bit, für fehlerhafte SQLs: Die Abbildung zeigt, dass MVTOP mit seinem SQL auf MVERR geht (Datenbanklink zu DB 2). In DB2 wurde ein Synonym MVERRSYN angelegt, welches auf die darunterliegende MVERR zeigte. Diese MVERR wurde gedropt und das Synonym wurde auf MVNEW umgebogen. Das hinterlegte SQL in MVTOP hat das SYNONYM scheinbar aufgelöst und in den darunterliegenden Objektnamen MVERR umgewandelt.

Da nur die MV und nicht der prebuild table gelöscht werden, findet DB3 weiterhin ein Objekt MVERR und sieht keinen Anlass den Cache zu erneuern. Die Daten werden also fälschlicherweise noch von MVERR selektiert, statt von MVNEW.

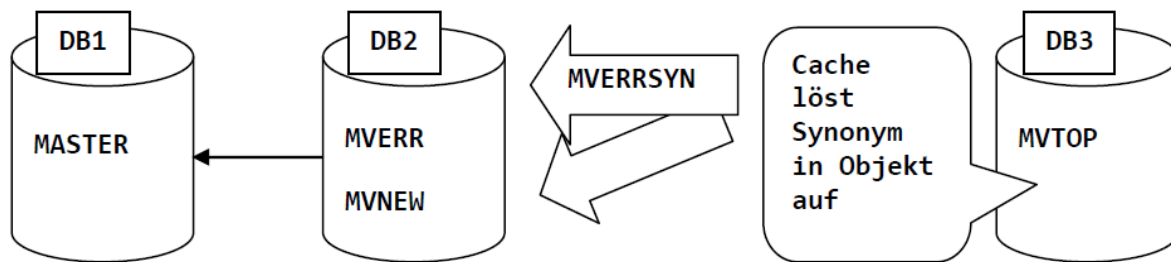


Abb. 1: Caching Probleme bei der prebuildtable Option in 10.2.0.5.0

Abhilfe würde eine explizite Erneuerung des Caches mittels `alter system flush shared_pool` schaffen. Allerdings sind dazu DBA-Rechte erforderlich. Häufig hat der Entwickler hierauf also keinen Einfluss. Mit einem `DROP TABLE MVERR` löschen Sie den prebuild table und sind auf der sicheren Seite.

Neuerungen in 12c

Im Bereich des Change Managements ist auch die interessanteste Neuerung in 12c angesiedelt. Der `outofplace-Refresh` kommt als vierter Parameter zur bereits bekannten Refresh-Prozedur hinzu:

```
DBMS_MVIEW.REFRESH('MV_NAME', 'F', FALSE, out_of_place => TRUE);
```

Beim Refresh wird Oracle im Hintergrund automatisch eine Kopie der MV in einer neuen Tabelle mit dem Präfix „RV\$“ (gefolgt von der hexadezimalen Objekt-Id der ursprünglichen MV) erzeugen. In dieser findet die neue Datensammlung statt. Danach transformiert Oracle analog zum prebuild table Vorgang die Tabelle in eine MV.

Zwar werden alle Arten (`fast`, `complete` und `force`) unterstützt, jedoch ist der `atomic_refresh=false` zwangsläufig einzusetzen, wenn `out_of_place=true` gewünscht ist. Oracle würde eine falsche Kombination der Parameter mit einer entsprechenden Fehlermeldung quittieren: Damit sind Refresh-Vorgänge, welche `ON COMMIT` getriggert werden, natürlich auch wieder aus dem Rennen. MVs mit LOB-Spalten oder materialized view logs werden derzeit noch nicht unterstützt. Andere Einschränkungen können Sie der Dokumentation entnehmen.

Weil der `out of place refresh` ein `INSERT` in eine vollkommen neue Tabelle vornimmt, wird Oracle noch keine Statistiken kennen. Diese sollten also noch einmal explizit berechnet werden, z. B.

```
dbms_stats.gather_table_stats (
    ownname => 'MOSAIC_CUSTOM'
    , tabname => 'RPT_BATCHES_MONTHLY_OLD'
    , estimate_percent => dbms_stats.auto_sample_size
    , block_sample => true);
```

atomic refreshes=false hat übrigens ähnliche Probleme. Für die Indizes auf der MV sollte das Ganze anders aussehen, da diese ja erst nach dem INSERT erzeugt wurden (siehe oben).

Geheimtipp

Häufig werden in der Praxis Datenbanklinks eingesetzt, welche dann von den Satelliten zum Master zeigen. Je nach Datenbestand kann das, was übers Netzwerk ausgetauscht wird, ziemlich viel werden. Wenn das SQL Joins enthält, hat die Satelliten Datenbank unter Umständen keinen Zugriff auf wichtige Statistiken in der Masterinstanz. Das könnte im schlimmsten Fall zu einem fulltable Scan führen. Dann lohnt sich ein Test mit dem `/*+driving_site(site)*/` Hint.

Hier kann in Klammern gesteuert werden, welche der beiden Datenbanken den Join ausführen soll. Liegt die kleinere Tabelle zum Beispiel im Satelliten, wird das Resultset durch den Join mit dem Datensatz im Master stark reduziert. Wenn also die Berechnung vom Master ausgeführt wird und nur das eigentliche Ergebnis über das Netzwerk mitgeteilt wird, verringert sich die Netzwerklast. Vorsicht ist bei userdefinedfunctions geboten. Die könnten nicht im Master definiert sein, was zu einer Art Daten-Ping-Pong werden könnte.

Fazit

Der Vortrag zeigt anhand klarer Praxisbeispiele und realistischer Zahlen, wann und weshalb sich der Einsatz von Materialized Views lohnt. Man könnte sie als eine Art Tabelle mit zusätzlicher Intelligenz beschreiben. Das bezieht sich vor allem darauf, dass sie wissen, welche Daten sie enthalten, wie sich andere Abfragen diese zu Nutze machen können und wie häufig dieser Datenbestand aktualisiert wird.

Für eventuelle Rückfragen stehen wir Ihnen gerne unter der Rufnummer 06102 298668 oder per E-Mail an office@syntegris.de zur Verfügung.

Der Sprecher: Jonas Gassenmeyer



Jonas Gassenmeyer ist Consultant bei der syntegris information solution GmbH, besitzt den Master in Wirtschaftsinformatik und ist Oracle Certified Associate in Oracle 12c (SQL und PL/SQL). Sein mehrjähriges Branchen-Know-how im Finanzsektor und die Projekterfahrungen im Chemie-/Pharma-Bereich teilt er in diversen Fachvorträgen. In diesem Jahr war Jonas Gassenmeyer zum Beispiel als Sprecher auf der DOAG Konferenz in Nürnberg vertreten.

Kontakt: office@syntegris.de

Freitag, 18. Dezember 2015, 13:30 - 14:15 Uhr

Randolf Geist:

Oracle Parallel Execution – Analyse und Troubleshooting

Zusammenfassung

Oracle Database Enterprise Edition erlaubt mittels dem Parallel Execution Feature die automatische Verteilung einer SQL-Ausführung auf mehrere Parallel Execution Server. Dabei können spezielle Performance-Probleme auftreten, die es so mit normaler serieller Ausführung nicht gibt.

In diesem Vortrag wird eine Übersicht geboten, welche Herausforderungen es mit paralleler Ausführung im Allgemeinen geben kann, und wie diese auf Oracle Parallel Execution zutreffen.

Desweiteren wird erläutert, wie diese speziellen Probleme identifiziert und analysiert werden können. Dieses Verständnis ist wichtig für eine systematische Herangehensweise beim Troubleshooting von Parallel Execution und daher empfehlenswert für alle DBAs, Entwickler und Anwender, die Parallel Execution ernsthaft verwenden wollen.

Einleitung

Parallel Execution sollte nur in speziellen Fällen eingesetzt werden, in denen vorher evaluiert wurde, dass es keine bessere Lösung gibt. Der wichtigste Punkt ist, dass Parallel Execution im Grunde ein „Brute Force“-Ansatz ist. Dies bedeutet, dass man nicht versucht, ein Performance-Problem durch Effizienzerhöhung zu lösen, sondern durch die Verwendung von mehr Ressourcen, insbesondere CPU und Arbeitsspeicher.

Es sollte also klar sein, dass Parallel Execution tatsächlich dazu führt, dass man insgesamt mehr und härter arbeitet im Vergleich zu einer optimalen seriellen Ausführung, vor allem durch den Overhead, der durch Parallel Execution verursacht wird. In vielen Fällen kann man aber eine geeignete Performance-Steigerung dadurch erreichen, weniger bzw. effizienter zu arbeiten – angefangen bei effizienteren Ausführungsplänen, bis hin zu Verwendung von optimierten Speicherstrukturen in der Datenbank, wie cleveres Index-Design, Cluster oder Index Organized Tables.

Diese Optionen sollten also evaluiert werden, bevor man versucht, eine Performance-Steigerung mittels Parallel Execution zu erreichen.

Spezielle Herausforderungen

Es gibt einige Probleme, die nur auf Parallel Execution zutreffen. Die wichtigsten haben damit zu tun, wie die zu verrichtende Arbeit auf die N Prozesse verteilt wird, die gleichzeitig arbeiten sollen, um die Gesamtlaufzeit zu verringern.


Dabei gibt es zwei Grundformen:

1. Es kann passieren, dass sich ein Problem grundsätzlich nicht passend aufteilen lässt („Parallel Unfriendly“), wobei es hier noch die Variante geben mag, dass theoretisch eine Aufteilung möglich wäre, aber eine entsprechende Implementierung seitens Oracle (noch) nicht umgesetzt wurde.
Ein gutes Beispiel für diese Form ist eine Abfrage nach dem Median über einen gesamten Datenbestand. Oracle kann zwar die Datenermittlung parallelisieren (Parallel Full Table Scan), aber die eigentliche Berechnung des Median erfolgt im zentralen Prozess (Query Coordinator), der sonst nur für die Steuerung der Parallel Worker und Kommunikation mit dem Client zuständig ist. Insofern reduziert diese Parallel Execution sehr wahrscheinlich die Gesamtausführungszeit nur im geringen Maße, falls überhaupt, und wird vor allem nicht schneller, wenn man versucht, die Parallelität zu erhöhen. Der Grund hierfür liegt in diesem Fall darin, dass der gesamte Datenbestand im zentralen Prozess sortiert wird.

PARALLEL UNFRIENDLY

```
select median(id) from t2;
```

Id	Operation	Name	TQ	IN-OUT	PQ	Distrib
0	SELECT STATEMENT					
1	SORT GROUP BY					
2	PX COORDINATOR					
3	PX SEND QC (RANDOM)	:TQ10000	Q1,00	P->S	QC	(RAND)
4	PX BLOCK ITERATOR		Q1,00		PCWC	
5	TABLE ACCESS FULL	T2	Q1,00		PCWP	



2. Ein Problem kann grundsätzlich aufgeteilt werden und wird so auch von Oracle unterstützt, fällt also nicht unter 1., aber die Verteilung der Arbeit auf die N Prozesse ist nicht gleichförmig („Parallel Execution Skew“).
In diesem Fall sind nicht alle Prozesse permanent ausgelastet, so dass es zu signifikanter „Idle Time“ dieser Prozesse kommen kann, was natürlich dazu führt, dass die gewünschte Reduktion der Ausführungszeit nicht in dem Maße erreicht werden kann, wie erwartet.
Im schlechtesten Fall kann Parallel Execution in so einer Konstellation sogar

langsamer als eine serielle Ausführung werden, da Parallel Execution eben mit einigem Overhead kommt, so dass ein vergleichbarer Ausführungsplan langsamer bei Verwendung von Parallel Execution sein kann.

Parallel Execution Skew

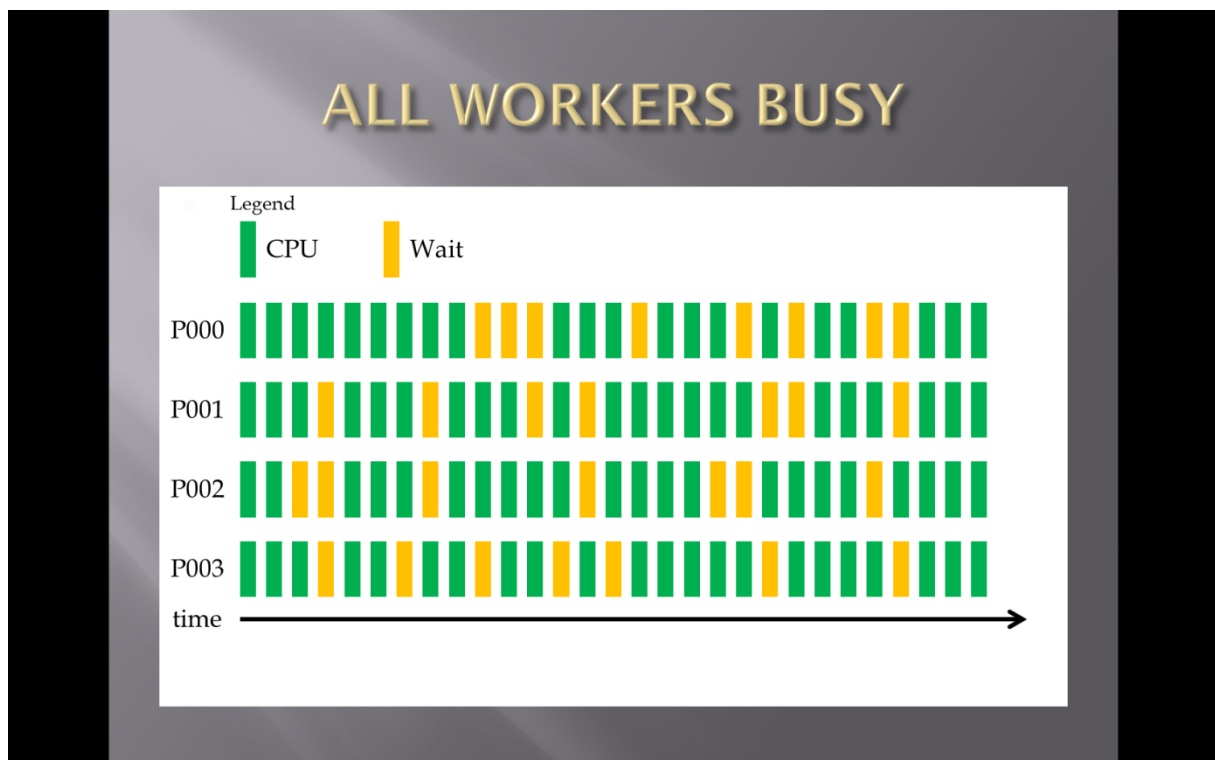
Die unter 2. erwähnte Problematik dient besonderer Erwähnung, da sie in der Praxis mit am häufigsten auftritt und dazu führen kann, dass Parallel Execution wesentlich weniger die Ausführungszeit verringert, als theoretisch möglich wäre.

Bei der Ungleichverteilung der Arbeit auf die Prozesse gibt es wiederum zwei Arten, die man grundsätzlich unterscheiden kann. Dazu ein paar Grafiken, die die verschiedenen Formen veranschaulichen sollen.

Gute Auslastung

Betrachtet man die Auslastung der beteiligten Prozesse, sollte sich idealerweise ein Bild ähnlich dem folgenden ergeben. Exemplarisch sind hier vier Prozesse visualisiert (P000 – P003) und die Boxen stehen für einen bestimmten Zeitraum (zum Beispiel ein Sampling der Aktivität jede Sekunde → Active Session History).

Die Farbe der Box zeigt zusätzlich an, ob der Prozess grundsätzlich auf CPU ist und arbeitet, oder auf etwas aktiv wartet zur Weiterverarbeitung – bei einer Datenbank ist dies häufig Warten auf Daten (I/O waits).



Hier sind die vier untersuchten Prozesse also grundsätzlich immer ausgelastet, und idealerweise auch meistens auf CPU.

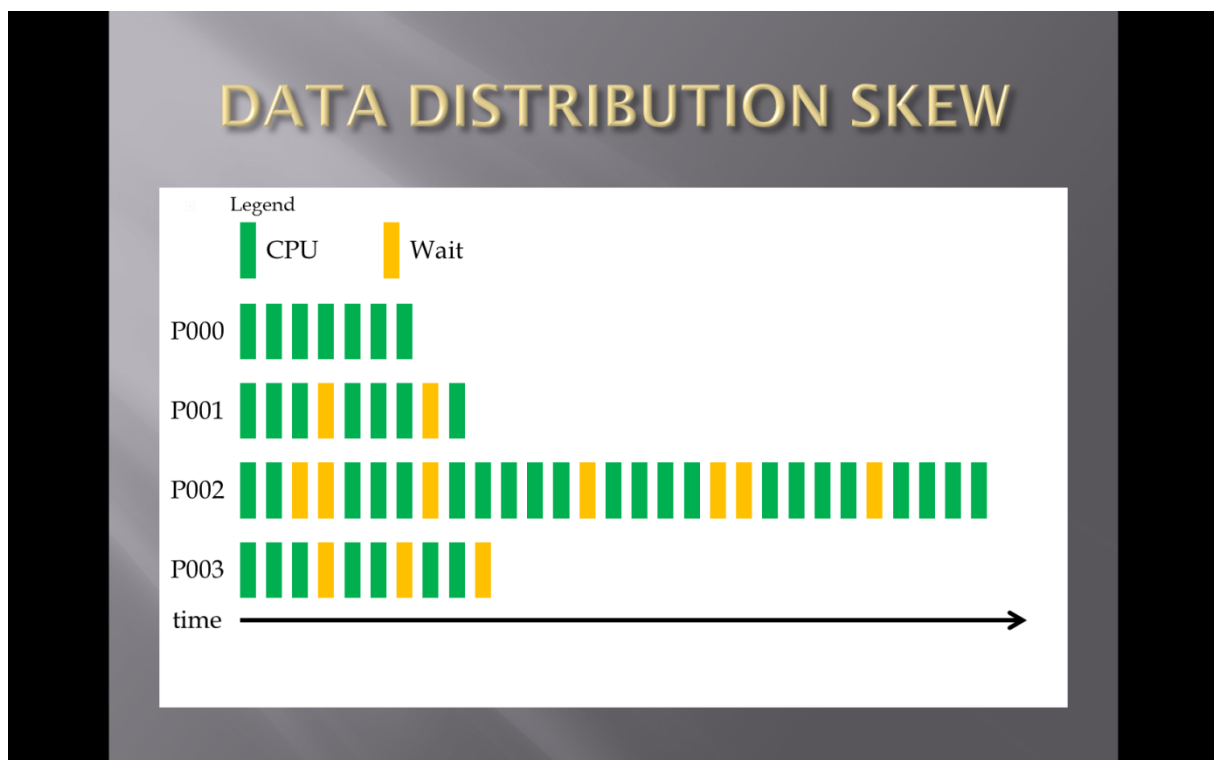
Sollten die Prozesse zwar ausgelastet sein, aber die meiste Zeit warten, dann ist eine Performance-Steigerung gegenüber serieller Ausführung auch nur sehr begrenzt möglich.

Es ist sehr wichtig zu verstehen, dass wir hier nur die Auslastung der Prozesse betrachten. Dies ist keine Bewertung der tatsächlichen Arbeit, die verrichtet wird. Insofern ist es möglich, einen völlig ineffizienten Ausführungsplan mittels Parallel Execution auszuführen, der zwar eine wunderbare Auslastung der beteiligten Prozesse erreicht, aber trotzdem viel zu lange benötigt.

Umgekehrt ist es möglich, einen ansonsten effizienten Ausführungsplan zu haben, der mittels Parallel Execution ausgeführt wird – werden dabei aber die Prozesse nicht richtig ausgelastet, ist eine Reduzierung der Ausführungszeit trotz ansonsten effizientem Ausführungsplan mittels Parallel Execution nur sehr begrenzt möglich, falls überhaupt.

Data Distribution Skew

Werden die zu verarbeitenden Daten ungleich auf die beteiligten Prozessen verteilt, dann kann folgendes Bild entstehen:

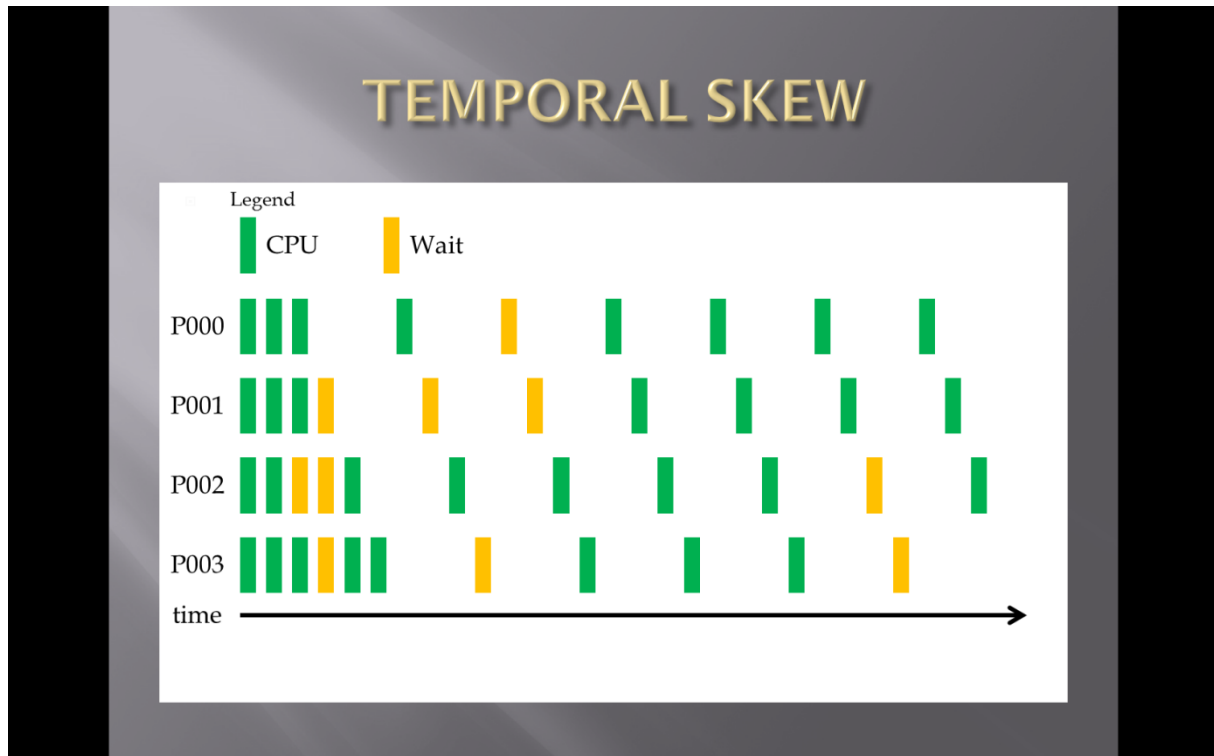


Hier ergibt sich also eine signifikante „Idle Time“ bei dreien der beteiligten vier Prozesse – und der Löwenanteil der Arbeit muss von einem der vier verrichtet werden.

Offensichtlich wird hier die Gesamtlaufzeit von den Prozessen bestimmt, die den größten Teil der Arbeit verrichten müssen, und diese kann im schlechtesten Fall sogar gleich oder über der einer seriellen Ausführung liegen.

Temporal Skew

Es gibt noch eine andere Möglichkeit, wie die Gesamtauslastung der beteiligten Prozesse weniger als theoretisch möglich sein kann: Die zu verarbeitenden Daten werden zwar gleich verteilt, aber die Prozesse sind zwischendurch immer wieder „idle“, so dass sie zwar alle ungefähr zur gleichen Zeit fertig sind, und auch eine ähnliche Datenmenge zu verarbeiten haben, aber die Reduzierung der Gesamtlaufzeit durch die signifikante „Idle Time“ limitiert ist:



Man kann darüber diskutieren, ob es sich in diesem Fall tatsächlich um „Idle Time“ handelt, da meistens auf zu empfangende Daten von anderen Prozessen gewartet wird, insofern könnte man es auch als eine bestimmte Form des „aktiven“ Wartens bezeichnen.

Diese Form tritt übrigens häufig im Zusammenhang mit „Data Distribution Skew“ auf, falls eine Umverteilung der Daten notwendig ist (benötigt den Austausch der Daten zwischen den Prozessen), und im vorherigen Schritt ein Großteil der Daten von nur einem oder wenigen Prozessen verarbeitet wurde.

Analyse und Diagnose

Wie können nun solche Fälle diagnostiziert werden, bei denen die Auslastung der an der Parallel Execution beteiligten Prozesse nicht dem Ideal entspricht?

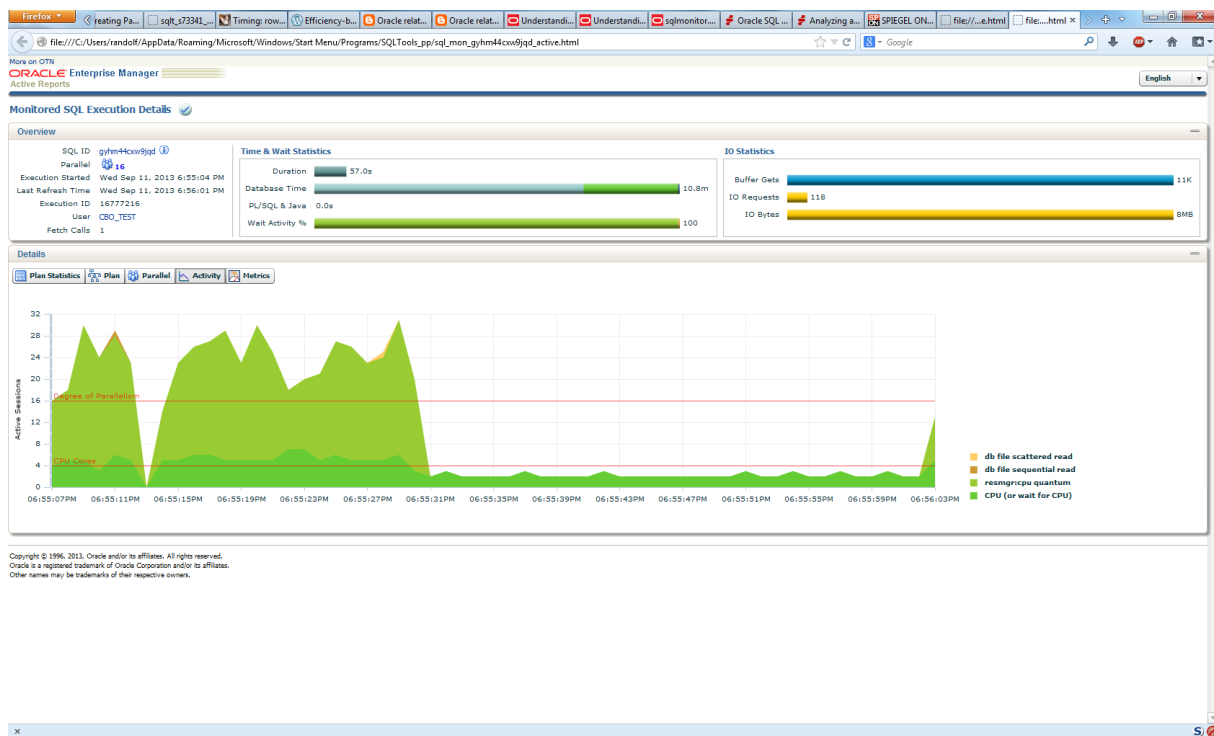
Diese Möglichkeit der Analyse ist wichtig, um zu verstehen, warum eine Parallel Execution nicht zu der gewünschten Reduzierung der Laufzeit führt.

Es muss unterschieden werden zwischen dem Fall, bei dem eine grundsätzliche Ineffizienz vorliegt – dieser zeichnet sich dadurch aus, dass „zu viel“ gearbeitet wird, und dem hier beschriebenen Phänomen, bei dem grundsätzlich „zu wenig“ gearbeitet wird.

In der Vergangenheit war mit den von Oracle zur Verfügung gestellten Mitteln eine solche Diagnose schwierig. Seit der Version 11.2 stellt das sogenannte „Real-Time SQL Monitoring“ Informationen zur Verfügung, die eine solche Analyse vereinfachen. Leider benötigt das Feature die Diagnostic + Tuning Pack Zusatzlizenz, steht also ohne diese nicht zur Verfügung.

In der sogenannten „Enterprise Manager“ oder auch „Active“-Version (und nur dort) eines „Real-Time SQL Monitoring“-Berichts gibt es den Abschnitt / Tab Aktivität (Activity).

Dieser erlaubt eine visuelle Darstellung der Auslastung der beteiligten Prozesse, und somit eine zuverlässige Aussage (mit Ausnahme von bestimmten Fällen, die ich in meinem „Parallel Execution Internals Masterclass“ Seminar erkläre), ob es signifikante Ausreißer in der Auslastung gab:



Im Bild oben kann man erkennen, dass ein Parallel Degree (DOP) von 16 angegeben ist, aber die Aktivität nur für eine bestimmte Zeit am Anfang der Ausführung im (oder sogar über) dem erwarteten Bereich lag, und für den Rest der Zeit in einem viel niedrigeren Bereich.

Dies ist die einfachste Art und Weise festzustellen, ob es ein Problem mit der Auslastung der Prozesse gab oder nicht. Leider benötigt sie die entsprechenden Lizenzen, und ist daher nicht ganz kostengünstig.

Es gibt noch weitere Möglichkeiten der Diagnose, diese zu beschreiben, sprengt aber den Rahmen dieses Vortrags.

Problembhebung

Falls ein Problem mit der Aktivität der beteiligten Prozesse als Grund für eine zu lange Laufzeit bei Verwendung von Parallel Execution festgestellt wurde, wie kann dieses adressiert werden? Je nach genauen Umständen gibt es unterschiedliche Lösungsmöglichkeiten, die auch mit unterschiedlichem Umsetzungsaufwand einher kommen.

Die einfachste Lösung kann in der Änderung der sogenannten Distributionsmethode bestehen (Parallel Distribution Method) – entweder durch Beeinflussung der Datenvolumenabschätzungen des Optimizers, oder explizite Angabe des PQ_DISTRIBUTE-Hints. Jedoch ist diese Maßnahme nicht in allen Fällen sinnvoll und kann auch zu extremer CPU- und Speicher- bzw. TEMP-Auslastung führen.

Andere Maßnahmen können die Limitierung des Skews durch Beeinflussung der Join-Reihenfolge sein, oder im aufwendigsten Falle das spezielle Umschreiben der Abfrage, um das Problem zu begrenzen oder sogar zu umgehen, je nach genauer Sachlage.

Möglicherweise kann das Problem auch nur umgegangen werden, indem die Art und Weise, wie die Daten in der Datenbank repräsentiert werden, angepasst wird.

Der Sprecher: Randolph Geist



Randolf Geist ist seit 15 Jahren als freiberuflicher Oracle Datenbank-Experte tätig und auf Performance-Themen spezialisiert. Im Bereich der Oracle Optimizer-Technologie und SQL Performance Analyse gehört er zu den Top-Experten weltweit. Er veröffentlicht regelmäßig entsprechende Beiträge auf seinem Blog und seinem YouTube-Kanal, und gibt sein Wissen in Webinaren, den Oracle OTN-Foren und auf Konferenzen weiter. Desweiteren werden seine Experten-Seminare bei der Oracle University angeboten. Randolf ist Mitglied des OakTable Networks und des Oracle ACE Director-Programms.

Kontakt: randolf.geist@oracle-performance.de

Freitag, 18. Dezember 2015, 14:30 - 15:15 Uhr

Thorsten Bruhns:

Monitoring von Oracle-Datenbanken mit check_mk

Check_MK Monitoring System

Das Check_MK Monitoring System hat seinen Ursprung im Projekt Check_MK, das 2008 ins Leben gerufen wurde und heute die wichtigste Komponente im gesamten System ist. Es entstand als eine Erweiterung für den Nagios-Core, um die Verwaltung von großen Systemumgebungen stark zu automatisieren und um bekannte Performance-Probleme von Nagios mit Hilfe einer innovativen Architektur zu lösen.

Der Vortrag wird sich mit den Besonderheiten bei der Überwachung von Oracle Datenbanken mit Hilfe des speziell dafür entwickelten Plugins mk_oracle befassen. Es werden die Grundlagen eines Check_MK Monitoring Systems sowie die Highlights von mk_oracle erläutert.

Architektur des Check_MK Monitoring Systems

Das Monitoring System besteht aus einer Vielzahl von Komponenten, die zusammen das Check_MK Monitoring System bilden.

Das System wird mit Hilfe der Open Monitoring Distribution – kurz OMD – als 1 RPM für Intel x86 Linux bereit gestellt, so dass die Installation extrem einfach ist. Es ist für alle gängigen Linuxdistributionen verfügbar und bietet die Möglichkeit, unterschiedliche Versionen und mehrere Sites gleichzeitig auf einem Host betreiben zu können. So ist eine hohe Flexibilität bei Upgrades sowie der Betrieb von mehreren Mandanten parallel möglich. Die Steuerung übernimmt hier OMD, was als Projekt unter der GPL 2010 gegründet wurde, um die Installation von nagiosbasierten Monitoringlösungen möglichst einfach zu gestalten. Ziel von OMD soll sein, das sich der Administrator nicht um die Installation und Konfiguration der Teilkomponenten kümmern muss und zudem parallel verschiedene Versionen installiert werden können. Der Administrator soll sich primär um die Konfiguration der Ziele kümmern und sich keine Gedanken über die Softwareinstallation machen müssen. Im praktischen Betrieb bei OPITZ CONSULTING hat sich gezeigt, das dieses Ziel vollständig erreicht wurde!

2008 ist Check_MK als Erweiterung für den Nagios-Core entstanden. Es sollte ein System entwickelt werden, das das Pflegen von Nagioskonfigurationen automatisiert, in dem entweder per snmpwalk oder über ein Agentenmodell Ergebnisse von den Zielsystemen eingeholt werden. Diese Daten werden in Check_MK geparkt und in entsprechende Services umgewandelt, die dann als Konfigurationsdatei für den Check_MK Micro Core (CMC) bzw. Nagioscore bereit gestellt werden. So können automatisiert eine große Anzahl von Systemen ins Monitoring aufgenommen werden, ohne umfangreich Konfigurationen pflegen zu müssen. Als Nebeneffekt werden so auch neue Checks – z. B. neue Tablespace – erkannt, da

der Agent auf dem zu überwachenden System immer alle Informationen über alle erkannten Tablespaces liefert.

Die Definition von Schwellwerten erfolgt mit Regelwerken, die mit Hilfe von Tags und eines hierarchischen Model vererbt werden können. Die Regelwerke sind neben der automatischen Erkennung und das Erzeugen der Konfigurationsdateien die wichtigste Funktion von Check_MK.

Der Check_MK-Agent für Windows ist in C geschrieben, während er für alle anderen Plattformen in Shell geschrieben wurde. So ist eine einfache Pflege und hohe Sicherheit bzgl. der Funktionalität im Agenten gegeben, da alle Sourcen frei verfügbar sind. Darüber hinaus besitzt das System ein Plugin-Modell, das die Erweiterung des Agenten ermöglicht. Daher hat der normale Agent lediglich die Basisfunktionen für das Hostsystem enthalten, während alle Erweiterungen wie mk_oracle für Oracle Datenbanken als Plugin eingebunden werden.

der Agent liefert mit seinen Plugins lediglich die Rohdaten für das Monitoringsystem, so dass die Logik extrem einfach gehalten ist. Hier werden als Beispiel alle Filesysteme, Tablespaces, Oracle Instanzen, Prozesse usw. als Informationen geliefert. Ob hier ggf. eine Schwellwertverletzung vorhanden ist, entscheidet nicht der Agent sondern der Monitoring Core. Als Nebeneffekt kann der Core diese Daten als Performancedaten abspeichern und so Trenddaten liefern. Zudem kann der Core so erkennen, ob in einer Datenbank ein neuer Tablespace angelegt wurde und eine entsprechende Warnung erzeugen, damit der zugehörige Service mit ggf. notwendigen Schwellwerten angelegt werden kann.

mk_oracle Plugin - Historie

Das Plugin wurde 2011 im Rahmen einer Kundenanforderung von Mathias Kettner entwickelt und stetig erweitert. 2013 erfolgte eine Evaluierung zur Nutzung von Check_MK als Monitoringlösung zur Überwachung von Oracle Datenbanken im Rahmen von Managed Services bei OPITZ CONSUTLING. Die Entscheidung für das System ist gefallen, weil die Grundfunktionen sehr gut waren – insbesondere was die Überwachung von Hostsystemen und das Web-Frontend betraf. Das Plugin wurde 2014 grundlegend überarbeitet, damit es die üblichen Hochverfügbarkeitslösungen wie Real Application Cluster und physical Standby Datenbanken in Check_MK abbilden kann.

Bei der Entwicklung wird konsequent darauf geachtet, das alle Editionen und Versionen ab Oracle 10.1 unterstützt werden. Es werden automatisch nur die Checks erkannt, die ab der eingesetzten Version verfügbar sind.

mk_oracle Plugin - technische Details

Das Plugin bietet Innovationen, die im Umfeld von nagiosbasierten Lösungen einmalig sind. Das Plugin liefert mit 1 Aufruf alle notwendigen Informationen für alle Checks aller Datenbanken. Es wird hier konsequent zwischen synchronen und asynchronen Checks unterschieden. Das heißt, das ein synchroner Check mit jedem Aufruf aktuelle Informationen

einer Instance liefert, während der asynchrone Check im Hintergrund ausgeführt wird, sein Ergebnis zwischen speichert und nach einer definierbaren Zeit erst neu ermittelt wird. So wird sicher gestellt, dass wichtige Instanceinformationen immer aktuell abgefragt werden, während der Tablespacecheck nur alle 10 Minuten aktualisiert wird, weil das zugehörige SQL minütlich eine viel zu große Last verursachen würde. Dabei werden alle synchronen und asynchronen Checks jeweils nur mit 1 Login ausgeführt, was die Last in der Datenbank spürbar reduziert. Da der Tablespacecheck alle Informationen aller Datendateien liefert, ist nur 1 SQL notwendig, um alle Tablespaces einer Datenbank zu überwachen. Die Auswertung der Ergebnisse erfolgt auf dem Monitoringhost, wo zudem die Regelwerke angewendet werden.

Im Kern ist der Instancecheck der mit Abstand wichtigste Check. Er prüft neben dem Login in die Instance noch weitere Parameter wie ‚FORCE LOGGING‘, ‚ARCHIVELOGMODE‘ und ‚Uptime der Instance‘. Alle weiteren Checks wie ‚Tablespaceauslastung‘, ‚RMAN Backups‘, ‚Prozesslimit‘, ‚Fast Recovery Area‘, u.v.m hängen vom Instancecheck ab. Das heißt, dass bei fehlerhaftem Login der Instancecheck in einen entsprechenden Status geht und die anderen Checks nicht weiter berücksichtigt werden. Das sorgt dafür, dass am Ende nur der eigentliche Loginfehler einen Alarm erzeugt, ohne eine Lawine von Alarmen zu verursachen, wie es in vielen Monitoringlösungen zu beobachten ist.

Neben den zahlreichen Checkmöglichkeiten wurde auch einiges im Bereich Security berücksichtigt. Die Nutzung des Oracle Wallets ist möglich, um das Hinterlegen von Klartextkennwörtern in der Konfigurationsdatei zu vermeiden. Darüber hinaus kann mit Hilfe der Konfigurationsdatei das Plugin individuell gesteuert werden, wenn bestimmte Checks nicht erwünscht sind oder bestimmte Einstellungen für den Login in eine Instance erforderlich sind.

Der Anfang 2015 eingeführte Remotemodus schafft die Möglichkeit, mk_oracle plattformunabhängig nutzen zu können, da es in diesem Modus nicht mehr auf dem zu überwachenden Datenbankserver ausgeführt werden muss. So ist es möglich, Oracle auf HP/UX zu überwachen, ohne bash installieren zu müssen. Dieser Mechanismus erfordert lediglich einen Host mit Check_MK-Agenten mit Oracle Instantclient und mk_oracle-Plugin. Hier sind, mit Ausnahme der automatischen Erkennung von neuen Instanzen, alle Funktionen uneingeschränkt nutzbar.

Fazit

Das Check_MK Monitoring System hat sich im Laufe der Jahre zu einer vollständigen Monitoringlösung entwickelt, die sich durch einfache und schnelle Installation auszeichnet und über das Webinterface intuitiv administrierbar ist.

Bei OPITZ CONSULTING hat sich die Lösung bewährt und die Überarbeitung hat sich in der Praxis als richtig heraus gestellt. Zudem wurden auf Kundenwunsch seitens Mathias Kettner

noch zusätzliche Checks implementiert. Für die Zukunft sind Erweiterungen noch ausstehend – es wird also spannend bleiben und stetig weiter entwickelt.

Zum Test kann jederzeit eine Demoversion mit allen Möglichkeiten oder die Community-Edition auf der Homepage (www.mathias-kettner.de) herunter geladen werden. Letztere kann uneingeschränkt genutzt werden.

Für den produktiven Einsatz ist der Abschluss eines Supportvertrages zu empfehlen, der im Vergleich zu kommerziellen Monitoringlösungen sehr preiswert ist.

OPITZ CONSULTING setzt das Check_MK Monitoring System im Rahmen von Managed Services bei zahlreichen Kunden erfolgreich ein. Über die Partnerschaft mit Mathias Kettner erfolgt die Pflege und Weiterentwicklung des mk_oracle-Plugins zur Überwachung von Oracle Datenbanken.

Der Sprecher: Thorsten Bruhns



Thorsten Bruhns ist als Solution Architect bei OPITZ CONSULTING (Oracle Platinum Partner) beschäftigt. Er hat über 15 Jahre Erfahrung im Bereich Entwicklung, Betrieb und Aufbau von Oracle-Datenbankumgebungen mit den Schwerpunkten Hochverfügbarkeit, Performance-Tuning und Design von Virtualisierungslösungen auf Basis VMware und Oracle VM.

Er ist Architekt für die bei OPITZ CONSULTING eingesetzte Monitoringslösung Check_MK im Bereich Managed Services Infrastructure. Zudem ist er maßgeblich an der Weiterentwicklung des Oracle-Plugins beteiligt.

Kontakt: thorsten.bruhns@opitz-consulting.com

Freitag, 18. Dezember 2015, 14:30 - 15:15 Uhr

Andreas Jordan:

Flashback - Warum Oracle Zeitreisen anbieten kann, der Microsoft SQL-Server aber leider nicht

Vordergründig stellt der Vortrag die Funktionalität „Flashback-Query“ von Oracle vor, die der MS SQL Server so nicht bietet. Doch eigentlich geht es um die unterschiedliche Arbeitsweise der beiden Datenbankmanagementsysteme, die diese Funktionalität erst ermöglicht bzw. nicht ermöglicht. Dabei werden unter anderem die Begriffe Lesekonsistenz und Isolationslevel gestreift.

Wie sahen die Daten gestern aus?

Der Blick zurück wird bei Oracle durch die Erweiterung der Abfrage um den „AS OF“-Zusatz realisiert:

```
SELECT * FROM mitarbeiter AS OF TIMESTAMP SYSDATE - 1;
```

Mit dieser Abfrage wird der Stand der Tabelle wiedergegeben, wie er 24 Stunden vor Beginn der Abfrage gültig war. Schon an dieser Stelle sei erwähnt: Das klappt natürlich nicht in jedem Fall. Es müssen bestimmte Voraussetzungen erfüllt sein, die im Folgenden beleuchtet werden.

Ablauf einer Transaktion

Bei einer Transaktion gilt das „Alles oder Nichts“-Prinzip. Es bedeutet, dass alle im Rahmen einer Transaktion durchgeführten Änderungen entweder vollständig umgesetzt werden oder aber komplett rückgängig gemacht werden. Die beiden wichtigsten Aspekte beim Ablauf einer Transaktion sind „den Weg zurück sichern“ und „die Änderung garantieren“.

Um den Weg zurück zu sichern, wird zu jeder Datenänderung die entgegengesetzte Datenänderung generiert und gesichert. Bei Oracle spricht man vielfach vom „Before Image“, generell von Undo-Informationen. Um die Änderungen nach der Bestätigung des Commit zu garantieren, wird vor dem Versenden dieser Bestätigung sicher gestellt, dass auch alle Datenänderungen selbst gesichert wurden. Bei Oracle spricht man auch vom „After Image“, generell von Redo-Informationen.

Von der Verarbeitung dieser Informationen völlig unabhängig erfolgt die Übertragung der geänderten Daten aus dem Buffer Pool in die Datendatei zu einem anderen (meist später liegenden) Zeitpunkt. Diese Verarbeitung ist für die folgenden Überlegungen nicht relevant und wird daher nicht weiter betrachtet.

Technische Umsetzung

Beim MS SQL Server werden die Informationen zunächst im Hauptspeicher, genauer gesagt im Log Buffer gehalten. Dieser mit wenigen MB relativ kleine Bereich enthält die Redo- und

die Undo-Informationen in Form von Statements. Er wird in kurzen Intervallen, auf jeden Fall jedoch beim Commit, auf die Festplatte geschrieben. Dort landen die Informationen im Transaktionsprotokoll, einer Datei, die mitunter relativ groß werden kann. Denn je nach Konfiguration der Datenbank werden die einzelnen Bereiche dieser Datei (Virtual Log Files) erst dann zur Wiederverwendung freigegeben, wenn die Informationen durch ein Backup (genauer gesagt durch eine Transaktionsprotokollsicherung) verarbeitet wurden.

Auch Oracle besitzt einen Log Buffer mit der gleichen Funktionalität wie der MS SQL Server. Allerdings werden die Undo- und Redo-Informationen im Folgenden nicht gemeinsam verarbeitet. So besitzt Oracle eine zusätzliche Komponente: Den Undo Tablespace. Dieser enthält die Daten-Werte aus den Undo-Informationen, also die „Before Images“. Da es sich um einen Tablespace handelt, liegen die Informationen sowohl im Hauptspeicher als auch auf der Festplatte vor, alle Änderungen werden ebenfalls durch Redo-Informationen gesichert. Damit liegen diese Informationen auch nach dem Ende der Transaktion und sogar nach einem Neustart der Instanz weiterhin vor.

Die Redo-Informationen werden bei Oracle, ähnlich wie beim MS SQL Server, in Redo Log Files geschrieben. Oracle verwendet dabei mindestens zwei Dateien mit einer festen Größe von mindestens 4 MB. Ist die aktive Datei voll, so wird diese geschlossen und die andere Datei zur aktiven Datei. Die geschlossenen Dateien werden vom Archiver in die Archive Destination kopiert und stehen dort für die Wiederherstellung der Datenbank zur Verfügung.

Wichtigster Unterschied zwischen den beiden Systemen und der Schlüssel zu Flashback-Query sind die längerfristig zugänglichen Undo-Informationen bei Oracle.

Lesekonsistenz

Diese Undo-Informationen werden bei Oracle dazu genutzt, lesenden Abfragen einen konsistenten Stand der Informationen zum Zeitpunkt des Beginns der Abfrage zur Verfügung zu stellen. Dabei werden diese nicht von ändernden Zugriffen blockiert, da ältere Stände der Informationen aus den Undo-Informationen rekonstruiert werden. Jedenfalls so lange, wie der Undo Tablespace groß genug ist, um alle benötigten Informationen aufzunehmen.

Beim MS SQL Server richtet sich das Verhalten nach dem verwendeten Isolationslevel. Beim Standard „Read Committed“ erzeugen ändernde Abfragen Sperren auf den betroffenen Datensätzen, lesende Zugriffe werden dadurch blockiert und müssen auf das Ende der schreibenden Transaktion warten.

Die alternativen Isolationslevel „Snapshot“ und „Read Committed Snapshot“ verändern das Verhalten und ermöglichen wie bei Oracle einen lesekonsistenten gleichzeitigen Zugriff. Erreicht wird dies durch den Aufbau von Zeilen-Versionen in der tempdb. Die Aktivierung dieser Isolationslevel ändert jedoch den Aufbau der Daten in den Dateien, da in jeden Datensatz ein zusätzlicher 14 Bytes großer Zeiger zu den Zeilen-Versionen eingefügt wird.

Ein wichtiger Unterschied zu Oracle bleibt jedoch: Die Zeilen-Versionen sind nur bis zum Neustart der Instanz verfügbar, ein Bereinigungsprozess sorgt auch schon vorher für die Entfernung nicht mehr von laufenden Transaktionen benötigter Zeilen-Versionen.

Flashback-Query

Warum ist die Dauer der Speicherung von Undo-Informationen so wichtig? Weil Flashback-Query nichts anderes ist, als eine konsequente Erweiterung der Lesekonsistenz in die Vergangenheit.

Mit Hilfe der Undo-Informationen kann der Zustand der Daten nicht nur zum Zeitpunkt des Beginns der Abfrage, sondern auch zu anderen Zeitpunkten in der Vergangenheit bereitgestellt werden. Voraussetzung ist natürlich, dass diese Informationen noch im Undo Tablespace vorliegen.

Konfigurationsparameter

Zwei Konfigurationsparameter haben direkten Einfluss auf die Verfügbarkeit von Flashback-Query. Zum einen die Option `UNDO_RETENTION` für das System. Diese Angabe in Sekunden bestimmt, wie lange die Undo-Daten mindestens nach Ende der Transaktion noch vorgehalten werden sollen. Diese Angabe wird jedoch nur für automatisch vergrößernde Undo Tablespaces beachtet und auch nur, solange `MAXSIZE` noch nicht erreicht ist.

Zum anderen kann die Option `RETENTION GUARANTEE` für den Undo Tablespace die unbedingte Einhaltung der `UNDO_RETENTION` garantieren. Dies kann jedoch zu Abbrüchen von schreibenden Transaktionen führen, wenn nicht mehr genug Platz vorhanden ist.

Fazit

Die getrennte und persistente Speicherung der Undo-Daten ist der Schlüssel zur Zeitreise. Anhand dieses Features zeigt sich auch, welche Auswirkung die unterschiedliche Architektur der Datenbankmanagementsysteme auf die Verarbeitung von Abfragen hat.

Noch ein abschließender Hinweis: In Produktionsumgebungen wird Flashback-Query sicherlich sehr selten eingesetzt, aber für Entwicklung und Tests kann das Feature durchaus sinnvoll eingesetzt werden.

Der Sprecher: Andreas Jordan



Andreas Jordan ist Senior Chief Consultant bei der ORDIX AG. Seit über 10 Jahren berät er Kunden bei der Konfiguration und Optimierung von Datenbanksystemen. Dabei ist er sowohl mit dem Microsoft SQL Server als auch mit Oracle bestens vertraut. Zusätzlich zu seiner Beratertätigkeit ist Andreas Jordan auch als Referent im ORDIX-Seminarzentrum tätig.

Kontakt: Andreas.Jordan@ordix.de

Unsere Sponsoren



DataStax stellt Apache Cassandra™ in einer Datenbankplattform bereit, die speziell für die Leistungs- und Verfügbarkeitsanforderungen von Web-, Mobil- und IoT-Anwendungen ausgelegt ist. Damit erhalten Unternehmen eine sichere „always-on“ Datenbank, die sich einfach betreiben lässt – egal, ob sie in einem einzigen Rechenzentrum skaliert wird oder über verschiedene Rechenzentren und Clouds hinweg.

Über 500 Kunden in mehr als 50 Ländern vertrauen auf die Datenbanktechnologie von DataStax – darunter einige der weltweit innovativsten Firmen wie Netflix, Adobe, Intuit und eBay. DataStax mit Sitz in Santa Clara, Kalifornien/USA, wird von branchenführenden Investoren unterstützt wie Comcast Ventures, Crosslink Capital, Lightspeed Venture Partners, Kleiner Perkins Caufield & Byers, Meritech Capital, Premji Invest und Scale Venture Partners. Erfahren Sie mehr unter DataStax.com oder indem Sie uns unter [@DataStax](https://twitter.com/DataStax) folgen.

Über Apache Cassandra

Apache Cassandra™ ist ein verteiltes Datenbankmanagementsystem, das entwickelt wurde, um sehr große Datenmengen über viele Standardserver hinweg zu verwalten. Dadurch gewährleistet das Open-Source-System Hochverfügbarkeit und vermeidet einen Single Point of Failure. Cassandra unterstützt zuverlässig rechenzentrenumspannende Cluster mittels asynchroner Replikation ganz ohne Master – das sorgt für niedrige Latenzzeiten für alle Clients.

Hauptmerkmale

- Verteilte NoSQL Datenbank-Technologie
- „Always-on“ Datenbankarchitektur
- Horizontal skalierbare Datenbank
- Big Data
- Unternehmenssicherheit
- Echtzeitanalysen
- In-Memory Computing
- Lokal oder in der Cloud einsetzbar

Kontakt:

DataStax Deutschland GmbH

Thomas Gregg

Enterprise Sales Manager

E-Mail: thomas.gregg@datastax.com

Telefon: +49 173 54 73 452

Weitere Informationen erhalten Sie unter www.datastax.com oder indem Sie uns unter [@DataStax](https://twitter.com/DataStax) folgen.

dbaservices®

ADMINISTRATING BUSINESS DATABASES 24x7

Im Jahre 2001 wurde im hessischen Neu-Isenburg der Grundstein für die dbaservices® GmbH (ehemals allnabout GmbH) gelegt. Als spezialisiertes

Dienstleistungsunternehmen in der Informationstechnologie betreuen wir europaweit zahlreiche ORACLE und MSSQL Datenbanken. Die Besonderheit unserer ITIL basierten Dienstleistungen ist die Bereitstellungen von standardisierten und individuellen Managed Database Outtasking Modelle. Darunter befinden sich 24/7 Service Level Agreements, die die Vitalität und Verfügbarkeit von produktionsrelevanten Datenbanken rund um die Uhr gewährleisten. Ein weiterer Schwerpunkt unserer Tätigkeiten liegt in der Beratung, Skizzierung und der Implementierung von Datenbanksystemen. Dazu gesellen sich datenbanktypische Projekte hinsichtlich Datenbankmigration, -Upgrade, -Backup, -Performance Tuning und -Sicherheit.

Nationale und internationale Kunden, unter Anderem aus der Automobilzulieferindustrie, Versicherung, Logistik und Energie mit hochkritischen Datenbanksystemen, schätzen zum Teil bereits seit über 10 Jahren, unsere hohe Verlässlichkeit und Qualität.

2014 entschied sich die Geschäftsleitung den Namen dbaservices® GmbH anzunehmen. Der Name ist eindeutig und entspricht genau den Dienstleistungen, die dbaservices® anbietet. Seit Kurzem steht „Database as a Service“ www.dbaas.de als neuer Dienst zu Verfügung, der eine flexible Oracle Datenbanklizenznutzung ermöglicht.

Kernkompetenzen:

- Beratung und Implementierung
- 24x7 Monitoring und Bereitschaft
- ORACLE und MS SQL Datenbanksysteme
- 3rd Level Support
- Managed Database Outtasking Modelle
- Oracle Database as a Service

Kontakt:

Parul M. Mohan, CEO
dbaservices® GmbH
Hermannstr.54
63263 Neu-Isenburg

Telefon: +49 (0) 6102-88 27 68 - 11
E-Mail: parul.mohan@dbaservices.de
www.dbaservices.de

HANSER

Aktuelle, praxisgeprüfte und unmittelbar einsetzbare Fachinformationen für angehende und erfahrende IT-Profis – das bietet Ihnen der Carl Hanser Verlag mit seinen Computerbüchern. So lösen Sie Ihre Aufgaben und Probleme der Praxis sicher und schnell und arbeiten sich rasch und fundiert in neue Themen ein. Werden Sie beruflich erfolgreicher – mit Hanser Computerbüchern.

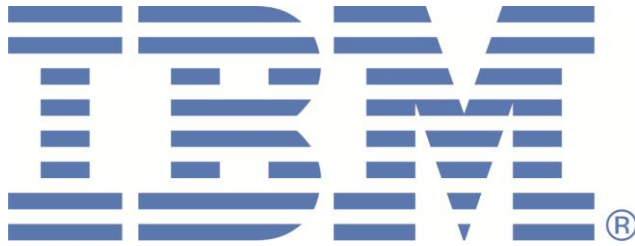
Das Computerbuchprogramm beinhaltet die Themen Programmierung, Online und Mobil, Softwareentwicklung, IT- und Projektmanagement, Windows Server System, Oracle und IT-Grundlagen. Egal was Sie programmieren, ob Java, responsive Webdesign, C++ oder Python. Im Hanser Online-Shop finden Sie zum Thema Computer für jedes Bedürfnis das richtige Fachbuch, als gedrucktes Buch oder als E-Book! Mehr dazu unter www.hanser-fachbuch.de/computer.

Kontakt:

Carl Hanser Verlag GmbH & Co. KG
Kolbergerstraße 22
81679 München

Telefon: +49 89 99830-0
Fax: +49 89 984809
E-Mail: info@hanser.de

www.hanser-fachbuch.de



und Non-Profit-Organisationen IT-Lösungen für ihre Herausforderungen zu entwickeln.

IBM ist ein global integriertes Technologie- und Beratungsunternehmen mit Sitz in Armonk, New York. Mit Niederlassungen in mehr als 170 Ländern ist IBM der Technologie- und Transformationspartner, um gemeinsam mit Unternehmen, Regierungen

Innovation bildet dabei den Kern der IBM Strategie. Das Lösungsportfolio bietet ein breites Spektrum von Consulting-, Cloud- und IT-Services über die zugehörigen intelligenten IT-Infrastrukturen aus Software und Hardware sowie Finanzierung.

Für Unternehmen aller Größen wird die IT zu einem zentralen Element, um die Transformation ihrer Geschäftsmodelle voranzutreiben und die Chancen durch die zunehmende Globalisierung der Märkte erfolgreich zu nutzen. Daher konzentriert sich IBM heute auf mehrere Wachstumsinitiativen – Business Analytics, Cloud Computing, Mobile Enterprise, Social Computing, geografische Wachstumsmärkte und die Smarter-Planet-Strategie.

Mitarbeiterinnen und Mitarbeiter von IBM und Kunden rund um die Welt arbeiten eng zusammen, um die Expertise in den Bereichen Forschung und Entwicklung, Technologie sowie Beratung so einzusetzen, dass Unternehmen dynamischer und effizienter agieren können. Auch Städte und Volkswirtschaften können durch umfassende Weiterentwicklungen ihrer Infrastruktur, beispielsweise bei Fragen rund um Energie, Transport, Administration, Medizin und Ausbildung, intelligenter werden.

Mit einem auf die Kunden und deren Wertschöpfung zugeschnittenen Vertrieb bündelt das Unternehmen seine Industrie- und Lösungsexpertise, wobei die Geschäftsbereiche regional und nach Branchen strukturiert sind. So profitieren Kunden von spezialisierten Teams mit umfassendem Branchenwissen und einer durchgängigen Betreuung durch Ansprechpartner direkt vor Ort.

Seit der Gründung im Jahr 1911 hat sich IBM immer wieder neu definiert und ist durch weit über Technologien hinausgehende Innovationen und einem Jahresumsatz von 92.8 Milliarden US-Dollar im Jahr 2014 zu einer der stärksten Marken der Welt aufgestiegen. Jährlich investiert das Unternehmen mehr als sechs Milliarden US-Dollar in Forschung und Entwicklung. Die IBM Deutschland Research & Development GmbH in Böblingen ist eines der größten Technologiezentren der IBM weltweit.

IBM hat in 2014 die Rekordzahl von 7.534 US-Patenten erteilt bekommen – im Schnitt erhielten die IBM Erfinder mehr als 20 Patente pro Tag. Das Unternehmen belegt damit zum 22. Mal in Folge den Spitzenplatz in der Rangliste des US-Patentamts. Mehr als 400 dieser Erfindungen stammen aus dem deutschen IBM Entwicklungszentrum in Böblingen sowie dem Forschungszentrum der IBM in Rüschlikon bei Zürich.

Mit ihrem umfassenden Lösungsangebot bietet IBM Mitarbeiterinnen und Mitarbeitern vielfältige Einsatz- und Karrieremöglichkeiten im In- und Ausland, sowie die Mitarbeit in internationalen Projektteams. Unter dem Stichwort "Employability" setzt IBM auf kontinuierliche Aus- und Weiterbildung – IBM investiert in diesem Bereich weltweit jährlich rund 500 Millionen US-Dollar.

Kontakt:

IBM Deutschland GmbH

IBM-Allee 1

71139 Ehningen

Telefon: +49 800 225 5426

E-Mail: halloibm@de.ibm.com

www.ibm.com/de



informatica
Put potential to work.™

Informatica Corporation (Nasdaq: INFA) ist weltweit führender Anbieter von Datenintegrationssoftware. Unternehmen auf der ganzen Welt setzen auf Informatica, um das Potenzial ihrer Informationen zu erschließen und wichtige Geschäftsnotwendigkeiten zu verwirklichen. Informatica Vibe, die

branchenweit erste und einzige integrierbare virtuelle Datenmaschine (VDM), bildet die Grundlage für die unvergleichlichen „Map Once. Deploy Anywhere.“-Funktionen der Informatica Plattform. Weltweit nutzen über 5.800 Unternehmen Informatica, um den Wert ihrer sich vor Ort, in der Cloud oder in sozialen Netzwerken befindlichen Informationsbestände auf Geräten, Mobilgeräten, in sozialen Netzwerken oder Big-Data-Systemen optimal zu nutzen.

Weitere Informationen:

<https://www.informatica.com/data-integration-magic-quadrant.html>

<https://www.informatica.com/products/informatica-platform.html>

<https://www.informatica.com/products/informatica-platform/vibe.html>

<https://www.informatica.com/cloud.html>

Kontakt:

Informatica GmbH
Wolfgang Epting
Product Specialist

Telefon: 0711 – 13984 0

E-Mail: wepting@informatica.com

www.informatica.com



MarkLogic ist eine neue Generation von Datenbank, die Unternehmensdaten in wertvolle und gewinnbringende Informationen verwandelt. MarkLogic ist die einzige Enterprise NoSQL-Datenbank.

Namhafte Unternehmen und Behörden vertrauen auf MarkLogic. Die Plattform ist hoch performant, skalierbar, flexibel und bietet höchste Sicherheitsstandards. Mit MarkLogic bewältigen Kunden Herausforderungen, die durch komplexe, ständig wechselnde heterogene Datenstrukturen entstehen.

Durch MarkLogic erhalten Kunden mehr Erkenntnisse über ihre Daten und können die Entwicklung geschäftskritischer Anwendungen beschleunigen. Integrierte Suchfunktionen und Application Services ermöglichen es innerhalb kurzer Zeit Datenbankanwendungen zu entwickeln. So erhalten Kunden nicht nur einen schnellen Zugriff auf große Datenmengen sondern gewinnen binnen kürzester Zeit neue Erkenntnisse daraus.

Neue Möglichkeiten für Ihre Daten mit Enterprise NoSQL (Beyond Relational):

<http://de.marklogic.com/resources/beyond-relational-german/>

Kontakt:

MarkLogic (Deutschland) GmbH
Theatinerstraße 11
80333 München

Telefon: 089 – 71042-2151

E-Mail: info@marklogic.de

www.marklogic.de



Die syntegris information solutions GmbH bietet umfassende Unterstützung bei der Planung, Entwicklung, Implementierung und Optimierung von Softwarelösungen. Dies gilt für das gesamte Spektrum, angefangen bei

der fachlichen Konzeption, dem Prozessdesign über die Entwicklung und Modernisierung der Businesslogik mit JEE oder APEX, bis zum produktiven Betrieb der Anwendung, der Administration und dem Tuning von Datenbanken. Wir erstellen komplette Business Universen, DWH-Architekturen oder einzelne Data Marts für Ihre systemübergreifende betriebliche Steuerung und das Reporting. Wir konzipieren und erstellen für unsere Kunden individuelle maßgeschneiderte Lösungen. Syntegris ist seit der Gründung Oracle Partner und mittlerweile auch Oracle Gold Partner Database Specialized.

Kontakt:

syntegris information solutions GmbH
Frau Sonja Schuster
Hermannstraße 54-56
63263 Neu-Isenburg

Telefon +49 6102 298668
E-Mail: sonja.schuster@syntegris.de

www.syntegris.de