

Standby-Datenbanken aktiv nutzen

Oracle Active Data Guard

Mit der Option zur Oracle Database 11g Enterprise Edition werden ressourcenintensive Aktivitäten von einer Produktionsdatenbank auf eine oder mehrere synchronisierte Standby-Datenbanken verlagert. **Sebastian Solbach**

Auf einen Blick

Inhalt

Die Zusatzoption Oracle Active Data Guard macht einen Nur-Lese-Zugriff auf Standby-Datenbanken verfügbar und realisiert darüber beispielsweise Abfragen, Sortierungen, Berichte sowie den Web-basierten Zugriff, wobei zugleich die ankommenden Änderungen von der Produktionsdatenbank angewendet werden. Die Option unterstützt ferner die Erstellung schneller inkrementeller Sicherungen und bietet Vorzüge von Hochverfügbarkeit und Disaster-Schutz im Fall geplanter oder ungeplanter Ausfälle in der Produktion.

Plattform

Oracle 11g/11g R2

Technik/Anwendung

Datenbank-Administration und -Programmierung

Voraussetzungen

Oracle-Datenbank 11g, Oracle-Standby-Datenbank/ Data Guard

Autor

Dipl.-Betriebswirt (BA) Sebastian Solbach ist seit 2001 als Systemberater bei der Oracle Deutschland GmbH beschäftigt, seit 2007 mit dem Schwerpunkt Oracle Maximum Availability Architecture (MAA) der Oracle-Datenbank.

Normalerweise befindet sich eine Standby-Datenbank im ständigen Recovery-Modus und ist – zumindest vor Oracle 11g – nur im Fehlerfall oder durch Unterbrechung des Recovery-Mechanismus zu verwenden. Mit Active Data Guard (ADG) bietet Oracle 11g die Möglichkeit, auf das Standby-System lesend zuzugreifen, während dort gleichzeitig das Recovery läuft.

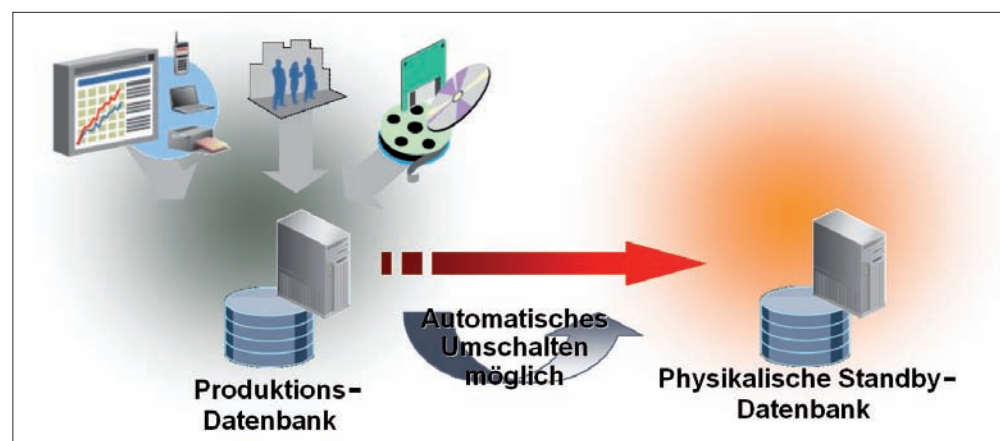
Oracle Data Guard

Data Guard ist der Oracle-Name für eine oder mehrere Standby-Datenbanken samt einer gut funktionierenden Disaster-Recovery-Lösung. Eine Standby-Datenbank basiert auf dem Standard-Recovery-Mechanismus, der an einem anderen Ort (in einem anderen Brandschutzabschnitt oder auf einem anderen Kontinent) ständig durchgeführt wird, um im Fehlerfall sofort auf die Standby-Location umzuschalten. Als Disaster-Recovery-Lösung wird eine Lösung auf Basis von Software und/oder Hardware bezeichnet, mit der Sie sich gegen alle Arten von Ausfällen absichern. Dies sind zum Beispiel Fehler, die durch den Ausfall einzelner Komponenten oder sogar durch den Ausfall eines gesamten Rechenzentrums – der sogenannten Site – auftreten. Zusätzlich sichern Sie sich mithilfe von Data Guard auch gegen Fehler ab, die durch den Benutzer oder durch Software verursacht werden.

Die einfachste Art des Disaster-Recovery ist eine Datensicherung (Backup), die aber nicht immer hundertprozentige Datenaktualität (Zero Data Loss) garantieren kann. Fallen alle Platten der produktiven Datenbank aus, sind die Änderungen seit dem letzten Backup verloren. Eine

andere Schwierigkeit beim Recovery ist die benötigte Zeit, bis die Datenbank wieder verfügbar ist. Diese Zeit, auch als „Mean Time to Recover“ (kurz MTTR) bezeichnet, muss für große und wichtige Datenbanken in einem vertretbaren Ausmaß liegen. Dennoch bilden die grundlegenden Backup- und Restore-Mechanismen die Grundlage für Data Guard. Oracle Data Guard übernimmt und automatisiert hierbei den Betrieb einer oder mehrerer transaktionskonsistenter Kopien (Standby) der Produktiv-Instanz, im weiteren Verlauf als Primary bezeichnet. Wenn diese Produktiv-Instanz ausfällt, kann eine Standby-Datenbank manuell oder automatisch aktiviert werden, um die Rolle der Primary zu übernehmen (Bild 1). Voraussetzungen für Data Guard sind, neben dem Archivelog-Modus der Datenbank, der Einsatz des gleichen Datenbank-Releases und Patch-Levels auf Ziel- und Ausgangsplattform. Zur Administration von Standby-Datenbanken ist SQL*Plus einsetzbar. Bevorzugt ist der Einsatz des Data Guard Broker. Dieser übernimmt die Verwaltung und Parametrisierung sowohl der Primär- als auch der Standby-Datenbanken. Über das Kommandozeileninterface des Data Guard Broker (DGMGRL) sind viele Aktionen mit nur einem Befehl durchführbar:

- Umschalten vom Primär-System auf das Standby-System (Switchover), damit zum Beispiel auf dem Primär-System Wartungsarbeiten durchführbar sind. Hiermit wird die Standby-Datenbank zur produktiven Instanz. Es kann jederzeit wieder zurückgeschaltet werden.
- Aktivieren der Standby-Datenbank im Fehlerfall (Failover) – ein Zurückschalten ist nicht ohne Weiteres möglich. ▶



Funktionsweise Oracle Data Guard (Bild 1)

Listing 1: Abfrage zum ADG-Status

```
SELECT 'Using Active Data Guard' ADG
FROM V$MANAGED_STANDBY M,
V$DATABASE D WHERE M.PROCESS LIKE
'MRP%' AND D.OPEN_MODE = 'READ ONLY';

ADG
-----
Using Active Data Guard
```

■ Wiederherstellen einer zuvor ausgefallenen produktiven Datenbank (Reinstat).

Eine grafische Administration über das Enterprise Management Grid Control, das Management-Werkzeug für Oracle-Datenbanken, ist ebenfalls möglich, setzt aber auf den Funktionalitäten des Data Guard Broker auf. Alle Änderungen, die in einer Oracle-Daten-

bank erfolgen, werden vom Logwriter-Prozess (LGWR) in den Redolog-Dateien gespeichert. Genau diese Informationen werden von der Standby-Seite benötigt, um die entsprechenden Änderungen nachzuziehen. Dieses Nachfahren wird als Apply-Prozess bezeichnet, der separat vom Transport der Redolog-Dateien zur Standby-Seite angehalten und gestartet werden kann. Vor Oracle 9i konnten Sie diese Informationen nur in vollständig archivierte Logs transferieren. In den aktuellen Versionen werden diese Informationen direkt vom Prozess Logwriter, der auch für das Schreiben der Redolog-Dateien verantwortlich ist, an das Standby-System gesendet. Dies kann sofort nach jeder Änderung (synchron) oder in kleineren Paketen (asynchron) erfolgen und ist insbesondere deshalb interessant, weil der Transportmechanismus Auswirkungen auf Active Data Guard hat (zum Beispiel darauf, wie zeitaktuell die Informationen sind, die auf dem Standby-System abfragbar sind). Reine Standby-Systeme werden heute schon fast als Luxus betrachtet. Da im Normalfall ein Disaster selten eintritt und auch geplante Wartungsarbeiten an Rechnern nicht zur Tagesordnung gehören, möchte man die Ressourcen der Standby-Systeme ebenfalls im normalen Alltag verwenden. Es gibt daher bei Unternehmen immer wieder Überlegungen, nur lesende (Read-only-)Applikationen, wie rechenintensive Auswertungen, auf das Standby-System zu verlagern. Mit Oracle 10g bewerkstelligen Sie dies nur, indem Sie den Apply-Prozess anhalten und die Stand-

by-Datenbank nur zum Lesen (read-only) öffnen. Dies hat zwei entscheidende Nachteile: Erstens werden nach dem Öffnen der Datenbank keine Änderungen mehr eingespielt und somit sind Berichte nur auf nicht hundertprozentig aktuellen oder statischen Daten ausführbar. Ein echtes Berichtswesen auf aktuellen Daten ist so nicht zu bewerkstelligen. Zweitens werden die MTTR-Zeiten dadurch ebenfalls länger, da im Fehlerfall erst einmal alle Änderungen nachzuziehen sind, die während der Unterbrechung des Apply-Prozesses auf der Primary durchgeführt wurden. Je nach angefallenen Änderungen kann dies mehrere Stunden dauern. Mit Oracle 11g Active Data Guard, einer kostenpflichtigen Option der Oracle-Datenbank-Enterprise-Edition, gehören diese Einschränkungen der Vergangenheit an.

Active Data Guard aktivieren

Voraussetzung für ADG ist eine klassische Standby-Datenbank [1]. Im Unterschied zur klassischen Standby läuft aber bei Active Data Guard der Apply-Prozess im Hintergrund weiter, während die Standby-Datenbank read-only geöffnet ist. Hierzu werden mit dem Data Guard Broker Command Line Interface (DGMGRL) der Apply-Prozess und das Recovery angehalten (*muc* ist der Name der Standby-Datenbank):

```
DGMGRL> EDIT DATABASE 'muc' SET STATE
= 'APPLY-OFF';
```

Dann wird die Datenbank über SQL*Plus read-only geöffnet:

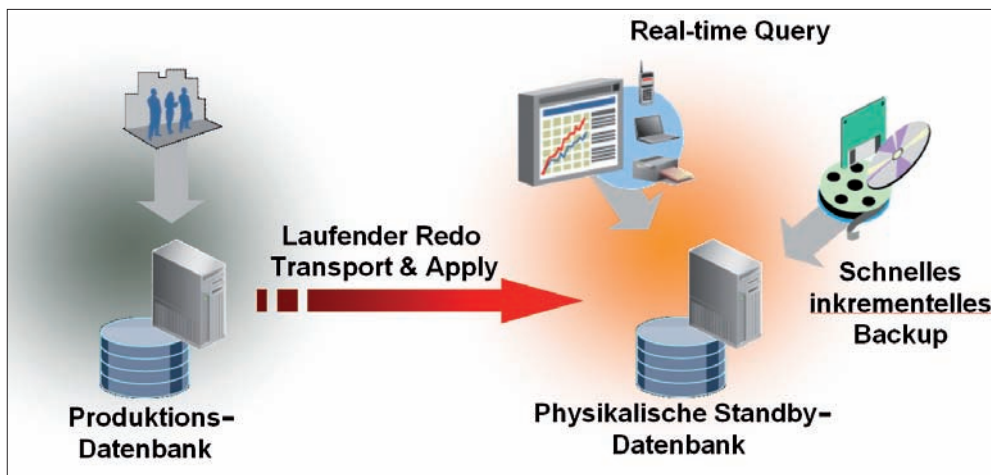
```
SQL> ALTER DATABASE OPEN READ ONLY;
```

Und zum Abschluss wird mit DGMGRL das automatische Recovery wieder eingeschaltet:

```
DGMGRL> EDIT DATABASE 'muc' SET STATE
= 'APPLY-ON';
```

SCHWERPUNKT

Oracle Active Data Guard mit Analysen in Echtzeit (Bild 2)



Nun ist Active Data Guard aktiviert und erlaubt Auswertungen und Analysen nahezu auf Echtzeit-Daten (Bild 2). Leider merkt sich in der Version (11.1.0.7) der Data Guard Broker nicht den Status der Datenbank, sodass beispielsweise nach einem Switchover Active Data Guard erneut zu aktivieren ist. Die Query in Listing 1 ermittelt auf der Standby-Seite, ob Active Data Guard aktiviert ist.

Mögliche Operationen

Bei einer Standby-Datenbank, die mit ADG geöffnet ist, sind folgende Operationen erlaubt:

- *SELECT*-Befehle, inklusive Befehle, die den Tablespace *TEMP* für Sortieroperationen benötigen,
- Verwendung von *ALTER-SESSION*-, *ALTER-SYSTEM*- und *SET-ROLE*-Anweisungen,
- Aufruf von Stored Procedures,
- Verwendung von Datenbank-Links zum Schreiben in andere Datenbanken oder Ausführen anderer Stored Procedures,
- Verwendung von *SET TRANSACTION READ ONLY* zur Lesekonsistenz,
- komplexe Abfragen (zum Beispiel *GROUPING SET* und *WITH*-Klausel).

Die Standby-Datenbank unterliegt denselben Restriktionen wie eine normale read-only geöffnete Datenbank:

- kein DML (Data Manipulation Language, mit Ausnahme einfacher *SELECT*-Anweisungen) oder DDL (Data Definition Language),
- keine Abfragen von lokalen Sequenzen,
- kein DML auf lokale temporäre Tabellen.

Reine Analysewerkzeuge, die nur die erlaubten Operationen durchführen, greifen ohne Änderungen auf ADG zu und entlasten die Primär-Seite. Leider bestehen reine Berichtsanwendungen oft nicht ausschließlich aus nur lesenden Transaktionen. Häufig werden gerade Login und andere Daten aus Auditing-Gründen protokolliert. Dies bewirkt, dass diese Applikationen nicht einfach auf die Standby-Datenbank zugreifen können, da schon das Login eine Fehlermeldung verursachen würde. Aber auch diese Art von Applikationen ist – mit kleinen Änderungen am Datenmodell – dazu geeignet, optimal mit ADG zusammenzuarbeiten. Alle Änderungen erfolgen dabei auf der Primär-Seite. Als Erstes wird ein Datenbank-Link benötigt, der die schreibenden Zugriffe auf eine andere Instanz oder die Primär-Seite umleitet und wie folgt angelegt werden kann:

```
SQL> CREATE PUBLIC DATABASE LINK
primary USING 'stg';
Database link created.
```

Danach werden die Tabellen umbenannt, auf die ein Update erfolgen soll:

```
SQL> rename
emp to emp_table;
Table renamed.
```

Der eigentliche Insert erfolgt über ein Synonym, das über den Datenbank-Link auf die Primär- oder eine andere Datenbank verweist:

```
SQL> create synonym
emp for
emp_table@primary;
Synonym created.
```

Auf der Primär-Seite zeigt der Datenbank-Link (über den Eintrag in der Datei *TNSNAMES.ora*) auf sich selber. Jetzt funktioniert auch ein Insert auf der Standby-Seite:

```
insert into emp
VALUES(8000, 'STAND>
BY', 'DATABASE', NULL,
SYSDATE, 0, 0, 10);
commit;
```

Falls die benötigten Updates nicht für die Applikation wichtig und nur von temporärer Natur sind, empfiehlt es sich, diese nicht zurück auf die Primär zu leiten, sondern hierfür eine Scratch-Datenbank auf dem Standby-Host zu verwenden. Die Scratch-Datenbank enthält nur die Objekte, auf denen ADG ein Update machen muss. Dieser Ansatz hat mehrere Vorteile:

- keine Auswirkungen auf die Datenbankperformance der Primär,
- keine Redo-Informationen, die zusätzlich transferiert werden,
- eine zu vernachlässigende Last auf dem Netzwerk, da die lokale Scratch-Datenbank über IPC angebunden werden kann,
- und damit kaum negativen Einfluss durch einen Netzwerk-Roundtrip für die Standby-Datenbank.

Bei der Verwendung von Datenbank-Links ist zu berücksichtigen, dass Informationen wie der Session State nicht mit transferiert werden, was unter Umständen für Audit-Daten notwendig ist. Hierzu und zur Problematik, Prozeduren remote aufzurufen, finden Sie Tipps unter [2].

Datenaktualität

Wie schnell aktuelle Daten bei der Standby-Datenbank zu sehen sind, hängt von mehreren Faktoren ab: Einerseits spielen das Aufkommen der Redolog-Informationen und die Netzwerk- ▶

Listing 2: Tabelle/SCN auf der Primary

```
SQL> create table emp2 as select * from emp;
Table created.

SQL> select current_scn from v$database;

CURRENT_SCN
-----
654921
```

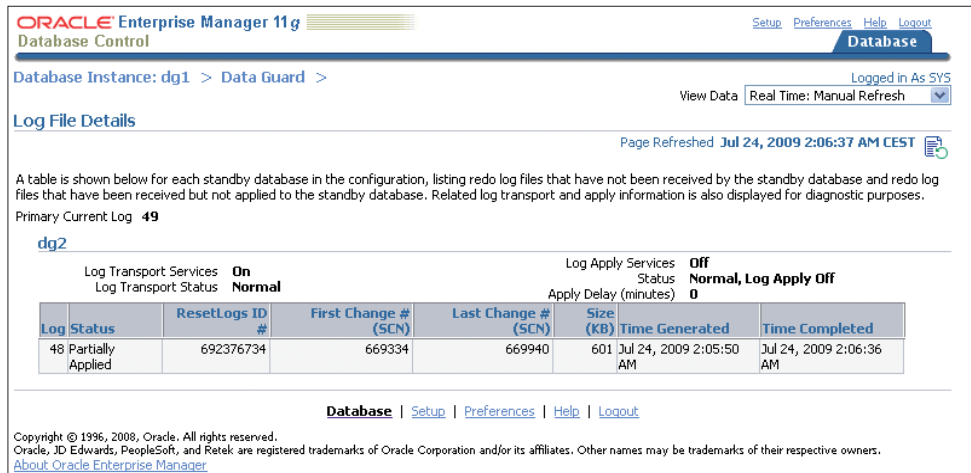
Listing 3: SCN/Tabellen auf der Standby

```
SQL> select current_scn from v$database;

CURRENT_SCN
-----
654885

SQL> select * from tab;

TNAME          TABTYPE  CLUSTERID
-----
BONUS          TABLE
DEPT           TABLE
EMP            SYNONYM
EMP_TABLE      TABLE
SALGRADE       TABLE
```



Database Control Data Guard: Statistiken (Bild 3)

bandbreite und -latenz eine Rolle, andererseits aber auch der gewählte Transportmechanismus (synchron/asynchron) und die Performance des Recovery. Um herauszufinden, wie viel Zeit zwischen dem eigentlichen Insert und den sichtbaren Daten auf der Standby liegt, fragen Sie die aktuelle „System Change Number“ (kurz SCN) der Primary und der Standby ab:

```
SQL> select current_scn from
v$database;
CURRENT_SCN
-----
          654892
```

Während bei der Primär-Datenbank die SCN die aktuelle Transaktions-ID angibt, ist dies auf der Standby-Seite die sogenannte Query-SCN, das heißt die SCN, mit der ein transaktionskonsistenter Zustand der Daten angezeigt wird. Solange die Query-SCN kleiner ist als die aktuelle SCN, werden die Daten noch nicht auf der Standby-Seite angezeigt. Das Beispiel in den Listings 2 und 3 verdeutlicht dieses Verhalten. Ist die

Query-SCN noch nicht auf diesem Wert, so erscheint auch die angelegte Tabelle noch nicht. Vergleichen Sie den Zeitstempel der beiden SCNs (von der Primär-Seite aus) mit einem Statement entsprechend Listing 4, wissen Sie letztendlich, wie viel Zeit zwischen einer aktuellen Abfrage und der Verfügbarkeit der Daten in der Standby-Datenbank liegt. Für das Beispiel wurde der Apply-Prozess angehalten, da normalerweise der Zeitversatz weit unterhalb einer Sekunde liegt. Sind die Zeiten

dennoch recht hoch, können Sie weitere Nachforschungen anstellen, um schließlich herauszufinden, an welchen Stellen die Performance verbesserbar ist. Ist es die Netzwerklatenz? Oder vielleicht doch eher der Apply-Prozess, der getuned werden sollte? Diesen Zeitversatz (die sogenannten Lag-Informationen) führen die Data-Guard-Statistiken mit, die Daten können jederzeit auf der Standby-Seite abgefragt werden. Der Transport-Lag deutet auf Probleme beim Transport und damit häufig auf Probleme mit dem Netzwerk hin:

```
SELECT name, value FROM
v$dataguard_stats WHERE name =
'transport lag';
```

```
NAME          VALUE
-----
transport lag +00 00:00:00
```

Redo Apply Lag deutet hingegen auf ein langsames Recovery der Standby (oder wie in meinem Falle ausgeschaltetes) hin:

Neu mit Active Data Guard 11g R2

- **Unterstützung einer größeren Anzahl von Standby-Systemen:** Mit Oracle Active Data Guard (ADG) ist die Anzahl der Standby-Systeme in 11g Release 1 auf neun beschränkt. Der Data Guard Broker des neuen 11g Release 2 bietet eine Unterstützung von bis zu 30 Standby-Systemen. Es bleibt dem Benutzer natürlich weiterhin freigestellt, für eine verbesserte Skalierung für diese Systeme Real Application Cluster (RAC) zu verwenden.
- **Real Time Query Status im Data Guard Broker:** Der Data Guard Broker behält im aktualisierten Release 2 den Real-Time-Query-Status (ADG) nach einem Switchover bei. In der Vergangenheit mussten in der Standby-Datenbank die Aktivierungsschritte erneut durchgeführt werden. Zur Aktivierung reicht übrigens im neuen Release ein `alter database open` aus, um

- bei einer Data-Guard-Broker-geführten Umgebung Active Data Guard zu aktivieren.
- **Quality of Service:** Sollte Datenaktualität wichtig sein, kann mit 11g Release 2 eine Art Quality of Service für Abfragen hinterlegt werden. Wird der maximale Zeitunterschied der Standby zur Primary, der für eine Session festgelegt wird, nicht eingehalten, wird anstatt der Daten eine Fehlermeldung an die Applikation zurückgegeben. Ist für eine Applikation für eine bestimmte Abfrage die Datenaktualität immens wichtig, kann aus der Applikation heraus die Synchronisation mit der Primär-Datenbank angestoßen werden. Mehr Service bietet auch das Feature von Data Guard, korrupte Blöcke einer Datenbank direkt, online und ohne Eingriff des Administrators von Standby-Umgebungen wiederherzustellen.

```
SELECT name, value FROM
v$dataguard_stats WHERE NAME =
'apply lag';
```

| NAME | VALUE |
|-----------|--------------|
| apply lag | +00 00:03:31 |

Größere Probleme oder Lags werden auch vom Enterprise Manager Database Control, dem Managementwerkzeug für Oracle-Datenbanken, festgestellt (Bild 3). Allerdings können mit dem Database Control nur Statistiken eingesehen werden, für eine Administration ist der größere Bruder (das Enterprise Manager Grid Control) notwendig, da bei Data Guard mehrere Datenbanken administriert werden und das Database Control nur eine Datenbank managen kann.

Schnelleres Inkrementelles Backup möglich

Da es sich bei einer Standby-Datenbank um eine blockidentische Kopie der Primär-Datenbank handelt, entlasten Sie das Primär-System dadurch, dass das normale Backup auf die Standby-Seite verlagert wird. Einzig und allein ein schnelles inkrementelles Backup ist ohne Active Data Guard nicht möglich. Für ein schnelles inkrementelles Backup schreibt ein Prozess der geöffneten Datenbank in ein Block Change Tracking File, das wie folgt aktiviert wird:

```
SQL> alter database enable block
change tracking using file
'/opt/oracle/oradata/blockchtrack';
```

In diesem File wird für das Backup-Tool RMAN festgehalten, welche Blöcke der Datenbank seit dem letzten Backup geändert wurden. Dies ermöglicht ein 20-fach schnelleres Backup gegenüber dem herkömmlichen inkrementellen Backupverfahren. Bei diesem muss RMAN die Table-

Listing 4: Zeitunterschied der SCNs abfragen

```
SQL> select scn_to_timestamp((select
current_scn from v$database))-scn_to_timestamp((select
current_scn from v$database@duc)) as Zeitunterschied from dual;
```

| ZEITUNTERSCHIED |
|-------------------------------|
| +000000000 00:03:25.000000000 |

spaces nach geänderten Blöcken durchsuchen. Da aber die Datenbank beim herkömmlichen Data Guard nicht geöffnet ist – sie befindet sich im sogenannten Mount-Status – ist dies auch nur mit Active Data Guard möglich.

Hinweis: Für den Einsatz des schnellen inkrementellen Backups auf der Standby-Seite ist der Bundle Patch #1 für Data Guard 11.1.0.7 dringend empfohlen.

Mehr Funktionalität durch Active Data Guard

Active Data Guard erweitert die Funktionalität jeder Standby-Datenbank immens. Nicht nur, dass die Primär-Datenbank durch das Auslagern von Berichten entlastet werden kann, sondern Sie behalten auch Ihre 100 Prozent Datensicherheit, falls Sie im sogenannten Zero-Data-Loss-Modus fahren.

Die Auswirkungen auf die Performance für diese Funktionalität sind übrigens marginal im Vergleich zur Performance einer normal gemounteten Standby. Aus diesem Grunde sollte eigentlich keine Datenbank, bei der eine Disaster-Recovery-Lösung benötigt wird, ohne Active Data Guard betrieben werden. **[am]**

[1] Aufbau einer Standby-Datenbank;
www.oracle.com/global/de/community/dbadmin/tipps/aufbau_standby/

[2] Active Data Guard 11g Best Practices;
www.oracle.com/technology/deploy/availability/pdf/maa_wp_11gr1_activedataguard.pdf

**WIR REDEN NICHT LÄNGER
VON DER KRISE,
WIR HANDELN**



KUTTIG

Consultingleistungen aus
unserem ORACLE-Portfolio **0,99 €**

je Minute, gültig bis 31.12.2009

BERATUNGSOFFENSIVE FÜR DEN MITTELSTAND

Start auf der DOAG 2009, 17.–19. November

Wir sind Aussteller und Referenten zu den Themen:

ORACLE Beehive, WebCenter Spaces und Information Rights Management.

Reservieren Sie sich Ihren Consultant unter helmut.heinrich@kuttig.com oder Tel: 02241 – 9833-417.