



TOOLGESTÜTZTE ORACLE PERFORMANCEANALYSE AUS CONSULTANT-SICHT

DOAG Konferenz 2011, Nürnberg

Marcus Mönnig

Lichtblick AG, Hamburg

Vorstellung

Marcus Mönnig, Informatiker (B.Sc.), OCP 10/11

E-Mail: mm@marcusmoennig.de

Internet: <http://marcusmonnig.wordpress.com/>

Berufliche Stationen der letzten 15 Jahre

Selbständiger Softwareentwickler

Angestellter Softwareentwickler

Oracle Consultant / DBA für ein mittelständiges
Softwarehaus (Software für die pharmazeutische Industrie)

Heute: DBA Lichtblick AG (Ökologischer Energieversorger)

Vorstellung „Mumbai“

What is Mumbai?

Mumbai is a freeware Windows application targeted at Oracle DBAs and consultants, with a special focus on performance analysis

Features

- Instant access to important database views vital for DBAs and consultants
- Retrieved data in Mumbai can be sliced and diced in various ways (sorting, filtering, grouping, group arithmetic, etc.)
- Retrieve alert.log and trace files from the database server to Mumbai and process/analyze them there.
- Start and stop 10046 traces for any session or process and trace the statements from your SQL console session with just a few clicks
- Powerful 10046 trace file viewer if you need to dig into the details of a TKPROF or OraSRP report. OraSRP reports can be generated from within Mumbai.
- Heap dump analyzer that aggregates data from trace files
- ...

Agenda

- Wie ermittle ich den „richtigen“ Ausführungsplan?
- Was passiert gerade auf der Datenbank?
- Was passierte in der Vergangenheit auf der Datenbank?

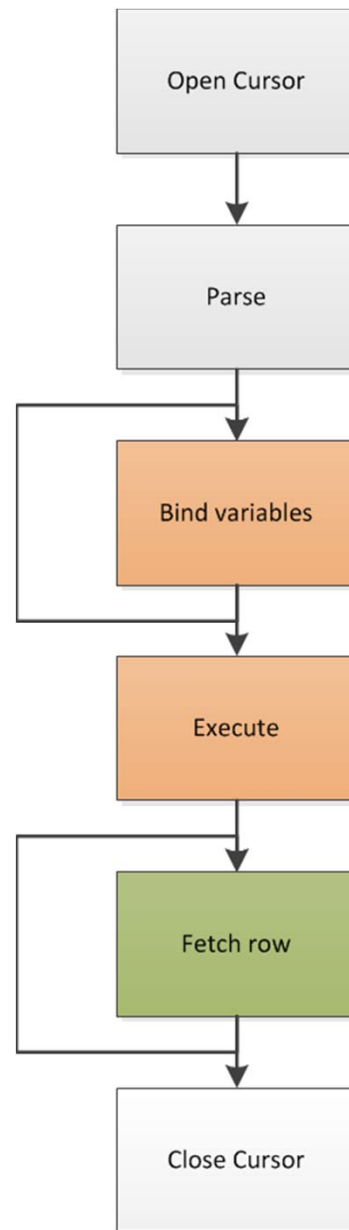
Möglichkeiten sich einen Ausführungsplan anzeigen zu lassen

- SQL Kommando „EXPLAIN PLAN“ und SELECT auf Plan table
- Oracle Package DBMS_XPLAN zur Anzeige von Ausführungsplänen aus folgenden Quellen:
 - Plan table (DISPLAY)
 - Plan aus AWR Daten (DISPLAY_AWR)
 - Plan der noch im Shared-Pool vorhanden ist (DISPLAY_CURSOR)
 - Plan einer SQL Plan Baseline (DISPLAY_SQL_PLAN_BASELINE)
 - Plan für ein SQL Tuning Set (DISPLAY_SQLSET)

Alternative zu EXPLAIN PLAN

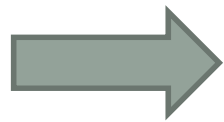
- AD-hoc Generierung von Ausführungsplänen durch
 1. Parsing des SQL-Statements
 2. Execution des SQL-Statements
 3. Optional: Fetch der Ergebnis-Rows
 4. Anzeige des tatsächlich genutzten Plans durch
DBMS_XPLAN.DISPLAY_CURSOR

Lifetime of a cursor



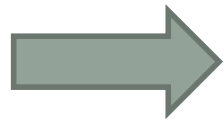
Was stimmt denn nicht mit EXPLAIN PLAN?

- Keine BIND & EXECUTE Phase



EXPLAIN PLAN berücksichtigt keine Bindevariablen (bezüglich Wert & Typ)

- Keine FETCH Phase



Keine Ausführungsstatistik die anzeigt ob die Annahmen des Optimizers eintrafen



Demo 1

Demo 1 zur Ausführungsplänen

Ausführungspläne – Zusammenfassung

- EXPLAIN PLAN sollte bei SQL Statements mit Bindevariablen nicht verwendet werden
- PARSE, BIND, EXECUTE, DBMS_XPLAN.DISPLAY_CURSOR als Alternative ist immer sinnvoll
- PARSE, BIND, EXECUTE, **FETCH**, DBMS_XPLAN.DISPLAY_CURSOR wegen der zusätzlichen Ausführungsstatistiken sinnvoll zum weiteren Tuning (wenn Ausführungszeit und –last des SQL Statements es möglich machen)

Agenda

- Wie ermittle ich den „richtigen“ Ausführungsplan?
- **Was passiert gerade auf der Datenbank?**
- Was passierte in der Vergangenheit auf der Datenbank?

Was passiert gerade auf der Datenbank?

- **Verfügbare Tools**
 - AWR/ASH ggf. über Oracle Enterprise Manager/DB console
 - TOAD, Spotlight on Oracle, ...
- **Problem aus Consultant-Sicht mit diesen Tools**
 - Verfügbarkeit
 - Lizenzierung
 - Installation, Konfiguration
 - Keine Zeit diese Probleme auszuräumen
- **Lösung: Session Snapper SQL Skript von Tanel Poder**
 - Keine Lizenzierung, keine Installation, keine Änderung in der DB

Session Snapper Skript

Merkmale von Snapper

- SQL*PLUS Skript
- Großer, anonymer PL/SQL Block (keine Installation, Änderung der DB)
- Kompatibel mit Oracle 9.2 bis 11.2
- Vereinfacht: Ad-hoc Version AWR/ASH für kurze Zeitintervalle ohne historische Datenhaltung
- Setzt nur einen Benutzer mit SELECT_CATALOG_ROLE voraus
- Ermittelt Deltas für Performancewerte aus V\$ Views für einen definierten Zeitraum (einige Sekunden)
 - Session Statistics aus V\$SESSTAT
 - Session Time model info aus V\$SESS_TIME_MODEL
 - Session Wait statistics aus V\$SESSION_EVENT und V\$SESSION_WAIT
- Nachbildung der ASH-Funktionalität (Sampling aus V\$SESSION)



Demo 2

Demo 2 zu Snapper

Zusammenfassung zu Snapper

- Schnell einsetzbar
- Übergreifende Lösung bezüglich DB-Version und Edition
- Mächtig und flexibel einsetzbar

- Graphische Aufbereitung in Mumbai erleichtert die Analyse
- Die volle Flexibilität ist aber nur über den manuellen Aufruf nutzbar

Agenda

- Wie ermittle ich den „richtigen“ Ausführungsplan?
- Was passiert gerade auf der Datenbank?
- **Was passierte in der Vergangenheit auf der Datenbank?**

Was passierte in der Vergangenheit auf der Datenbank?

Existierende Lösungen:

- Automatic Workload Repository (AWR)
 - nur für Enterprise Edition mit Diagnostic Pack Lizenz
 - Bietet textuelle Reports und graphische Darstellung über EM
- Statspack
 - Seit langem und immer noch verfügbar
 - Textuelle Reports, leider keine graphische Darstellung

Textuelle Reports über AWR und Statspack-Daten

Differenzbildung von Performancedaten zwischen Start- und Endsnapshot

```
STATSPACK report for

Database      DB Id      Instance      Inst Num  Startup Time  Release      RAC
~~~~~
                2118755765 VALDB                1 18-Jun-11 19:30 10.2.0.4.0 NO

...

Snapshot      Snap Id      Snap Time      Sessions  Curs/Sess  Comment
~~~~~
Begin Snap:      58140 05-Jul-11 00:00:00      79      4.0
End Snap:      58229 05-Jul-11 17:00:00     356     16.4
  Elapsed:      1,020.00 (mins) Av Act Sess:      1.7
  DB time:      1,685.10 (mins)      DB CPU:      1,354.56 (mins)

...
```


Problem der Durchschnittsbildung

Beispiel:

Was passierte in der letzten Woche auf der DB?

(SP-Snapshots alle 30 Min. → 336 Snapshots pro Woche)

a.) Ein Report zwischen Snapshot 1 und 336?

„*You can't extrapolate detail from aggregate*“
(Cary Millsap) → Detailinformation geht verloren

b.) 335 Reports (für jedes Halbstundeintervall)?

Wer soll die alle lesen und auch noch die zeitlichen Entwicklungen überschauen?

Lösung: graphische Darstellung

Für AWR-Daten im Oracle EM zum Teil verfügbar

Für SP-Daten nicht verfügbar

Auswertung von AWR/Statspack Daten in Mumbai

- Erstellung von tabellarischen Übersichten und graphischen Diagrammen über den Analysezeitraum für folgende Basisdaten
 - Parameteränderungen (nur tabellarisch)
 - Daten aus V\$SYSSTAT über den „Instance Activity Report“
 - Wait events und CPU-Zeit
 - Average Active Session (AAS)
 - Time Model Statistics
 - OS Statistics
 - Top SQL nach Elapsed time, CPU time, Buffer gets und Anzahl von Ausführungen
- Diagramme zeigen die zeitliche Entwicklung eines Problems und geben Hinweise für welche Zeiträume die Erstellung einzelner Reports sinnvoll ist



Demo 3

Demo 3 zur Auswertung von Statspack/AWR Daten in Mumbai

Danke für Ihr Interesse!

- Marcus Mönnig's Oracle and Mumbai Blog

<http://marcusmoennig.wordpress.com>

- Tanel Poder's Session Snapper

<http://tech.e2sn.com/oracle-scripts-and-tools/session-snapper>

- Kontaktdaten

Marcus Mönnig

E-Mail:

mm@marcusmoennig.de

Internet:

<http://marcusmoennig.wordpress.com/>