

GNC Global Network Communication GmbH
GNC @ nussdorfer laende 23, 1190 vienna
T +43 664 246 5008, cp@gnc.at



TOM Version 1.3

Effizienz an Hand von Beispielen

Christian Pfundtner

Vortragender



Christian Pfundtner
cp@gnc.at

ORACLE

10g Certified Master

Senior Consultant
Oracle Certified Master DBA
Oracle Certified RAC Expert
Oracle ACE



- 👤 Wer oder was ist TOM?
 - 2 Slides about TOM
 - 👤 Was bringt TOM 1.3 an Detailverbesserungen?
 - 👤 Beispiel von der Installation bis zur effektiven Nutzung
-



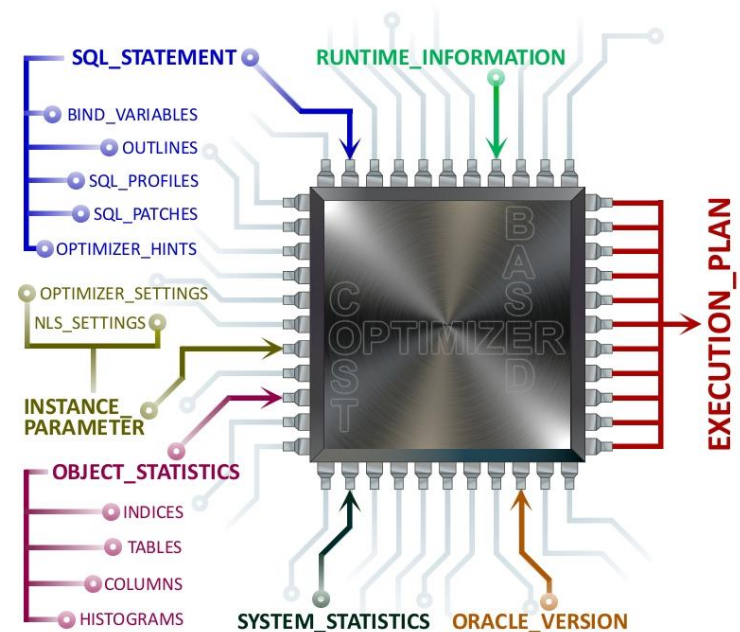
Wer oder was ist TOM?

- ☛ TOM ist eine GNC-Eigenentwicklung zur Überwachung des CBO.
 - ☛ TOM liefert umfassende Informationen zu SQL Statements, deren Execution Plans und alle anderen optimizer-relevanten Statistiken.
 - ☛ TOM alerted, wenn ein SQL Statement plötzlich langsamer wird, wobei in den meisten Fällen auch die potentiellen Gründe dafür genannt werden.
 - ☛ TOM zeigt auf, welche SQL Statements bis auf Literale und Konstanten ident sind, aber trotzdem mehrfach existieren.
-



Warum benütze ich TOM?

- weil der CBO von vielen Faktoren beeinflusst wird und es bisher kein Tool gibt, das alle diese Informationen (auch historisch) zur Verfügung stellen kann
- weil es oft (fast) unmöglich ist, mit den existierenden Informationen festzustellen, welche SQL Statements langsamer geworden sind
- weil wir Feedback unserer Kunden bekommen haben, dass Sie eine Hilfestellung zur Analyse bzw. eine Lösung dieses Problems benötigen.



Overview TOM Version 1.3

Interne Optimierungen



TOM intern

- ☛ deutliche Reduktion des Platzbedarfs
 - ☛ viel schnelleres Datensammeln bei geringerem Ressourcenbedarf
 - ☛ verbesserte Unterstützung von Oracle Real Application Cluster
 - ☛ vereinfachte Installation und Konfiguration
-

Inside TOM 1.3

Reduktion des Platzbedarfs



- ☛ umfassendes Redesign der internen Strukturen
 - ☛ smarteres Erkennen von redundanten Daten
 - ☛ Einbindung von Oracle 11g/11gR2 Features
-

Inside TOM 1.3

Speedup beim Datensammeln



- intelligente Erkennung von „statischen“ Informationen
 - effektivere Zugriffe auf Oracle Memory Strukturen
-

Inside TOM 1.3

Real Application Cluster Support



- GATHER und EVALUATE nur noch auf einem Knoten
 - Reduktion des Overheads beim Datensammeln
 - nur noch ein Datensammel-Job pro RAC Datenbank
 - Aufspüren von Statements, die auf einer Instanz schneller laufen als auf der anderen
-

Überblick Beispiel



- 📁 TOM installieren – wie einfach geht das?
 - 📁 TOM konfigurieren – was sind die ersten Schritte?
 - 📁 TOM by Example
 - Datenbank Administrator
 - Applikationsverantwortlicher
 - Entwickler
-



Installing TOM

- Konsolidierung der Installationsscripts
 - Erzeugte Log Files und SQL Files mit Instanz- und Hostnamen
 - Installation auch mittels Übergabeparameter möglich
 - Oracle 11g XDB Option nicht mehr zwingend nötig
 - Einschränkungen bei den Lizenztypen
 - Einschränkungen beim Alerting (keine eMails)
-



Whats completely new in TOM 1.3

- ☛ Support für Oracle 11g und 11gR2 Features
 - Adaptive Cursor Sharing
 - Result Cache
 - ☛ Reporting direkt im Client Tool möglich (**SQLPLUS**, **TOAD**, etc.)
 - kein Zugriff auf das „**TOM_DIR**“ mehr nötig
 - ☛ Erkennung von (einigen) RBO Tuning Tricks
 - Statements melden und Empfehlungen geben
 - ☛ Objekt Wachstum Monitoring
 - welche Objekte wachsen – und wie stark
 - ☛ Erkennung von „nur durch Whitespaces“ unterschiedlichen Statements
 - unnötiges Parsen kann vermieden werden
-

Whats new in TOM 1.3

Verbesserung bestehender Funktionen



- 📌 verbesserter Real Application Cluster Support
 - Daten werden nur auf einem Node gesammelt.
 - weniger Overhead im RAC
 - 📌 verbesserte Empfehlungen für Reorganisation von Objekten
 - Vorschlag für den optimalen Weg der Reorganisation als Default
 - Alternative Wege werden ebenfalls angegeben.
 - 📌 mehr Flexibilität beim Sammeln von Statement Daten
 - selektive Sammlung von Typen (**SELECT**, **INSERT**, etc.)
 - **INSERT** Statements werden nicht mehr automatisch gesammelt.
 - 📌 smartere Erkennung von „langamer werdenden Statements“
 - Nicht nur wenn ein „exakt gleiches“ SQL Statement einen anderen Plan bekommt, TOM erkennt wenn ein „inhaltlich gleiches“ SQL Statement einen anderen Plan ausführt.
-

Configuring TOM

First Steps: Was will ich monitoren?



- ☛ Welche Schemas/Benutzer möchte ich oder möchte ich nicht monitoren?
 - Standard Oracle Systembenutzer werden default-mäßig nicht überwacht.
 - Eine gezielte Überwachung von einzelnen Applikationen (oder Datenbank Benutzern) ist möglich.
-

Configuring TOM

First Steps: Typen von Statements



Welche Typen von Statements sollen überwacht werden?

e SELECT

- Das ist definitiv ein Muss.

e UPDATE, DELETE

- Das ist in den meisten Fällen sinnvoll.

e MERGE

- Nur wenn es entsprechend benutzt wird.

e INSERT

- Ein Tuning ist nicht möglich.
(ausgenommen **insert into...as select**, den **select** Teil)
 - Wird ab TOM 1.3 standardmäßig nicht mehr überwacht.
-

Configuring TOM

First Steps: Wer kann TOM benutzen?



TOM Zugriffe regeln

DBA

- voller Zugriff auf **TOM** Schema

Entwickler, Applikationsadministrator

- Rolle: **TOM_USER_ROLE**
 - Synonym Script: entweder **PUBLIC** oder **PRIVATE**
-

Configuring TOM

First Steps: Wer erhält e-mail Alerts?



Um eine rasche Reaktion zu ermöglichen, kann das Feature e-mail Notification definiert werden.

- ☛ wird während der Installation abgefragt
 - ☛ kann jederzeit geändert werden
 - ☛ kann jederzeit aktiviert/deaktiviert werden
 - ☛ setzt ab Oracle 11g die XDB Option voraus
-

TOM by Example

Datenbank Administrator (ohne TOM)



- 🕒 **03:17h:** Der automatische Oracle „Gather New Statistics Job“ erzeugt neue Objekt Statistiken und invalidiert damit viele SQL Statements.
 - 🕒 **05:57h:** Hr. Meier (hat Schlafprobleme und kommt immer so früh) beginnt mit seiner Arbeit. SQL Statements werden neu geparsed, und einige bekommen einen neuen Execution Plan. Hr. Meier merkt zwar, dass manche Verarbeitungen langsamer sind, aber es funktioniert ja (noch) akzeptabel.
 - 🕒 **08:00h:** Es melden sich nach und nach weitere Benutzer an, und das System wird langsamer und langsamer und langsamer...
 - 🕒 **08:27h:** Der erste Benutzer versucht den DBA zu erreichen, der ist aber noch auf dem Weg ins Büro.
 - 🕒 **08:33h:** „Nichts geht mehr“. Die Antwortzeiten sind inzwischen so schlecht, dass ein normales Arbeiten nicht mehr möglich ist.
-

TOM by Example

Datenbank Administrator (ohne TOM)



- 🕒 **08:45h:** Der DBA kommt im Büro an. Seine e-mail Inbox quillt über. Das Telefon läutet unentwegt.
 - 🕒 **08:56h:** Das Login in die **dbConsole** läuft immer noch, weil der Datenbank Server zu stark überlastet ist.
 - 🕒 **09:01h:** Der DBA kann mit der Suche nach der Ursache beginnen. Doch leider hat die Firma keine Enterprise Manager Packs lizenziert.
 - 🕒 **irgendwann:** Der DBA entdeckt, dass neue Statistiken schuld daran sind.
 - 🕒 **5 Minuten später:** Er hat auf die vorhergehenden Statistiken resettet und diese gesperrt (in Summe 73 Tabellen und deren Indices).
 - 🕒 **noch 10 Minuten später:** Die Systemlast ist wieder normal, der DBA „vergisst“ den Vorfall. Es läuft ja wieder alles.
-

TOM by Example

Datenbank Administrator (ohne TOM)



- 👤 **4 Monate später:** Die Performance der Datenbank ist in den letzten Wochen immer schlechter geworden, weil die Datenmenge gestiegen ist. Es wird beschlossen, eine stärkere Hardware zu kaufen.
 - 👤 **noch 1 Monat später:** Die neue Hardware ist da, installiert, und die Datenbank ist übernommen. Die Performance wird nur mäßig besser.
 - 👤 **noch 2 Monate später:** Trotz mehrmaliger Datenbank Tuning-Versuche, wird die Performance nicht wirklich besser. Es muss wohl an der Applikation liegen.
 - 👤 **noch 1 Monat später:** Eskalation mit dem Applikationshersteller, der sagt, dass es nicht an der Applikation liegen kann. Gemeinsamer Termin um das Problem zu finden...
 - 👤 **und noch zwei Tage:** Gefunden! Die Objekt Statistiken sind massiv falsch.
-

TOM by Example

Datenbank Administrator (mit TOM)



- 🕒 **03:17h:** Der automatische Oracle „Gather New Statistics Job“ erzeugt neue Objekt Statistiken und invalidiert damit viele SQL Statements.
 - 🕒 **05:57h:** Hr. Meier (hat Schlafprobleme und kommt immer so früh) beginnt mit seiner Arbeit. SQL Statements werden neu geparsed, und einige bekommen einen neuen Execution Plan. Hr. Meier merkt zwar, dass manche Verarbeitungen langsamer sind, aber es funktioniert ja (noch) akzeptabel.
 - 🕒 **06:11h:** TOM sammelt Daten (`GATHER_ALL_DATA_JOB`)
 - 🕒 **06:17h:** TOM evaluiert die gesammelten Daten. (`GATHER_ALL_DATA_JOB`), findet, dass drei Statements langsamer sind und generiert entsprechende Alerts für den DBA.
 - 🕒 **06:18h:** TOM Alert e-mails treffen ein und werden gleichzeitig via SMS Gateway an den DBA geschickt.
-

TOM by Example

Datenbank Administrator (mit TOM)



- 👤 **07:05h:** Der DBA sitzt beim Frühstück und liest das SMS. Er beschließt, sich das Ganze von seinem Remote-Zugang anzusehen.
 - 👤 **07:22h:** Der DBA findet wegen der Alerts, dass drei SQL Statements vermutlich aufgrund der neuen Objekt Statistik auf einer Tabelle jetzt einen Full Table Scan machen (`SQL_ID` Report der alerted Statements).
 - 👤 **07:27h:** Der DBA resettet die eine Tabellen Statistik und schickt die `SQL_ID` Reports an den Support des Applikationsherstellers mit der Bitte, diese sich doch anzusehen, ob die Statements vielleicht „getuned“ werden können.
 - 👤 **08:00h:** Es melden sich nach und nach weitere Benutzer an, und das System zeigt keine Performance-Probleme.
 - 👤 **2 Monate später:** Beim nächsten Upgrade der Applikation werden unter anderem die problematischen Statement mit Hints versehen.
-



Was hat TOM erspart?

- ☹ viel Zeit,
da die Performance nicht abgefallen ist (Gehälter der Benutzer)
 - ☹ viel Arbeit (Analysen, Meetings, etc.)
 - ☹ viele Nerven (Diskussionen, Eskalationen, etc.)
 - ☹ viel Geld (neue Server, zusätzliche CPU Lizenzkosten, etc.)
 - ☹ viele Peinlichkeiten (für den DBA der die Statistiken gesperrt und dann vergessen hat)
-

TOM by Example

Applikationsverantwortlicher (ohne TOM)



- 🗣️ **09:13h:** Ein Benutzer ruft beim Applikationsverantwortlichen an und meldet, dass eine bestimmte Auswertung bei ihm 20 Minuten dauert.
 - 🗣️ **09:15h:** Der Applikationsverantwortliche lässt die Auswertung bei sich laufen und erhält nach zehn Sekunden das Ergebnis.
 - 🗣️ **09:17h – 10:22h:** Telefonat zwischen den beiden. Was macht der Benutzer? Hat er etwas geändert? Etc. Sie beschließen, dass die IT sich des Problems annehmen muss.
 - 🗣️ **2 Tage später:** Die Kollegen von der IT finden nichts. Der PC ist in Ordnung. Andere Applikationen laufen problemlos. Sie beschließen, das Problem beim Hersteller zu melden.
-

TOM by Example

Applikationsverantwortlicher (ohne TOM)



- 🐛 **3 Wochen und viele, viele Stunden Problemanalyse später:** Es ist immer noch keine Lösung gefunden worden. Der PC soll neu installiert werden.
 - 🐛 **noch 1 Woche später:** Der PC wurde installiert und vom Applikationsverantwortlichen getestet. Alles so wie es sein soll...
 - 🐛 **PC wird ausgeliefert:** Der Benutzer erhält den neuen PC, meldet sich an (mit Windows Roaming Profiles), startet die Applikation und es ist wieder langsam! Es muss wohl am Profile liegen. Profile wird gelöscht, 17 andere Applikationen müssen rekonfiguriert werden, und auf einmal passt die Performance (aber keiner weiß warum).
-

TOM by Example

Applikationsverantwortlicher (mit TOM)



- 09:13h: Ein Benutzer ruft beim Applikationsverantwortlichen an und meldet, dass eine bestimmte Auswertung bei ihm 20 Minuten dauert.
 - 09:15h: Der Applikationsverantwortliche lässt die Auswertung bei sich laufen und erhält nach zehn Sekunden das Ergebnis.
 - 09:16h: Hat TOM vielleicht etwas bemerkt? Es sind keine Alerts da. Der Applikationsverantwortliche startet ein **TOM_GATHER.ALL_DATA** und ein **TOM_GATHER.EVALUATION_DATA** und erhält einen Alert. Mehrere Statements sind vermutlich aufgrund von NLS Settings langsamer.
 - 10:11h: Der Applikationsverantwortliche fragt in der IT den DBA was NLS Settings sind, und was man dagegen machen kann.
-

TOM by Example

Applikationsverantwortlicher (mit TOM)



- 🐼 **10:15h:** Der DBA telefoniert mit dem Benutzer. 10 Minuten später ist alles behoben. Durch die Installation einer anderen Oracle Applikation war in der Windows Benutzer Registry das NLS Setting auf Deutsch mit sprachenspezifischer Sortierung eingestellt.
-



Was hat TOM erspart?

- ⌚ viel Zeit, da die Performance nicht abgefallen ist (Gehalt des Benutzer)
 - ⌚ viel Arbeit (Analysen, Meetings, PC Neuinstallation, etc.)
-

TOM by Example

Kundensupport & Entwickler (ohne TOM)



- 👤 **11:10h:** Ein Kunde meldet ein Performance Problem beim Support. Der Support will wissen:
 - genaue Spezifikationen zum System
 - Identifikation des/der Statements
 - Execution Pläne der Statements
 - etc.
 - 👤 **16:33h:** Der DBA war die letzten fünf Stunden damit beschäftigt, alle Informationen zu sammeln (zumindest die, die er finden konnte).
 - 👤 **nächster Tag:** Der Support-Mitarbeiter hat die Daten analysiert und dafür einige Stunden benötigt. Er urgirt die fehlenden Informationen.
-

TOM by Example

Kundensupport & Entwickler (ohne TOM)



- 👤 **noch 2 Tage später:** Nach über 10x hin und her ohne einen Erfolg zu haben, eskaliert der Kunde das Problem. Ab jetzt sind mehrere Kollegen mit dem Problem beschäftigt.
 - 👤 **und noch 2 Tage:** Nach vielen weiteren e-mails und Telefonaten konnten letztendlich die benötigten Daten für die Analyse gesammelt werden. Die Bearbeitung geht weiter an den Entwickler.
 - 👤 **und noch 2 Tage:** Der Entwickler nimmt sich des Falles an, kann das Problem aber leider nicht reproduzieren. Es entwickelt sich wieder ein Hin und Her mit dem Kunden.
 - 👤 **und wieder 10 Tage:** Nach einer erneuten Eskalation ist der Kunde bereit, die Reisekosten für einen Einsatz vor Ort zu bezahlen.
 - 👤 **und noch 2 Tage:** Ein Entwickler kommt zum Kunden, kann das Problem reproduzieren, aber nicht sofort lösen.
-

TOM by Example

Kundensupport & Entwickler (ohne TOM)



- 🐞 **3 Wochen nach dem eigentlichen Problem:** Endlich gelingt es die Ursache zu finden. Der Kunde hat einen undokumentierten `init.ora` Parameter gesetzt (um ein anderes Problem zu lösen), der dazu geführt hat, dass bestimmte Optimizer Features deaktiviert waren. Der Softwarehersteller besteht darauf, dass der Parameter entfernt wird.
 - 🐞 **und noch ein Tag vergeht:** Ratlosigkeit beim Kunden. Wenn man den Parameter entfernt, funktioniert eine andere Applikation wieder nicht. Letztendlich hat der DBA die rettende Idee, den Parameter in einem Logon Trigger zu setzen.
-

TOM by Example

Kundensupport & Entwickler (mit TOM)



- 👤 **11:10h:** Ein Kunde meldet ein Performance Problem beim Support. Der Support will wissen:
 - genaue Spezifikationen zum System
 - Identifikation des/der Statements
 - Execution Pläne der Statements
 - etc.
 - 👤 **11:15h:** Der DBA erzeugt einem `TOP_SQL` Report sowie einen `SQL_ID` Report für alle 17 von TOM gemeldeten Statements und schickt Sie dem Kundensupport.
 - 👤 **11:25h:** Der DBA schaut sich die TOM Alerts an und stellt fest, dass alle mit Optimizer Environment Settings zu tun haben. Er telefoniert daraufhin noch einmal mit dem Kundensupport.
-

TOM by Example

Kundensupport & Entwickler (mit TOM)



- 🕒 **13:15h:** Der Entwickler kommt von der Mittagspause zurück und hat eine Supportanfrage auf dem Tisch, ob denn ein bestimmter Oracle Parameter Einfluss auf die Performance haben könne. Er setzt den Parameter und überprüft die Performance der betroffenen Applikationsteile. Ja, die Performance wird dadurch schlechter.
 - 🕒 **14:30h:** Der Kunde wird informiert, dass der Parameter nicht gesetzt werden darf. Als Workaround wäre ein Logon Trigger angebracht.
 - 🕒 **14:55h:** Der DBA hat den Logon Trigger implementiert, das Problem ist gelöst.
-



Was hat TOM erspart?

- ⌚ viel Zeit für die Problemanalyse auf allen Seiten
 - ⌚ keine Zusatzkosten für den Kunden (Arbeitszeit, Reisekosten, etc.)
-



Leistet TOM das wirklich?

- ☛ Ja, das sind Beispiele, wie die Nutzung von TOM viele Oracle optimizer-relevante Probleme schnell und effektiv lösen kann.
 - ☛ Dabei ist es nicht wichtig, dass Sie selbst mit den Auswertungen von TOM etwas anfangen können (siehe Applikationsverantwortlicher). Es ist einfach wichtig, die für eine Analyse benötigten Daten, schnell und effektiv einem „Wissenden“ zugänglich zu machen. Dieser kann auch extern sein.
-



Verfügbarkeit TOM 1.3

- An TOM 1.3 wird derzeit unter Hochdruck gearbeitet.
 - Aktuell gibt es eine GNC-interne TOM 1.3 Alpha Version.
 - Die Beta Version werden wir an ausgewählte Kunden und Interessenten verleihen, die uns beim Optimieren von TOM unterstützen wollen. Wir freuen uns, wenn Sie in diesen Kreis mit aufgenommen werden wollen.
-

Ausblick auf TOM 2.0

(Subject to be changed!)



- GUI, um es den „nicht DBAs“ einfacher zu machen
 - zentrales Repository für TOM (Performance Daten in einer zentralen Datenbank sammeln und analysieren)
 - Oracle Database Konfigurationsempfehlungen
 - Instanz Einstellungen
 - I/O Performance Themen
 - User Management
 - Wer darf was im TOM machen/sehen
 - Definition mehrere e-mail Adressen (personenbezogen)
-

GNC Global Network Communication GmbH
GNC @ nussdorfer laende 23, 1190 vienna
T +43 664 246 5008, cp@gnc.at



Q & A