

## DOAG Konferenz 2015

Martin Klier, managing partner database technology at Performing Databases.



NOVEMBER 2015

# Oracle Core für Einsteiger: Das Alert Log

## Einleitung

Die wesentliche Aufgabe eines Oracle Datenbankadministrators oder -operators ist es, in Frieden und Störfall gleichermaßen den Überblick in "seinen" Datenbankinstanzen zu gewinnen und zu behalten. Eine solche Datenbankinstanz besteht aus einer zunächst unübersichtlichen Vielzahl von Speicherbereichen und Prozessen. Bei einer durchschnittlichen 12cR1 Multitenant-Instanz ist so beispielsweise von mindestens 60 Hintergrundprozessen<sup>1</sup> auszugehen. Dazu kommen noch die Server- (Vordergrund-) Prozesse, die für das Bedienen von User-Sessions notwendig sind. Diese Vielzahl von aktiven Komponenten braucht ein klares Konzept, wie Fehlermeldungen den Operator erreichen, und wie man zwischen der bloßen Signalisierung eines Vorfalles und dessen Details abgrenzt.

Oracle hat sich an der Stelle für eine Teilung entschieden: Die Datenbank-Komponenten (=Hintergrund- und Vordergrundprozesse) loggen Ihre Fehler und Nachrichten in eine fortlaufende Protokolldatei mit dem Namen "Alert Log", das für jede Instanz einzeln geführt wird. Demzufolge gibt es für jede Instanz auf einem Host ein eigenes Alert Log, und auf jedem RAC-Knoten ebenfalls. Weiterführende Informationen zu entsprechend kritischen Störungen finden sich in mehreren Kategorien von referenzierten Trace- und Dump Files.

Die Oracle Dokumentation<sup>2</sup> definiert dieses Alert Log folgendermaßen:

"Eine Datei, die ein chronologisches Protokoll von Nachrichten und Fehlern der Datenbankinstanz enthält."

---

<sup>1</sup> SQL> select \* from v\$process

<sup>2</sup> <http://docs.oracle.com/database/121/CNCPT/glossary.htm#CHDJEHBJ>

# Architektur

Als Grundlage des Folgenden sei hier die Architektur des Oracle RDBMS grob zusammengefasst. Als Überblick dient eine Graphik aus der Oracle Online-Dokumentation<sup>3</sup>. Die Zusammenhänge sind stark vereinfacht, sollten aber einen Einblick in die grundlegende Funktionsweise geben.

## Lesen

Beim klassischen Lesevorgang parst der Serverprozess (=Session) unter Rückgriff auf die SQL-Optimizer-Bibliotheken ein Statement (z.B. SELECT), und legt dafür einen Cursor im Library Cache des Shared Pool ab. Bei der Ausführung holt derselbe Prozess die notwendigen Blöcke vom Tablespace, legt diese im Buffer Cache ab, und extrahiert die für die Abfrage relevanten Teile. Ist eine Aufbereitung oder ein aufwändigerer Join notwendig, verwendet der Serverprozess seine PGA, um zu sortieren, Hash Tables abzugleichen oder Merges auszuführen.

## Schreiben / Verändern

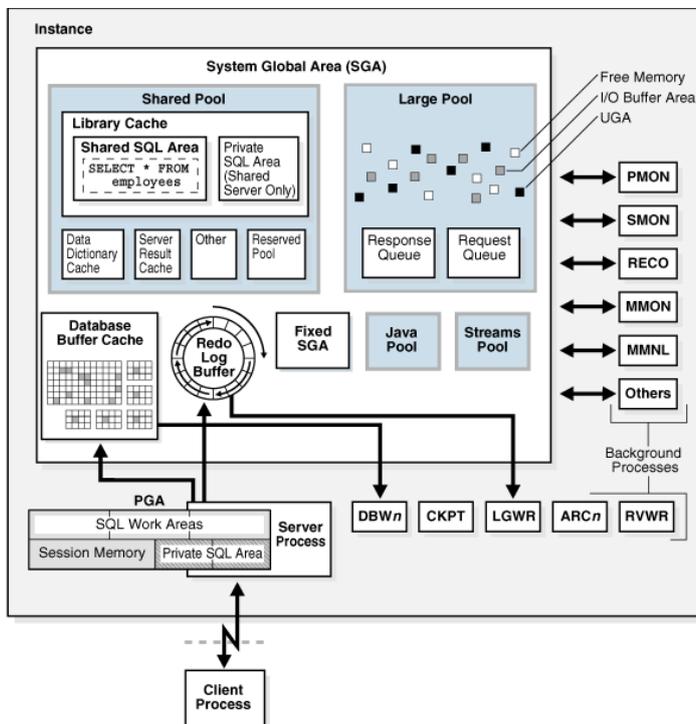
Der Schreibvorgang gleicht im Prinzip dem Lesevorgang - oft genug beinhaltet er einen solchen. Nach dem Parsen (z.B. eines

UPDATE) werden die Daten jedoch darüber hinaus im Buffer Cache verändert. Zur Absicherung des Cache gegen Verlust schreibt der Serverprozess die von ihm getätigten Veränderungen pro Block über seinen Private Strand im Shared Pool auch noch in den Redo Log Buffer, und der Hintergrundprozess Log Writer (LGWR) diese spätestens beim Commit in die Online Redo Log Files. Aus diesen Redo-Daten und dem Original-Block aus dem Tablespace wird im Fall eines Absturzes der Zustand des Buffer Cache rekonstruiert (Instance Recovery).

Vollständige Online Redo Logs sichert der Archiver (ARCn) in sogenannte Archived Redo Logs. Diese bilden eine spezielle Form eines eingebauten inkrementellen Backups: Aufbauend auf einer Vollsicherung z.B. mit dem Recovery Manager (RMAN) lassen sich die kurz "Archivelogs" genannten Bausteine für einen Roll-Forward einsetzen: Nach der Rücksicherung bewegt sich der Stand der Datenbank durch die Vergangenheit in Richtung Gegenwart, bis zum "Vollständigen Recovery" oder, falls gewünscht zu einem genau spezifizierten Punkt in der Zeit ("Point in Time Recovery") vor der Störung.

Der Database Writer (DBWn), der im Zusammenwirken mit dem Checkpoint-Prozess (CKPT) konsistente Abbilder der Buffer aus dem Cache auf den Tablespace synchronisiert, läuft unabhängig von aktuellen User-Operationen als "Lazy Writer". Damit wird sichergestellt, dass mittelfristig geschriebene Einträge im Online Redo Log unnötig werden, und überschrieben werden dürfen.

Im Rahmen der Veränderung von Datenzeilen durch den Serverprozess entstehen Undo-Daten, die in Form sogenannter Undo-Segmente dem gleichnamigen Tablespace zugeordnet sind. Sie dienen für die Fähigkeit, Transaktionen rückgängig zu machen ("Rollback") und zur Umsetzung des Transaktionskonzeptes allgemein, z.B. für die Consistent Read Isolation des Oracle RDBMS.



<sup>3</sup> <http://docs.oracle.com/database/121/CNCPT/process.htm#CNCPT1245>

# Darreichungsformen

Oracle präsentiert uns das Alert Log in verschiedenen Formen. Je nach dem welchem Zweck unser Zugriff dient, und mit welchen Mitteln er ausgeführt wird, bieten sich die unterschiedlichen Formate an.

---

## Textdatei

Sicher die älteste, bekannteste und zumeist verwendete Form des Alert Log ist die Textdatei. Sie hat ihren genormten Speicherort im "ADR"<sup>4</sup>, dessen Betriebssystem-Pfad der Datenbank durch den Parameter "diagnostic\_dest" bekanntgemacht wird.

```
Wed Jul 23 11:10:16 2014
Starting ORACLE instance (normal) (OS id: 7562)
Wed Jul 23 11:10:17 2014
CLI notifier numLatches:3 maxDescs:519
Wed Jul 23 11:10:17 2014
*****
Wed Jul 23 11:10:17 2014
Dump of system resources acquired for SHARED GLOBAL AREA (SGA)

Wed Jul 23 11:10:17 2014
Per process system memlock (soft) limit = 64K
Wed Jul 23 11:10:17 2014
Expected per process system memlock (soft) limit to lock
SHARED GLOBAL AREA (SGA) into memory: 1442M
Wed Jul 23 11:10:17 2014
Available system pagesizes:
4K, 2048K
Wed Jul 23 11:10:17 2014
Supported system pagesize(s):
Wed Jul 23 11:10:17 2014
PAGESIZE  AVAILABLE_PAGES  EXPECTED_PAGES  ALLOCATED_PAGES  ERROR(s)
Wed Jul 23 11:10:17 2014
4K          Configured           2             368642          NONE
Wed Jul 23 11:10:17 2014
2048K       1001                   721            0              NONE
Wed Jul 23 11:10:17 2014
RECOMMENDATION:
Wed Jul 23 11:10:17 2014
1. Increase per process memlock (soft) limit to at least 1442MB
to lock 100% of SHARED GLOBAL AREA (SGA) pages into physical memory
Wed Jul 23 11:10:17 2014
*****
LICENSE_MAX_SESSION = 0
LICENSE_SESSIONS_WARNING = 0
Initial number of CPU is 2
Number of processor cores in the system is 2
Number of processor sockets in the system is 1
Shared memory segment for instance monitoring created
Picked latch-free SCN scheme 3
Using LOG_ARCHIVE_DEST_1 parameter default value as USE_DB_RECOVERY_FILE_DEST
Autotune of undo retention is turned on.
```

Üblicherweise<sup>5</sup> setzt sich der Pfad wie folgt zusammen:

<diagnostic\_dest>/diag/rdbms/<db\_name\_in\_Kleinbuchstaben>/<Instanzname>/trace/alert\_<\$ORACLE\_SID>.log

Zum Beispiel auf dem Referenzsystem:

/u01/app/oracle/diag/rdbms/orcl/ORCL/trace/alert\_ORCL.log

Am häufigsten wird der DBA diese "klassische" Form verwenden, wenn er einfach ein fortlaufendes Log der Datenbank anzeigen möchte (z.B. mit Unix "tail") oder dem (Oracle) Support einen Überblick über die Gesamtsituation und jüngere Historie geben möchte. Nachteilig wirkt sich bei dieser Lektüre die entweder unübersichtliche Größe oder die referenzlose Teilung durch manuelle Rotationsmechanismen aus.

---

<sup>4</sup> Automatic Diagnostic Repository (<http://docs.oracle.com/database/121/CNCPT/startup.htm#CNCPT1913>)

<sup>5</sup> nach Oracle Flexible Architecture, kurz "OFA"

## XML-Datei

Die modernste Form zur Darstellung der anfallenden Information ist das XML-Format. Oracle stellt entsprechend auch sein Alert Log in der Gestalt zur Verfügung. Ein Beispiel ist beigefügt.

```
<msg time='2014-07-23T11:10:16.999+02:00' org_id='oracle' comp_id='rdbms'  
msg_id='opistr_real:1071:2538814769' type='NOTIFICATION' group='startup'  
level='16' host_id='kl11101788vm04' host_addr='192.168.99.50'  
pid='7562' version='1'>  
<txt>Starting ORACLE instance (normal) (OS id: 7562)  
</txt>  
</msg>  
<msg time='2014-07-23T11:10:17.001+02:00' org_id='oracle' comp_id='rdbms'  
type='UNKNOWN' level='16' host_id='kl11101788vm04'  
host_addr='192.168.99.50' pid='7562'>  
<txt>CLI notifier numLatches:3 maxDescs:519  
</txt>  
</msg>  
<msg time='2014-07-23T11:10:17.005+02:00' org_id='oracle' comp_id='rdbms'  
type='UNKNOWN' level='16' host_id='kl11101788vm04'  
host_addr='192.168.99.50' pid='7562'>  
<txt>*****  
</txt>  
</msg>  
<msg time='2014-07-23T11:10:17.005+02:00' org_id='oracle' comp_id='rdbms'  
type='UNKNOWN' level='16' host_id='kl11101788vm04'  
host_addr='192.168.99.50' pid='7562'>  
<txt>Dump of system resources acquired for SHARED GLOBAL AREA (SGA)  
</txt>  
</msg>  
<msg time='2014-07-23T11:10:17.006+02:00' org_id='oracle' comp_id='rdbms'  
type='UNKNOWN' level='16' host_id='kl11101788vm04'  
host_addr='192.168.99.50' pid='7562'>  
<txt>Per process system memlock (soft) limit = 64K  
</txt>  
</msg>  
<msg time='2014-07-23T11:10:17.006+02:00' org_id='oracle' comp_id='rdbms'  
type='UNKNOWN' level='16' host_id='kl11101788vm04'  
host_addr='192.168.99.50' pid='7562'>  
<txt>Expected per process system memlock (soft) limit to lock  
</txt>  
</msg>  
<msg time='2014-07-23T11:10:17.006+02:00' org_id='oracle' comp_id='rdbms'  
type='UNKNOWN' level='16' host_id='kl11101788vm04'  
host_addr='192.168.99.50' pid='7562'>  
<txt>SHARED GLOBAL AREA (SGA) into memory: 1442M  
</txt>
```

Der entsprechende Dateipfad im ADR lautet:

<diagnostic\_dest>/diag/rdbms/<db\_name\_in\_Kleinbuchstaben>/<Instanzname>/alert/log.xml

oder konkret z.B.:

/u01/app/oracle/diag/rdbms/orcl/ORCL/alert/log.xml

Diese Datei bietet sich für eine Analyse durch das menschliche Auge nicht an, ist aber prädestiniert für die maschinelle Auswertung. Nicht zuletzt Oracle selbst nutzt das Format, um ein virtuelles Datenbankobjekt zur Verfügung zu stellen (siehe nächster Absatz) oder eigene Tools damit zu speisen (z.B. ADRCLI).

## Datenbankobjekt - Fixed Table

Die Datenbank-Engine stellt uns mit der Tabelle x\$dbgalertext einen Zugriff per SQL auf die Daten des Alert Log bereit<sup>6</sup>.

Die nebenstehenden Graphiken zeigen ein Beispiel dafür. Dieser Weg eignet sich vorrangig für Rechercharbeiten oder Monitoringzwecke im laufenden Betrieb. Für die Suche nach der Ursache eines Fehlstartes der Instanz beispielsweise ist er denkbar ungeeignet.

```
40 | select originating_timestamp, message_id, message_group, message_text  
41 | from x$dbgalertext  
42 | where trim(COMPONENT_ID)='rdbms';
```

ORIGINATING_TIMESTAMP	MESSAGE_ID	MESSAGE_GROUP	MESSAGE_TEXT
23.07.14 11:10:16,998000000 +02:00	opistr_real:1071:2538814769	startup	Starting ORACLE instance (normal) (OS id: 7562)
23.07.14 11:10:17,001000000 +02:00	(null)	(null)	CLI notifier numLatches:3 maxDescs:519
23.07.14 11:10:17,005000000 +02:00	(null)	(null)	*****
23.07.14 11:10:17,005000000 +02:00	(null)	(null)	Dump of system resources acquired for SHARED GLOBAL AREA (SGA)
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	Per process system memlock (soft) limit = 64K
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	Expected per process system memlock (soft) limit to lock
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	SHARED GLOBAL AREA (SGA) into memory: 1442M
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	Available system pagesize(s):
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	4K, 2048K
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	Supported system pagesize(s):
23.07.14 11:10:17,006000000 +02:00	(null)	(null)	PAGESIZE AVAILABLE_PAGES EXPECTED_PAGES ALLOCATED_PAGES ERROR(s)
23.07.14 11:10:17,007000000 +02:00	(null)	(null)	4K Configured 2 368642 NONE
23.07.14 11:10:17,007000000 +02:00	(null)	(null)	2048K 1001 721 0 NONE
23.07.14 11:10:17,007000000 +02:00	(null)	(null)	RECOMMENDATION:
23.07.14 11:10:17,007000000 +02:00	(null)	(null)	1. Increase per process memlock (soft) limit to at least 1442MB
23.07.14 11:10:17,007000000 +02:00	(null)	(null)	To lock 100% of SHARED GLOBAL AREA (SGA) pages into physical memory
23.07.14 11:10:17,007000000 +02:00	(null)	(null)	*****
23.07.14 11:10:17,036000000 +02:00	ksunfy:18678:2937430291	startup	LICENSE_MAX_SESSION = 0

Ein Hinweis ist sicher angebracht: Der Zugriff

auf x\$dbgalertext ist nicht dokumentiert oder durch Oracle empfohlen, so dass sich die Struktur und der Inhalt der Tabelle bei Veränderungen der Version ohne Mitteilung ändern können. Der ebenfalls nicht dokumentierte Weg über den View V\$DIAG\_ALERT\_EXT ist aus Gründen der entstehenden Last und der ggf. trägen Antwortzeiten oft nahezu unbenutzbar.

<sup>6</sup> <http://blog.tanel Poder.com/2009/03/21/oracle-11g-reading-alert-log-via-sql/>

---

## Purging

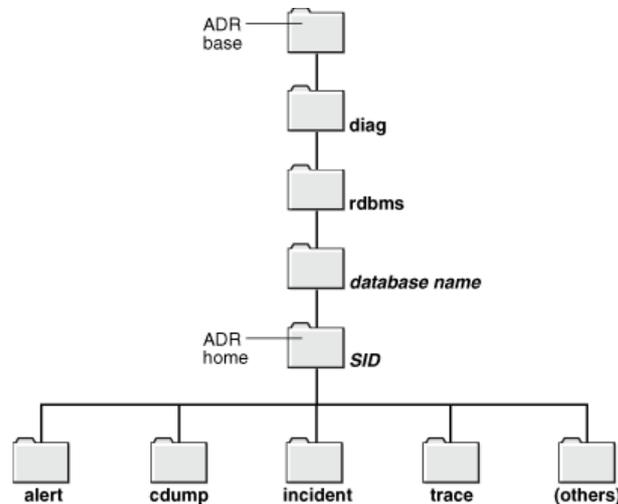
Für Trace- und immer wieder beschnittene XML-Dateien des ADR sorgt Oracle selbständig alle sieben Tage für das Housekeeping nach zuvor definierten Aufbewahrungsregeln<sup>7</sup> (SHORTP\_POLICY und LONGP\_POLICY in ADRCI). Diese greifen jedoch nur für Dateien die nicht mehr berührt wurden - ein fortlaufend geschriebenes Logfile wird daher unbegrenzt weiter wachsen. Deshalb muss sich der verantwortungsbewusste DBA in Eigenverantwortung um die Archivierung bzw. Aufräumung kümmern. Ein Hilfsmittel dazu ist z.B. unter Linux das Tool logrotate.

Das Textfile-Alert Log wird zum Anfügen / Schreiben geöffnet, und jedes Mal wieder geschlossen<sup>8</sup>. Damit ist es jederzeit möglich, das File zu verschieben, umzubenennen oder zu komprimieren ohne dass offene File Handles übrig bleiben, oder eine Sperre den Vorgang längere Zeit aufhält.

---

## Speicherorte von weiteren Diagnoseinformationen

Oracle bietet als Hilfe zur Selbsthilfe mit V\$DIAG\_INFO einen View, der unter Berücksichtigung der konkreten Datenbank-Konfiguration anzeigt, welche Diagnoseinformationen in welchem Pfad liegen.



(Graphik aus der Oracle Dokumentation 12c)

---

<sup>7</sup> <http://www.usn-it.de/index.php/2011/10/05/oracle-manage-the-lifetime-of-trace-files-et-al-with-adrci-purge-interval/>

<sup>8</sup> [https://asktom.oracle.com/pls/asktom/?p=100;11.0::P11\\_QUESTION\\_ID:1352202934074](https://asktom.oracle.com/pls/asktom/?p=100;11.0::P11_QUESTION_ID:1352202934074); leicht nachzuprüfen mit den Unix-Kommandos lsof oder fuser

# Inhalt

Inhaltlich interessant wird das Alert Log natürlich vor allem bei Fehlern - aber auch für das Monitoring oder rückwirkende Recherchen ist es von Nutzen. Im Folgenden ist anhand mehrerer Beispiele dargestellt, welche Informationen sich ablesen lassen. Die Dokumentation sagt zum Inhalt des Alert Log, es schließe unter anderem ein:

- Alle internen Fehler (ORA-600), block corruptions (ORA-1578), und Deadlocks (ORA-60)
- Administrative Operationen wie die SQL\*Plus Kommandos STARTUP, SHUTDOWN, ARCHIVE LOG, und RECOVER
- Diverse Meldungen und Fehler in Bezug auf die Funktionen der Shared-Server- und Dispatcher-Infrastruktur
- Fehler aus dem automatischen Refresh von Materialized Views
- u.v.m.

Einen guten Eindruck vermitteln auch die "MESSAGE\_GROUPS" in x\$dbgalertext (siehe auch oben). Diese beinhalten startup, shutdown, background\_proc, process start, process end, schema\_ddl, admin\_ddl, sqltune, Access Violation usw.

## Beispiel 1 - Startup

Beim Start der Instanz protokolliert Oracle diverse Informationen - zunächst zur Systemumgebung, später zur eigenen Konfiguration. Am Ende gibt es Auskunft über den Verlauf des Starts, u.a. mit einer Meldung für jeden gestarteten Hintergrundprozess. Auch für solche wie den CJK (Scheduler), der erst deutlich verzögert in Erscheinung tritt, sobald die Instanz einige Zeit stabil lief.

```
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options.
ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1
System name: Linux
Node name: kll1101788vm04
Release: 2.6.39-400.212.1.el6uek.x86_64
Version: #1 SMP Fri Dec 20 16:45:16 PST 2013
Machine: x86_64

Using parameter settings in server-side spfile +ORADATA/KLM12102/PARAMETERFILE/spfile.264.853679903
System parameters with non-default values:
processes = 300
sga_max_size = 4G
lock_sga = TRUE
pga_aggregate_limit = 1705M
nls_language = 'AMERICAN'
nls_territory = 'AMERICA'
sga_target = 4G
control_files = '+ORAREDO1/KLM12102/CONTROLFILE/current.257.853672225'
control_files = '+ORAREDO2/KLM12102/CONTROLFILE/current.256.853672227'
control_file_record_keep_time = 31
db_block_size = 8192
compatible = '12.1.0.2.0'
log_archive_format = '%t_%s_%r.dbf'
db_create_file_dest = '+ORADATA'
db_create_online_log_dest_1 = '+ORAREDO1'
db_create_online_log_dest_2 = '+ORAREDO2'
db_recovery_file_dest = '+ORAFRA'
db_recovery_file_dest_size = 4815M
log_checkpoints_to_alert = TRUE
undo_tablespace = 'UNDOTBS1'
inmemory_size = 2G
db_block_checking = 'TRUE'
remote_login_passwordfile = 'EXCLUSIVE'
db_domain = ''
dispatchers = '(PROTOCOL=TCP) (SERVICE=KLM12102XDB)'
local_listener = 'LISTENER_KLM12102'
session_cached_cursors = 500
audit_file_dest = '/u01/app/oracle/admin/KLM12102/adump'
audit_trail = 'DB'
db_name = 'KLM12102'
open_cursors = 300
pga_aggregate_target = 480M
statistics_level = 'TYPICAL'
diagnostic_dest = '/u01/app/oracle'
enable_pluggable_database = TRUE

NOTE: remote asm mode is local (mode 0x1; from cluster type)
Starting background process PMON
Wed Sep 30 15:14:32 2015
PMON started with pid=2, OS id=2535
Starting background process PSP0
Wed Sep 30 15:14:32 2015
PSP0 started with pid=3, OS id=2537
Starting background process VKTM
Wed Sep 30 15:14:33 2015
VKTM started with pid=4, OS id=2539 at elevated (RT) priority
Starting background process GEN0
```

System Info

Non-default parameters

Process startup

Nicht fehlen dürfen oben auch Hinweise wie die Realtime-Priorität von VKTM, dem Prozess für die Zeitbasis. Nachdem die für NOMOUNT notwendigen Prozesse laufen, folgt der MOUNT-Status, d.h. die Anbindung des Control File. Auffällig hier sind die Zugriffe (=mounts) der Automatic Storage Management (ASM) Diskgroups, auf welchen die aktuell nachgefragten Dateien liegen.

```
Wed Sep 30 15:14:36 2015
ALTER DATABASE MOUNT /* db agent *//* {0:0:2} */
Wed Sep 30 15:14:39 2015
NOTE: ASMB mounting group 3 (ORAREDO1)
NOTE: Assigning number (3,0) to disk (/dev/disk/by-id/ASMORAREDO1)
SUCCESS: mounted group 3 (ORAREDO1)
NOTE: grp 3 disk 0: ORAREDO1_0000 path:/dev/disk/by-id/ASMORAREDO1
NOTE: ASMB mounting group 4 (ORAREDO2)
Wed Sep 30 15:14:40 2015
NOTE: dependency between database KLM12102 and diskgroup resource ora.ORAREDO1.dg is established
Wed Sep 30 15:14:40 2015
NOTE: Assigning number (4,0) to disk (/dev/disk/by-id/ASMORAREDO2)
SUCCESS: mounted group 4 (ORAREDO2)
NOTE: grp 4 disk 0: ORAREDO2_0000 path:/dev/disk/by-id/ASMORAREDO2
Wed Sep 30 15:14:40 2015
NOTE: dependency between database KLM12102 and diskgroup resource ora.ORAREDO2.dg is established
Wed Sep 30 15:14:44 2015
Network Resource Management enabled for Process LGWR (pid 2559) for Exadata I/O
Successful mount of redo thread 1, with mount id 328543582
Wed Sep 30 15:14:44 2015
Database mounted in Exclusive Mode
Lost write protection disabled
Completed: ALTER DATABASE MOUNT /* db agent *//* {0:0:2} */
ALTER DATABASE OPEN /* db agent *//* {0:0:2} */
```

## Mount Status

Wurde die Datenbank nicht ordnungsgemäß beendet, muss die Instanz beim Wechseln in den OPEN-Status zunächst ein Crash Recovery durchführen:

```
Completed: ALTER DATABASE MOUNT /* db agent *//* {0:0:2} */
ALTER DATABASE OPEN /* db agent *//* {0:0:2} */
Wed Sep 30 15:14:44 2015
NOTE: ASMB mounting group 1 (ORADATA)
NOTE: Assigning number (1,0) to disk (/dev/disk/by-id/ASMORADATA)
SUCCESS: mounted group 1 (ORADATA)
NOTE: grp 1 disk 0: ORADATA_0000 path:/dev/disk/by-id/ASMORADATA
Wed Sep 30 15:14:44 2015
Ping without log force is disabled
.
Wed Sep 30 15:14:45 2015
Beginning crash recovery of 1 threads
Wed Sep 30 15:14:45 2015
NOTE: dependency between database KLM12102 and diskgroup resource ora.ORADATA.dg is established
Wed Sep 30 15:14:45 2015
parallel recovery started with 3 processes
Wed Sep 30 15:14:45 2015
Started redo scan
Wed Sep 30 15:14:45 2015
Completed redo scan
read 1351 KB redo, 253 data blocks need recovery
Wed Sep 30 15:14:45 2015
Started redo application at
Thread 1: logseq 197, block 1848
Wed Sep 30 15:14:45 2015
Recovery of Online Redo Log: Thread 1 Group 4 Seq 197 Reading mem 0
Mem# 0: +ORAREDO1/KLM12102/ONLINELOG/group_4.261.853673677
Mem# 1: +ORAREDO2/KLM12102/ONLINELOG/group_4.260.853673679
Wed Sep 30 15:14:45 2015
Completed redo application of 0.90MB
Wed Sep 30 15:14:45 2015
Completed crash recovery at
Thread 1: logseq 197, block 4550, scn 48083059
253 data blocks read, 253 data blocks written, 1351 redo k-bytes read
Starting background process TMON
Wed Sep 30 15:14:45 2015
TMON started with pid=35, OS id=2623
Wed Sep 30 15:14:45 2015
LGWR: STARTING ARCH PROCESSES
Starting background process ARC0
Wed Sep 30 15:14:45 2015
ARC0 started with pid=36, OS id=2625
ARC0: Archival started
LGWR: STARTING ARCH PROCESSES COMPLETE
Wed Sep 30 15:14:45 2015
ARC0: STARTING ARCH PROCESSES
Starting background process ARC1
Wed Sep 30 15:14:45 2015
ARC1 started with pid=37, OS id=2627
Starting background process ARC2
Starting background process ARC3
Wed Sep 30 15:14:45 2015
ARC2 started with pid=38, OS id=2629
Wed Sep 30 15:14:45 2015
ARC3 started with pid=39, OS id=2631
ARC1: Archival started
ARC2: Archival started
```

## Begin OPEN

## Crash- / Instance Recovery

Ist das Recovery abgeschlossen, wird die Datenbank für den Regelbetrieb fertiggestellt. Dazu nimmt die Instanz das Redo-System in Betrieb, ebenso gehört die Rückführung der durch den Crash nicht abgeschlossenen Transaktionen durch SMON dazu. Auch hier sieht man immer wieder, wie der Beginn einer Serie von Zugriffen auf ASM Diskgroups mit "mounting" signalisiert wird.

Ganz am Ende, bevor die Datenbank geöffnet wird, startet bei diesem System ein vom DBA definierter STARTUP TRIGGER alle Pluggable Databases die vorhanden sind.

```
Wed Sep 30 15:14:45 2015
ARC3: Archival started
ARC0: STARTING ARCH PROCESSES COMPLETE
Wed Sep 30 15:14:45 2015
Thread 1 advanced to log sequence 198 (thread open)
Thread 1 opened at log sequence 198
  Current log# 5 seq# 198 mem# 0: +ORAREDO1/KLM12102/ONLINELOG/group_5.262.853673685
  Current log# 5 seq# 198 mem# 1: +ORAREDO2/KLM12102/ONLINELOG/group_5.261.853673689
Successful open of redo thread 1
Wed Sep 30 15:14:45 2015
MTTR advisory is disabled because FAST_START_MTTR_TARGET is not set
Wed Sep 30 15:14:45 2015
Network Resource Management enabled for Process LG00 (pid 2563) for Exadata I/O
```

First Log Switch

```
Wed Sep 30 15:14:45 2015
SMON: enabling cache recovery
Wed Sep 30 15:14:45 2015
NOTE: ASMB mounting group 2 (ORAFRA)
NOTE: Assigning number (2,0) to disk (/dev/disk/by-id/ASMORAFRA)
SUCCESS: mounted group 2 (ORAFRA)
NOTE: grp 2 disk 0: ORAFRA_0000 path:/dev/disk/by-id/ASMORAFRA
Wed Sep 30 15:14:45 2015
Archived Log entry 133 added for thread 1 sequence 197 ID 0x1150d49f dest 1:
Wed Sep 30 15:14:45 2015
NOTE: dependency between database KLM12102 and diskgroup resource ora.ORAFRA.dg is established
```

SMON  
cleaning starts

```
Wed Sep 30 15:14:46 2015
[2591] Successfully online Undo Tablespace 2.
Undo initialization finished serial:0 start:4294095920 end:4294096060 diff:140 ms (0.1 seconds)
Verifying minimum file header compatibility (11g) for tablespace encryption..
Verifying 11g file header compatibility for tablespace encryption completed
Wed Sep 30 15:14:46 2015
SMON: enabling tx recovery
Starting background process SMC0
Wed Sep 30 15:14:46 2015
SMC0 started with pid=41, OS id=2635
Wed Sep 30 15:14:46 2015
Database Characterset is AL32UTF8
No Resource Manager plan active
Starting background process IMC0
Wed Sep 30 15:14:46 2015
IMC0 started with pid=44, OS id=2641
replication_dependency_tracking turned off (no async multimaster replication found)
Wed Sep 30 15:14:46 2015
Incremental checkpoint up to RBA [0xc6.3.0], current log tail at RBA [0xc6.76.0]
Starting background process AQPC
Wed Sep 30 15:14:46 2015
AQPC started with pid=45, OS id=2643
```

```
Wed Sep 30 15:14:47 2015
Database Characterset for PDB$SEED is AL32UTF8
Opening pdb PDB$SEED (2) with no Resource Manager plan active
alter pluggable database all open
Database Characterset for PDB1 is AL32UTF8
Opening pdb PDB1 (3) with no Resource Manager plan active
Pluggable database PDB1 opened read write
Completed: alter pluggable database all open
Starting background process CJ00
Completed: ALTER DATABASE OPEN /* db agent *//* {0:0:2} */
```

Pluggable DBs

OPEN Status

Nun folgt der Echtbetrieb im OPEN-Status. Das RDBMS meldet kurz den Status der Fast Recovery Area (FRA), und widmet sich seinen Usern. Das gezeigte System erweitert hier einige Systemtabellen um weitere Intervall-Partitionen, und am Beispiel eines Backups sehen wir wie im Alert Log Veränderungen an Oracle Managed Files (OMF) vorgenommen werden. Hier sind es Backupfiles, aber prinzipiell würde ein Eintrag bezüglich eines hinzugefügten oder entfernten Datafiles eines Tablespace nicht anders aussehen.

Nach einem weiteren Log Switch sehen wir noch ganz unten Einträge zu inkrementellen Checkpoints, die auf die besondere Parametrierung dieses Systems zurückgehen, die ich nur wärmstens empfehlen kann: "log\_checkpoints\_to\_alert=true" gibt für jeden konsistenten Checkpoint nicht nur dessen Uhrzeit, sondern auch die System Change Number (SCN) und die aktuelle Cursorposition im Online Redo Log zurück. Diese Information kann für ein Point-in-Time-Recovery eines Tages entscheidend sein.

```

Wed Sep 30 15:14:53 2015
db_recovery_file_dest_size of 4815 MB is 34.89% used. This is a
user-specified limit on the amount of space that will be used by this
database for recovery-related files, and does not reflect the amount of
space available in the underlying filesystem or ASM diskgroup.
Wed Sep 30 15:14:56 2015
Closing scheduler window
Closing Resource Manager plan via scheduler window
Clearing Resource Manager plan at pdb PDB1 (3) via parameter
Wed Sep 30 15:15:41 2015
TABLE SYS.WRP$_REPORTS: ADDED INTERVAL PARTITION SYS_P638 (2099) VALUES LESS THAN (TO_DATE(' 2015-10-01 01:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLE SYS.WRP$_REPORTS_DETAILS: ADDED INTERVAL PARTITION SYS_P639 (2099) VALUES LESS THAN (TO_DATE(' 2015-10-01 01:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLE SYS.WRP$_REPORTS_TIME_BANDS: ADDED INTERVAL PARTITION SYS_P642 (2099) VALUES LESS THAN (TO_DATE(' 2015-09-30 01:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
Wed Sep 30 15:15:48 2015
Deleted Oracle managed file +ORAFRA/KLM12102/FED9B648382C1EA7E0433263A8C0C3A6/BACKUPSET/2015_06_30/nnndf0_tag20150630t161054_0.270.883757537
Wed Sep 30 15:16:02 2015
Deleted Oracle managed file +ORAFRA/KLM12102/FED9B648382C1EA7E0433263A8C0C3A6/BACKUPSET/2015_06_30/nnndf0_tag20150630t161054_0.267.883757561
Deleted Oracle managed file +ORAFRA/KLM12102/BACKUPSET/2015_06_30/annnf0_tag20150630t161517_0.280.883757719
Deleted Oracle managed file +ORAFRA/KLM12102/AUTOBACKUP/2015_06_30/s_883757722.273.883757723
Deleted Oracle managed file +ORAFRA/KLM12102/BACKUPSET/2015_06_30/ncnnf0_tag20150630t162707_0.260.883758429
Deleted Oracle managed file +ORAFRA/KLM12102/AUTOBACKUP/2015_06_30/s_883758430.279.883758431
Wed Sep 30 15:16:03 2015
Deleted Oracle managed file +ORAFRA/KLM12102/BACKUPSET/2015_06_30/ncnnf0_tag20150630t162714_0.282.883758435
Deleted Oracle managed file +ORAFRA/KLM12102/AUTOBACKUP/2015_06_30/s_883758436.259.883758437
Deleted Oracle managed file +ORAFRA/KLM12102/FEDB87066E473C24E0433263A8C0B658/BACKUPSET/2015_09_28/nnndf0_tag20150928t175110_0.292.891625871
Wed Sep 30 15:19:38 2015
Deleted Oracle managed file +ORAFRA/KLM12102/BACKUPSET/2015_09_28/nnndf0_tag20150928t175110_0.290.891625871
Deleted Oracle managed file +ORAFRA/KLM12102/FEDB87066E473C24E0433263A8C0B658/BACKUPSET/2015_09_28/nnndf0_tag20150928t175110_0.258.891625871
Deleted Oracle managed file +ORAFRA/KLM12102/BACKUPSET/2015_09_28/nnndf0_tag20150928t175110_0.271.891625879
Wed Sep 30 15:19:52 2015
ALTER SYSTEM ARCHIVE LOG
Wed Sep 30 15:19:52 2015
Thread 1 cannot allocate new log, sequence 199
Private strand flush not complete
Current log# 5 seq# 198 mem# 0: +ORARE01/KLM12102/ONLINELOG/group_5.262.853673685
Current log# 5 seq# 198 mem# 1: +ORARE02/KLM12102/ONLINELOG/group_5.261.853673689
Beginning log switch checkpoint up to RBA [0xc7.2.10], SCN: 48185883
Wed Sep 30 15:19:52 2015
Thread 1 advanced to log sequence 199 (LGWR switch)
Current log# 6 seq# 199 mem# 0: +ORARE01/KLM12102/ONLINELOG/group_6.263.853673693
Current log# 6 seq# 199 mem# 1: +ORARE02/KLM12102/ONLINELOG/group_6.262.853673695
Wed Sep 30 15:19:52 2015
Archived Log entry 134 added for thread 1 sequence 198 ID 0x1150d49f dest 1:
Wed Sep 30 15:20:01 2015
Network Resource Management enabled for Process LG01 (pid 2567) for Exadata I/O
Wed Sep 30 15:25:05 2015
Completed checkpoint up to RBA [0xc7.2.10], SCN: 48185883
Wed Sep 30 15:44:54 2015
Incremental checkpoint up to RBA [0xc7.1d9c.0], current log tail at RBA [0xc7.1fce.0]
Wed Sep 30 16:15:04 2015
Incremental checkpoint up to RBA [0xc7.330b.0], current log tail at RBA [0xc7.35ae.0]

```

FRA Status

Admin DDL (Partition Mgmt)

OMF handling  
by Backup Job

Regular operation  
log\_checkpoints\_to\_alert=true

---

## Beispiel 2 - Shutdown

Beim regulären Shutdown stoppt die Instanz nach und nach ihre Prozesse...

```
Wed Sep 30 20:01:48 2015
Shutting down instance (immediate)
Stopping background process IMCO
Stopping background process SMCO
Shutting down instance: further logons disabled
Wed Sep 30 20:01:50 2015
Stopping background process CJQ0
Stopping background process MMNL
Stopping background process MMON
License high water mark = 25
Wed Sep 30 20:01:52 2015
All dispatchers and shared servers shutdown
ALTER DATABASE CLOSE NORMAL
Wed Sep 30 20:01:54 2015
SMON: disabling tx recovery
```

Begin SHUTDOWN  
procedure

stopping processes

... und entlädt sich damit aus dem Speicher.

```
Thread 1 closed at log sequence 199
Successful close of redo thread 1
Completed: ALTER DATABASE CLOSE NORMAL
ALTER DATABASE DISMOUNT
Shutting down archive processes
Archiving is disabled
Wed Sep 30 20:01:57 2015
NOTE: Deferred communication with ASM instance
NOTE: deferred map free for map id 2
Completed: ALTER DATABASE DISMOUNT
Wed Sep 30 20:01:58 2015
ARCH: Archival disabled due to shutdown: 1089
Shutting down archive processes
Archiving is disabled
ARCH: Archival disabled due to shutdown: 1089
Wed Sep 30 20:01:59 2015
Stopping background process VKTM
Shutting down archive processes
Archiving is disabled
Wed Sep 30 20:01:59 2015
NOTE: Shutting down MARK background process
Wed Sep 30 20:02:02 2015
NOTE: Database dismounted; ASMB process exiting
NOTE: ASMB clearing idle groups before exit
Stopping background process RBAL
Wed Sep 30 20:02:07 2015
Instance shutdown complete
```

Dismounting Instance

SHUTDOWN  
procedure finished

## Beispiel 3 - Fehler

Am Beispiel eines Deadlock lässt sich gut erläutern, wie die "entscheidende Maßnahme" zur Bereinigung einer Situation zwar im Alert Log protokolliert wird . . .

Wed Sep 30 22:30:02 2015  
Incremental checkpoint up to RBA [0xc7.1c9c6.0], current log tail at RBA [0xc7.1cb53.0] **Error and corresponding Trace File**  
Wed Sep 30 22:31:43 2015  
**ORA-00060: Deadlock detected.** See Note 60.1 at My Oracle Support for Troubleshooting ORA-60 Errors. More info in file [/u01/app/oracle/diag/rdbms/klm12102/KLM12102/trace/KLM12102\\_ora\\_15102.trc.](#)

. . . jedoch die umfangreichen weiterführenden Informationen in einem Trace File besser aufgehoben sind, und dort mit jedem Texteditor analysiert werden können.

```
Trace file /u01/app/oracle/diag/rdbms/klm12102/KLM12102/trace/KLM12102_ora_15102.trc
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Advanced Analytics
and Real Application Testing options
ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1
System name: Linux
Node name: k111101788vm04
Release: 2.6.39-400.212.1.el6uek.x86_64
Version: #1 SMP Fri Dec 20 16:45:16 PST 2013
Machine: x86_64
Instance name: KLM12102
Redo thread mounted by this instance: 1
Oracle process number: 28
Unix process pid: 15102, image: oracle@k111101788vm04
```

```
*** 2015-09-30 22:31:42.708
*** SESSION ID:(17.32453) 2015-09-30 22:31:42.708
*** CLIENT ID:() 2015-09-30 22:31:42.708
*** SERVICE NAME:(pdb1) 2015-09-30 22:31:42.708
*** MODULE NAME:(SQL Developer) 2015-09-30 22:31:42.708
*** CLIENT DRIVER:(jdbcthin) 2015-09-30 22:31:42.708
*** ACTION NAME:() 2015-09-30 22:31:42.708
*** CONTAINER ID:(3) 2015-09-30 22:31:42.708
```

### Error

```
*** 2015-09-30 22:31:42.708
DEADLOCK DETECTED ( ORA-00060 )
See Note 60.1 at My Oracle Support for Troubleshooting ORA-60 Errors
[Transaction Deadlock]
```

The following deadlock is not an ORACLE error. It is a deadlock due to user error in the design of an application or from issuing incorrect ad-hoc SQL. The following information may aid in determining the deadlock:

### Details

```
Deadlock graph:
-----Blocker(s)-----
Resource Name process session holds waits process session holds waits
TX-0008001F-0000180B-00000000-00000000 28 17 X 31 379 X
TX-000A0012-00001AD6-00000000-00000000 31 379 X 28 17 X

session 17: DID 0001-001C-0000005A session 379: DID 0001-001F-00000011
session 379: DID 0001-001F-00000011 session 17: DID 0001-001C-0000005A

Rows waited on:
Session 17: obj - rowid = 0001668F - AAaWAPAAJAAAdg/AAB
(dictionary objn - 91791, file - 9, block - 120895, slot - 1)
Session 379: obj - rowid = 0001668F - AAaWAPAAJAAAbsjAAA
(dictionary objn - 91791, file - 9, block - 113443, slot - 0)
```

Im Falle des ORA-0060 besteht diese Information aus Hinweisen zu den beteiligten Transaktionen, wie Row-IDs und Session-Details.

# Praxistip

Auf Umgebungen die ohne klassische Unix-Tools auskommen müssen, liefert das Kommandozeilen-Tool des ADR mit dem schönen Namen ADRCI eine Möglichkeit, das Alert Log fortlaufend zu betrachten, ganz so wie das "tail" im Unix-Umfeld anbietet. Das folgende Beispiel zeigt die richtige Anwendung von "show alert -tail 10 -f" ("zeige die letzten 10 Zeilen des Alert Log und dann fortlaufend jede Zeile, die neu dazu kommt").

```
[oracle@kl11101788vm04 ~]$ (KLM12102) adrci
```

```
ADRCI: Release 12.1.0.2.0 - Production on Wed Sep 30 22:58:07 2015
```

```
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights reserved.
```

```
ADR base = "/u01/app/oracle"
```

```
adrci> show homes
ADR Homes:
diag/clients/user_oracle/host_3653034631_82
diag/asm/+asm/+ASM
diag/crs/kl11101788vm04/crs
diag/rdbms/klm12102/KLM12102
diag/tnslsnr/kl11101788vm04/listener
adrci> set home diag/rdbms/klm12102/KLM12102
```

```
adrci> show alert -tail 10 -f
2015-09-30 21:59:58.812000 +02:00
Opening pdb PDB$SEED (2) with no Resource Manager plan active
alter pluggable database all open
Database Characterset for PDB1 is AL32UTF8
2015-09-30 22:00:01.217000 +02:00
Opening pdb PDB1 (3) with no Resource Manager plan active
Pluggable database PDB1 opened read write
Completed: alter pluggable database all open
Starting background process CJQ0
Completed: ALTER DATABASE OPEN
CJQ0 started with pid=42, OS id=13570
2015-09-30 22:00:04.049000 +02:00
Shared IO Pool defaulting to 96MB. Trying to get it from Buffer Cache for process 13346.
```

```
=====
Dumping current patch information
=====
```

```
No patches have been applied
=====
```

```
2015-09-30 22:00:05.279000 +02:00
db_recovery_file_dest_size of 4815 MB is 34.02% used. This is a
user-specified limit on the amount of space that will be used by this
database for recovery-related files, and does not reflect the amount of
space available in the underlying filesystem or ASM diskgroup.
```

ADRCI Tail

Select Oracle Home

show alert -tail 10 -f

## Was fehlt?

Eine ganze Reihe operativer Informationen kann der DBA aus dem Alert Log ablesen. Allerdings finden sich etliche für den Betrieb relevante Informationen NICHT in diesem File. Die folgende Liste ist keinesfalls vollständig, sondern soll einen Eindruck über anderswo verfügbare, teils reichhaltige Diagnosedaten vermitteln. In Klammern sind mögliche Quellen für derartige Daten angegeben.

- Blockierungen, die keine Deadlock sind (v\$lock)
- Angemeldete User, Historie von SQL Statements (v\$session, Audit Trail)
- Gründe für Erreichen der Prozesslimits (v\$session, v\$process)
- Clientseitige Verbindungsfehler (Logs auf dem Client)
- Probleme beim Verbindungsaufbau (Listener-Logfile im ADR)
- Performance-Vorfälle (Oracle Wait Interface, AWR, Statspack)
- Start, Verbleib und Stop von Scheduler-Jobs (dba\_scheduler-Views)
- Vorfälle der Oracle Clusterware im RAC (crsd.log, ocspd.log, evmd.log, Alert Log der Clusterware)
- ASM-Vorfälle (eigenes Alert Log)

# Zusammenfassung

Das Alert Log ist ein unverzichtbarer Bestandteil für jeden, der sich ein Bild der "Lage" im System machen möchte. Es zeigt Fehler, aktuelle Meldungen und vielerlei was mit dem administrativen Status der Instanz zusammenhängt. Jedoch sind weiterführende Details von Störungen sowie viele für die Performance und die Aktivität des Systems relevante Informationen daraus nicht erhältlich und müssen aus Trace Files und den dynamischen Views der Instanz extrahiert werden. Richtig angewendet und verstanden, liefert das Alert Log aber unverzichtbare Teile zum Lagebild und ermöglicht in vielen Fällen erst Maßnahmen von der Rekonstruktion von Problemen der Vergangenheit bis hin zur Vermeidung von Fehlern der Zukunft.

## How to contact me

Martin Klier  
Managing Partner / Database Technology  
at Performing Databases



Wiesauer Strasse 27  
95666 Mitterteich  
GERMANY

Phone: +49 9633 631  
Fax: +49 9633 4199  
Mobile: +49 170 PERFDB7 (+49 170 7373327)

Email: martin.klier@performing-db.com  
Twitter: @MartinKlierDBA

## About my company

We are "performing databases" - and we are keen on caring for you: "Your reliability. **Our concern.**" Specialized in services around database technology, supreme performance and the highest availability of your systems is our ultimate and most important goal.



Performing Databases GmbH  
Your reliability. **Our concern.**

<http://www.performing-databases.com>

Geschäftsführer: Martin Klier, Benedikt Nahlovsky  
Sitz der Gesellschaft: Mitterteich  
HRB 4350, Amtsgericht Weiden i.d.Opf  
VAT ID: DE815489841

