

Real Application Testing - Tipps und Tricks

Ulrike Schwinn und Maik Sandmann
Oracle Deutschland B.V. & Co.KG

Schlüsselwörter: Database Replay, SQL Performance Analyzer, Testumgebung, Migration

Einleitung

Testen ist unerlässlich und sollte zu jedem Software-Entwicklungszyklus bzw. Qualitätssicherungsverfahren gehören. Testen ist allerdings auch aufwändig, der Aufbau der Testumgebung und das zur Verfügungstellen von adäquaten Testprozeduren kann etliche Personentage bzw. Personen-Monate kosten. Erschwerend kommt hinzu, dass man mit den meisten Testprozeduren nur eine Simulation durchführen kann, die in der Regel nicht dem realen Applikationscharakter entspricht. Seit Database Release 11g gibt es die Möglichkeit, mit der neuen Option „Real Application Testing“ ohne großen Aufwand unterschiedliche Tests für die Bewertung von Migrationen, neuen Features, neuen Architekturen oder auch Plattformen durchzuführen. „SQL Performance Analyzer“ und „Database Replay“ sind Bestandteil dieser neuen Option, die auch teilweise schon vor 11g zum Einsatz kommen kann. Der Artikel gibt einen Überblick über die beiden Technologien und beschreibt die Besonderheiten, Unterschiede und Einsatzmöglichkeiten. Zusätzlich werden Strategien zur Durchführung von Tests im Datenbankumfeld vorgestellt und in diesem Zusammenhang häufig gestellte Fragen beantwortet. So können auch Leser, die Real Application Testing (noch) nicht im Einsatz haben, Tipps für ihre eigene Umgebungen ableiten.

Real Application Testing – Besonderheiten

Der **SQL Performance Analyzer** (kurz SPA) ist – wie der Name schon anzeigt – ein Werkzeug, das auf die Analyse eines SQL Workloads spezialisiert ist. Einzige Voraussetzung ist das Vorhandensein eines **SQL Tuning Sets** – eine im Data Dictionary gespeicherte Sammlung von SQL Texten mit Execution Kontext und Statistiken. Sie können seit 10g sehr einfach über Packages oder mit Unterstützung des Enterprise Managers (siehe Abb1.) aus unterschiedlichen Quellen - beispielsweise inkrementell aus dem Cursor Cache - erzeugt werden. Um die Funktion schon in 9i zu nutzen, gibt es sogar die Möglichkeit, SQL Tuning Sets aus SQL Traces zu generieren. Die Test-Durchführung mit SPA ist denkbar einfach: Die SQL Statements aus dem SQL Tuning Set werden vor und nach einer Veränderung ausgeführt. Damit unterschiedliche Caching-Effekte die Ergebnisse nicht verfälschen, werden die Statements automatisch pro Test mindestens 2 mal ausgeführt. Als Ergebnis erhält man detaillierte Ausführungspläne mit den zugehörigen Ausführungsmetriken wie CPU, Elapsed Time etc. Im Unterschied zu Database Replay weist SPA folgende Besonderheiten auf:

- Ohne Backup und ohne definierte Methode zum Zurücksetzen des Tests verwendbar
- Detaillierte Informationen über alle unterstützten Statements im SQL Tuning Set
- Unterstützung von SELECT Statements und des SELECT Anteils im DML
- Vollständig nutzbar seit 9i
- Integrierte Empfehlungen für Maßnahmen zur Behebung von Regressionen (z.B. SQL Plan Baselines und SQL Performance Analyzer)

Mit neuen Funktionen im Package DBMS_SQLPA können seit 11g Release 2 sogar zwei unterschiedliche SQL Tunings Sets miteinander verglichen werden. Dies macht einen vollständigen

Vergleich von unterschiedlichen Applikationen möglich – z.B. ein Vergleich zwischen einer migrierten und einer (noch) nicht migrierten Umgebung.

ORACLE Enterprise Manager
Grid Control 11g

Database Instance: ord | Logged in As SYSTEM

SQL Tuning Sets

A SQL Tuning Set is a collection of SQL Statements that can be used for tuning purposes.

Search:

Filter on a name or partial name

Page Refreshed Sep 2, 2010 1:36:02 PM CEST

Details | Drop | Copy To A Database | Export To A File | Schedule SQL Tuning Advisor | Schedule SQL Access Advisor | Create | Import From A File

Select	Name	Schema	Description	SQL Count	Created	Last Modified
<input checked="" type="radio"/>	RACTEST	SYSTEM		0	8/26/10 11:45 AM	8/26/10 11:45 AM
<input type="radio"/>	EXADATA_SET	SYS		1	6/2/10 12:10 PM	6/2/10 12:15 PM
<input type="radio"/>	TEST2	SYS		10	5/27/10 5:12 PM	5/27/10 5:12 PM
<input type="radio"/>	TEST1	SYS		8	5/27/10 5:09 PM	5/27/10 5:09 PM

Related Links
[SQL Performance Analyzer](#)

Home | Targets | Deployments | Alerts | Compliance | Jobs | Reports | My Oracle Support | Setup | Preferences | Help | Logout

Abb. 1: SQL Tuning Sets im Enterprise Manager

Database Replay führt im Gegensatz zu SPA den gesamten Workload in einer Testumgebung aus - d.h. alle SQL, PL/SQL Operationen usw. können aufgezeichnet und abgespielt werden. Die aufgezeichneten binären Dateien sind **plattformunabhängig** und lassen sich somit zum Test auf unterschiedlichen Plattformen nutzen. Dabei ist es möglich, während der Aufzeichnung (auch Capture genannt) oder nachträglich beim Replay, Filter einzusetzen, um das Ergebnis sinnvoll einzuschränken bzw. um Fehler bei nichtexistierenden Schema Usern auf der Testseite zu vermeiden. Zur ersten Bewertung stehen spezielle Capture-, Replay- und Divergenz-Reports zur Verfügung. Um detaillierte Aussagen über die Performance zu machen, werden in der Regel zusätzliche Reports wie Compare Period, AWR bzw. AWR DIFF und ASH herangezogen. Zu beachten ist, dass einige **Einschränkungen** bei der Aufnahme des Workloads existieren, die im Handbuch Oracle Database Real Application Testing User's Guide im Kapitel 3 aufgelistet sind. So können beispielsweise keine Direct Load Operationen durch SQL*Loader aufgezeichnet werden. Abspielen lässt sich der Workload auch erst ab 11g Release 1, so dass ein Vergleich zweier 10g Umgebungen nicht möglich ist. Wie in der Einleitung schon erwähnt, ist Real Application Testing ohne großen Aufwand d.h. ohne zusätzliche **Installation** und **Skripts** nutzbar. Je nach Release-Stand sind allerdings entsprechende **Patches** erforderlich, um die Technologie auch in Releases vor 11g zur Verfügung zu stellen. Note 560977.1 gibt dabei einen aktuellen Überblick über die empfohlenen Patches. Nutzt man Database Replay für den Test einer Migration von 10g nach 11g muss man bei der Aufnahme in 10g den Parameter **PRE_11g_ENABLE_CAPTURE** auf den Wert TRUE setzen, um die Funktionen zu aktivieren. Grundsätzlich sind für den Replay spezielle **Workload Replay Clients** auf dem Testrechner nötig. Diese werden standardmässig mit ausgeliefert. Möchte man die Workload Replay Clients vom Testsystem isolieren, steht die zugehörige Instant Client Software zum Download auf OTN zur Verfügung (z.B. für Linux unter <http://www.oracle.com/technetwork/topics/linuxsoft-082809.html>).

Auch die Nutzung im Linemode wird durch das Package Framework ermöglicht. So können eigene Skripte über die **neuen Packages** DBMS_WORKLOAD_CAPTURE, DBMS_WORKLOAD_REPLAY und DBMS_SQLPA und die entsprechenden Views erzeugt werden. Vorgefertigte Skripte, die für den Ablauf von Database Replay notwendig sind, können beispielsweise von <http://apex.oracle.com/pls/apex/dbreplayscr> geladen werden. Beide Techniken sind auch graphisch mit Enterprise Manager Database Control 10.2.0.4, Grid Control 10.2.0.5 und 11g verwendbar. Hat man bei der Nutzung die Wahl zwischen Enterprise Manager Database Control und Grid Control sollte man **Grid Control** bevorzugen. Grid Control bietet eine vollständige

Automatisierung des gesamten Capture- und Replay-Prozesses an und verfügt im Unterschied zu Database Control über einen zusätzlichen Report - den **Compare Period Report** (siehe Abb. 2). Dieser gibt einen Überblick über wichtige Unterschiede oder Kennzahlen der Systeme im Vergleich – wie z.B. Divergenzen, Memory und Optimizer Parameter sowie Performancestatistiken (siehe ebenfalls Abb. 2). Besonders beim Vergleich von RAC und Single Instance Systemen kann dieser Report einen guten Gesamteindruck über die Performance vermitteln, da im Unterschied zu den einzelnen AWR DIFF Reports eine globale Übersicht über alle eingesetzten Instanzen gegeben wird.

(-) Information About Databases

	Capture	Replay
Database Name	BUVMRAC	ORCL
Original Database ID	723233154	1247245911
Version	11.2.0.1.0	11.2.0.1.0
Platform	Linux IA (32-bit)	Linux IA (32-bit)
Is It RAC?	YES	NO
Number of Instances	2	1

(+) Information About AWR and Time Periods

(+) Changes to Important Parameters

(+) Changes to Optimizer-Relevant Parameters

(+) Changes to Memory Configuration Parameters

(+) Instances of the Capture Database

(+) Instances of the Replay Database

(+) **Replay Divergence**

(+) **Main Performance Statistics**

(+) **Top SQL by Change in DB Time**

(+) **Hardware Usage Comparison**

(+) **ADDM Comparison**

Abb. 2: Ausschnitt aus einem Compare Period Report

Offensichtlich sind keine Kenntnisse über den Workload oder die Installation von Clients notwendig. Allerdings sollte man sich vorher vergewissern, ob **externe Quellen** wie external tables, database links etc. genutzt werden, um diese gegebenenfalls zur Verfügung zu stellen oder abzuklemmen. Auch hier bietet der Workflow in der GUI eine gute Hilfestellung. Darüber hinaus erfordern langlaufende **PL/SQL Codestrecken** eine differenzierte Handhabung des Replay Option Parameters SYNCHRONIZATION. Hier kann es sinnvoll sein, vom Defaultwert TRUE abzuweichen und FALSE zu nutzen.

Wer kann nun Real Application Testing nutzen? Welche Rechte sind erforderlich um einen Workload aufzunehmen und abzuspielen oder SQL Tuning Sets zu testen? Die SPA Packages stehen dem User PUBLIC zur Verfügung, zum Erzeugen von SQL Tuning Sets bzw. zur Analyse des Ergebnisses sind ADMINISTER SQL TUNING SET- und ADVISOR- Privilegien nötig. Bei der Database Replay Technologie sind ebenfalls keine DBA Privilegien erforderlich, sondern EXECUTE-Rechte auf die entsprechenden Packages sowie die Privilegien bzw. Rollen CREATE ANY DIRECTORY, CREATE SESSION, BECOME USER und SELECT_CATALOG_ROLE und die entsprechenden Rechte auf den Servern, um Dateien zu schreiben und zu kopieren.

Der Testablauf

Die **Dauer** und den **Inhalt** der Tests abzugrenzen sind wichtige Aufgaben bei der Festlegung der Testszenarios. Hierbei spielt die Definition der zeitlichen Komponente wie auch der parallel laufenden Applikationen eine wichtige Rolle. Nicht nur das lange Warten auf das Ergebnis, sondern auch AWR Reports oder gar AWR DIFF Reports über viele Stunden zu interpretieren, ist häufig nicht sinnvoll, da Schwachstellen und Bottlenecks nicht mehr aufzufinden sind. Anstelle eines einzigen umfangreichen Tests sollte, falls möglich, die Durchführung **mehrerer Tests** eingeplant werden. Zusätzlich sollte der

CAPTURE Filter genutzt werden, um keine unnötigen Daten bzw. Operationen beispielsweise vom SYS User oder dem Enterprise Manager OMS oder Agent User zu speichern und eine Konzentration auf die wichtigsten Applikationen zu gewährleisten. Stand August 2010 ist es noch nicht möglich mehrere Tests parallel durchzuführen, so dass die Tests sequentiell erfolgen müssen. Sehr leicht und ohne große Vorbereitung lassen sich Tests durchführen, die auf Änderungen von **Initialisierungsparameter** beruhen. Besonders in 11g lohnt ein Blick in die Liste der neuen Initialisierungsparameter. So können beispielsweise folgende Parameter bei Tests sehr hilfreich sein:

- OPTIMIZER_FEATURES_ENABLE für unterschiedliche Optimizer-Releasestände
- OPTIMIZER_USE_INVISIBLE_INDEXES bei Änderungen an den Indizes
- OPTIMIZER_USE_PENDING_STATISTICS zur Nutzung von PENDING Statistiken
- RESULT_CACHE_MODE um den Effekt von Ergebnis-Caching zu nutzen
- OPTIMIZER_USE_SQL_PLAN_BASELINES zum Ein- und Auschalten von SQL Plan Baselines
- PLSQL_CODE_TYPE für unterschiedliche Einstellungen des Kompilierungsmodus (NATIVE oder INTERPRETED)

Beeinhaltet der Workload einen signifikanten Anteil an komplexen SQL-Abfragen, so sollte generell vor der Verwendung von Database Replay mittels **SPA** sichergestellt werden, dass die SQL-Abfragen auch in der neuen Umgebung mit performanten Ausführungsplänen arbeiten. Ohne hohen Aufwand kann man dabei sehr schnell zu einem Ergebnis kommen, SQL Statements mit Regressionen filtern und die entsprechenden Maßnahmen ergreifen. Für die Nutzung von SPA ist nämlich nur ein entsprechendes SQL Tuning Set erforderlich, das inkrementell ohne Vorbereitung auf dem Produktionssystem erzeugt werden kann. Importiert in die Testumgebung können mit SPA schnell die ersten Tests durchgeführt und über den Vergleich der Metrikwerte und die genaue Analyse der Ausführungspläne bewertet werden.

Verwendet man Database Replay zur Analyse sollte im ersten Schritt die korrekte funktionale Durchführung überprüft werden, bevor Aussagen über die Performance mit AWR bzw. ASH Reports getroffen werden. Unterschiede und Fehler bei der Ausführung können das Ergebnis verfälschen und im Extremfall dazu führen, dass das Gesamtergebnis des Replays unbrauchbar ist. Database Replay bietet hier den Replay Report mit der Abschätzung der Divergenzgröße in Prozent und seit 11g Release 2 einen separaten **Divergenz-Report** (siehe Abb. 3) an, der die einzelnen Statements mit den zugehörigen Fehlern auflistet. Ein Fehler-Aufkommen von 5% bis 10% ist dabei in der Regel tolerierbar.

Diverged Calls During Replay: RACTEST

View the most relevant set of replayed calls that have diverged from the capture by filtering out all but the ones of interest. A large number of calls may be relevant. If so, consider grouping them by an attribute value they have in common.

Conditions for Displaying Diverged Calls

Filter Conditions for Diverged Calls

Only the calls that meet all the following filter conditions will be displayed among the results. Conditions with empty values will not contribute to filtering.

Type of Divergence: Error Observed:

SQL ID:

Session (ID:Serial#): Service:

Module:

Action:

Grouping of Filtered Calls

Summarize the diverged calls displayed by the filter using one of the following groupings.

Grouping Attribute:

Grouping on filtered attributes will be ignored.

Fetch Diverged Calls

Diverged Calls

Below is the table of diverged calls that matches the specified filter conditions.

TIP The SQL ID link for individual calls allows you to drill down to see the SQL text of the call and the variables bound. This link is disabled when the call details have not yet been loaded into the server. Click the 'Load Visible Calls' button to load the call details for each call currently displayed.

SQL ID	Timestamp	Type of Divergence	Divergence Details	Rows Observed	Error Observed	Service	Module	Action	Session
2cq9mk2hg37dc	Aug 25, 2010 2:15:04 PM CEST	Error No Longer Seen During Replay	Expected Error Number: 01013	0	No error	SYSSUSERS	sqlplus@bumucsvm2.de.oracle.com (TNS V1-V3)		47:1867

Abb. 3: Ausschnitt aus dem Divergenz-Report

Die weitere Analyse erfolgt danach über den Compare Period Report, den AWR Diff Report bzw. die ASH Reports. Läuft der Test im 11g Release 2 Umfeld, kann zusätzlich ein **globaler AWR Diff Report** im RAC Umfeld gute Dienste leisten. Dieser ist im ORACLE_HOME/rdbms/admin Verzeichnis unter dem Namen awrgdrpt.sql zu finden und erzeugt den globalen Vergleich aller RAC Instanzen bzgl. zweier verschiedener Zeiträume. Abbildung 4 zeigt einen globalen AWR DIFF Report, in dem ein Single Instance-Testlauf mit einem RAC-Testlauf (mit 2 Instances) verglichen wird.

WORKLOAD REPOSITORY RAC Compare Period Report for

Database Summary

Snapshot Set	Database				Snapshot Ids		Number of Instances		Number of Hosts		Report Total (minutes)	
	Id	Name	RAC	Block Size	Begin	End	In Report	Total	In Report	Total	DB time	Elapsed time
First (1st)	1247245911	ORCL	NO	8192	2158	2159	1	1	1	1	9.79	12.65
Second (2nd)	1318851898	BUVMRAC	YES	8192	3689	3690	2	2	2	2	12.63	3.78

Database Instances Included In Report

Set	DB Id	Inst #	Instance	Release	Host	Startup	Begin Snap Time	End Snap Time	Elapsed Time (min)	DB time (min)	Instance Up Time (hrs)	Avg Active Sessions	Platform
1st	1247245911	1	orcl	11.2.0.1.0	sccloud034.de.oracle.com	02-Aug-10 17:15	25-Aug-10 14:06	25-Aug-10 14:19	12.65	9.79	549.07	0.77	Linux IA (32-bit)
2nd	1318851898	1	buvmrac_1	11.2.0.1.0	bumucsvm3.de.oracle.com	23-Aug-10 14:11	23-Aug-10 15:30	23-Aug-10 15:34	3.77	4.02	1.38	1.07	Linux IA (32-bit)
		2	buvmrac_2	11.2.0.1.0	bumucsvm1.de.oracle.com	23-Aug-10 12:34	23-Aug-10 15:30	23-Aug-10 15:34	3.77	8.61	3.00	2.29	Linux IA (32-bit)

Report Summary

Host Configuration Comparison

- CPU and Memory values are from the end snapshot, averaged across all instances
- Other values are averages for all instances

	1st	2nd	Diff	%Diff
Number of CPUs:	1	2	1	100.0

Abb. 4: Ausschnitt aus einem globalen AWR Diff Report

Eine Hauptvoraussetzung bei der Durchführung von Tests ist das Vorhandensein einer **realen und konsistenten Testumgebung**. Falls die Ausstattung der Server im Vergleich unterschiedlich ist z.B. bzgl. CPU, Memory etc. wird sich dies im Ergebnis widerspiegeln. Dies gilt natürlich auch bei der Nutzung von unterschiedlichen Architekturen wie z.B. bei Nutzung von RAC oder Single Instance. Der Datenbestand hingegen sollte im Idealfall gleich sein. Um dies zu gewährleisten können unterschiedliche Techniken zum Einsatz kommen wie zum Beispiel Data Pump, RMAN DUPLICATE Kommando basierend auf Backup Dateien/aktiven Datenbankdateien oder die Transportable

Tablespace Technologie. Wichtig dabei ist, einen möglichst aktuellen und konsistenten Testumgebungsstand zu erzeugen, der mit dem Datenbankstand (SCN) vor dem Beginn des Captures übereinstimmt. Gibt es Unterschiede im Datenbestand, wird die Replay Prozedur in jedem Fall ablaufen, allerdings die Unterschiede bei der Durchführung mit Divergenzfehlern quittieren. Kommt SPA zum Einsatz und sind die Datenbestände unterschiedlich, sollten wenigstens die Statistiken gleich sein und die EXPLAIN PLAN Methode gewählt werden. So können wenigstens die Unterschiede in den Plänen herausgestellt werden. Sollen unterschiedliche Database Replay Tests in dieser Testumgebung ablaufen, kann zusätzlich die Wiederholbarkeit durch die FLASHBACK DATABASE Technologie und RESTORE POINTS oder durch die neue Snapshot Standby-Technik in 11g ermöglicht werden.

Häufig stellt sich die Frage, wie viel Einfluß die Nutzung von Database Replay auf die Performance haben kann. Eine Rolle spielt sicherlich die Wahl des Speicherorts für die Capture Dateien. Ein separates I/O System kann dabei von Vorteil sein - im Fall von RAC sollte dieses auch gemeinsam nutzbar sein. Zeigen sich Performance Engpässe bei der Aufzeichnung oder beim Abspielen, lohnt sich ein Blick in die Alert-Datei, da dort alle Operationen im Zusammenhang mit Database Replay aufgezeichnet werden. So kann beispielsweise eine lang andauernde Database Replay Operation mit Engpässen der Data Pump zusammenhängen, die standardmäßig als RESUMABLE Operation abläuft.

Migrationen

Real Application Testing findet seine Verwendung nicht nur im Testen von neuen Features bei gleichem Release-Stand der Datenbank, sondern ist auch sehr gut geeignet, um Aussagen über Migrationen zu treffen. Kundenaussagen, wie die folgende Aussage, bestätigen diese Einschätzung: „Das Resultat dieses Tests entspricht nahezu vollständig dem Bild, das sich uns seit Anfang der Woche in der Produktion zeigt.“ Um Migrationen bzw. Tests in migrierten Umgebungen erfolgreich durchzuführen, ist es selbstverständlich, sich mit den Neuigkeiten vertraut zu machen. Hervorzuheben sind dabei die Features, die zu Veränderungen im Ablauf führen, die Performance erhöhen können, Speicherplatz sparen oder die Datenbankfunktionalität erweitern können. Trotz all dieser Neuigkeiten sollte man im ersten Test „konservativ“ vorgehen und mit der „gewohnten, unveränderten“ **Parameterdatei** arbeiten, soweit dies möglich ist. Werden beispielsweise festgeschriebene Memory-Parameter mit DB_CACHE_SIZE und SHARED_POOL_SIZE im aktuellen Produktionsrelease verwendet, sollten diese im ersten Test beibehalten werden, um gleiche Bedingungen bei der Memorynutzung zu schaffen. Obsolete (auch deprecated) Parameter oder der Parameter COMPATIBLE, der die Nutzung des aktuellen Release-Stands anzeigt, müssen natürlich angepasst werden. Läßt man vor der Migration die Skripte utlu<version>i.sql (z.B. utlu112i.sql) ablaufen, bekommt man einen Überblick über obsolete und umbenannte Parameter. Das Skript läßt sich vorab separat über die Metalink Note 884522.1 laden. Gerade im Release 11g gibt es einige nennenswerte Änderungen im Security Umfeld, die bei den Tests bekannt sein sollten. So sind Passwörter standardmäßig case sensitiv – das Feature läßt sich allerdings mit dem Parameter SEC_CASE_SENSITIV_LOGON beeinflussen. Um einen Überblick über die neuen Features zu erhalten, lohnt sich daher in jedem Fall ein Blick in den New Features Guide.

Fazit

Die Verwendung von Real Application Testing besonders über die Integration in den Enterprise Manager ist ein leicht zu verwendendes Werkzeug, das besonders im Lasttest-Umfeld gute Dienste leisten kann. Zur Bewertung sind allerdings gute Kenntnisse über Ausführungspläne und AWR Reports unabdingbar. Die Konzentration auf SQL-Regressionen über SPA ist im ersten Schritt zu

empfehlen. Das SQL Plan Management und der SQL Performance Analyzer können dabei zum Einsatz kommen; erst danach sollten Workloads über Database Replay getestet werden.

Weiterentwicklungen speziell auch im Bereich Integration mit anderen Technologien und Werkzeugen sind geplant. Bei der Erstellung dieses Beitrag (Stand August 2010) standen dazu allerdings noch keine Informationen zur Verfügung.

Informationen

Folgende Auflistung gibt einen Überblick über die im Artikel verwendeten Links und Dokumente.

- DBA Community :
 - <http://www.oracle.com/global/de/community/dbadmin/index.html>
- Real Application Testing Handbuch
 - http://www.oracle.com/pls/db112/to_toc?pathname=server.112/e12254/toc.htm
- New Features Guide
 - http://www.oracle.com/pls/db112/to_toc?pathname=server.112/e10881/toc.htm
- Metalink Note 884522.1
- Metalink Note 560977.1
- Upgrade Advisor von 10.2 nach 11g – My Oracle Support DOC ID =251.1
- Database Replay Skripte
 - <http://apex.oracle.com/pls/apex/dbreplayscr>
- Download Workload Replay Client für Linux
 - <http://www.oracle.com/technetwork/topics/linuxsoft-082809.html>

Kontaktadressen:

Ulrike Schwinn
Oracle BU DB – Business Unit Database
ORACLE Deutschland B.V. & Co. KG
Riesstr. 25, 80992 München

Telefon: +49 89 1430 1865
Ulrike.Schwinn@oracle.com

Maik Sandmann
Oracle BU DB – Business Unit Database
ORACLE Deutschland B.V. & Co. KG
Thurnithistr. 2, 30519 Hannover

Telefon: +49 89 14302212
Maik.Sandmann@oracle.com