

Oracle Data Guard und RMAN - das perfekte Team

Claudia Hüffer
Oracle Deutschland B.V. & Co. KG
Hamburg

Schlüsselworte:

Data Guard, Recovery Manager

Einleitung

Oracle Data Guard ist DIE Disaster-Recovery-Lösung von Oracle. Eingeführt mit Oracle 7.3 ist die Oracle Standby Datenbank Technologie von Version zu Version stets weiterentwickelt worden und seit Oracle 9i unter dem Namen Data Guard bekannt.

Der Recovery Manager (RMAN) wurde mit Oracle 8.0 eingeführt und seither kontinuierlich im Funktionsumfang erweitert.

Der Recovery Manager spielt im Zusammenhang mit Data Guard Umgebungen eine zentrale Rolle. Dies beginnt beim erstmaligen Aufbau der Data Guard Umgebung, bei der seit Oracle 11g auch ohne Zwischenspeicherung eines Backups direkt eine Standby Datenbank mit RMAN aufgebaut werden kann. Das bestehende Backup/Recovery Konzept der Produktionsdatenbank kann mit geeigneten RMAN Kommandos dahingehend geändert werden, dass das Vorhandensein eventuell noch für eine Standby Datenbank benötigter archivierter Redo Logs beim automatischen Löschvorgang berücksichtigt wird. In Verbindung mit Oracle 11g R2 Active Data Guard ist der Recovery Manager in der Lage, bei auf der Primärseite erkannten Blockkorruptionen intakte Blöcke von der Standby-Seite zu lesen, so dass eine automatische Reparatur von korrupten Blöcken erfolgen kann.

Der vorliegende Beitrag gibt zunächst einen kurzen Überblick über die Oracle Data Guard und Recovery Manager Architektur. Im Anschluss daran wird der Einsatz vom RMAN in Data Guard Umgebungen an unterschiedlichen Beispielen erläutert. Diese umfassen das erstmalige Aufbauen einer Standby-Datenbank, Backup/Recovery im Data Guard Umfeld, Automatische Block-Reparatur in Active Data Guard Umgebungen und das automatische Löschen von archivierten Redo-Logs in Data Guard Umgebungen.

1. Oracle Maximum Availability Architecture (MAA)

Die Oracle Datenbank bietet zahlreiche Funktionalitäten, mit denen sich hochverfügbare aber auch fehlertolerante Systeme aufbauen lassen. Die verschiedenen Funktionalitäten adressieren dabei die unterschiedlichsten Ausfallszenarien. Ausfallzeiten von Applikationen lassen sich in geplante und ungeplante Ausfallzeiten unterteilen. Geplante Ausfallzeiten werden in der Regel durch Änderungen an Tabellendefinitionen, der Applikation, der Betriebssystemversion und/oder der verwendeten Software-Version verursacht. Funktionalitäten wie Online Redefinition, Editions (neu in 11g Release 2) und Rolling Upgrade fähige Patches ermöglichen Ausfallzeiten zu vermeiden. Im Fall von Betriebssystem oder Hardware Upgrades kann mit Oracle Data Guard eine Ausfallzeit auf wenige Minuten reduziert werden.

Bei Release-Wechseln der Oracle-Software (Patchsets, Major Release Upgrades) kann ebenfalls Oracle Data Guard genutzt werden, um mit einem Rolling Upgrade mit der Logical Standby

Datenbank das für den Upgrade notwendige Wartungsfenster auf wenige Minuten zu reduzieren. Im Fall von ungeplanten Ausfallzeiten können je nach Szenario verschiedene Techniken – auch in Kombination – zum Einsatz kommen. Zur Absicherung gegen Ausfallzeiten beim Ausfall eines Servers kann der Oracle Real Application Cluster eingesetzt werden. Treten Fehler an den Datenfiles auf, können diese entweder im Rahmen des Backup/Recovery Konzeptes mit RMAN und/oder Oracle Secure Backup wiederhergestellt werden oder von einer Oracle Data Guard Physical Standby Datenbank bereitgestellt werden. Ausfällen im Storage-Bereich kann mit sogenannten Failure Groups im Automatic Storage Management entgegengewirkt werden. Benutzerfehler, also logische Korruptionen, können mit den unterschiedlichen Flashback-Technologien oder auch Oracle Data Guard adressiert werden.

Zur Vorbeugung von Ausfallzeiten, die durch den Total-Ausfall eines Rechenzentrums verursacht werden, kann mit Oracle Data Guard in einem Disaster-Rechenzentrum eine Standby-Datenbank betrieben werden, die im Fehlerfall sofort die Produktionsaufgaben übernehmen kann. Dieses Umschalten kann seit Oracle 10g mit dem Fast Start Failover auch automatisiert werden. Beginnend mit Oracle 11g Release 1 können hierfür verschiedene Kriterien definiert werden.

Die verschiedenen Hochverfügbarkeits-Funktionen können miteinander kombiniert werden, um eine sogenannte "Maximum Availability Architektur" aufzubauen. Im Oracle Technet gibt es zahlreiche technische Whitepaper und Best Practices Artikel zum Aufbau und Betrieb einer Maximum Availability Architektur. Abbildung 1 zeigt die Komponenten einer Maximum Availability Architecture. Oracle Data Guard und der Recovery Manager sind elementare Bestandteile dieser Hochverfügbarkeitsarchitektur.

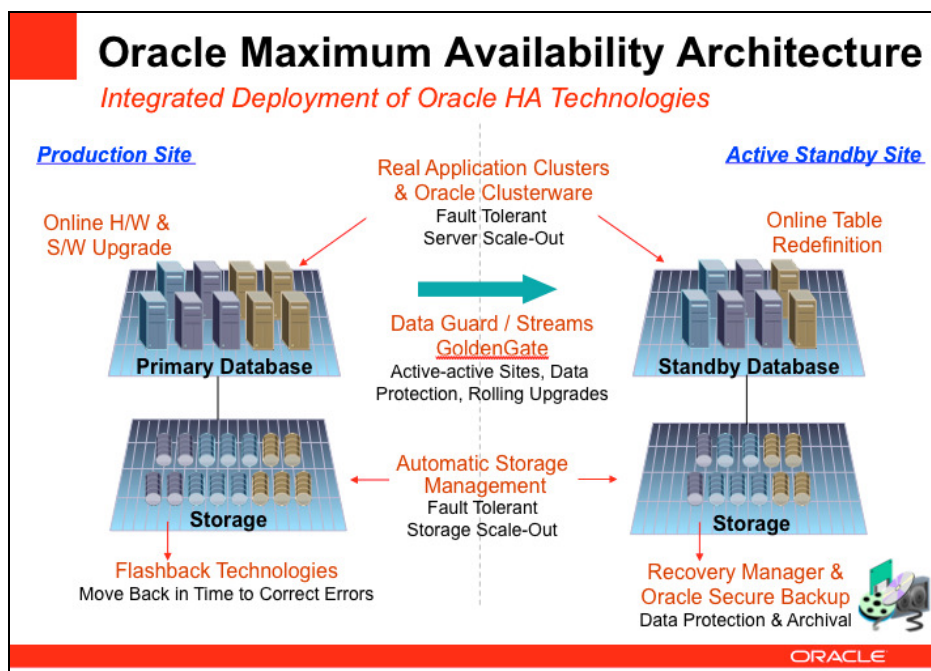


Abb. 1: Maximum Availability Architecture

2. Oracle Data Guard

Oracle Data Guard bietet eine Software Infrastruktur zur Erstellung und Verwaltung einer oder mehrerer Standby Datenbanken als transaktionskonsistente Kopien der Produktions-Datenbank. Data Guard dient dem Schutz der unternehmenskritischen Daten vor Fehlern, Ausfällen, Desastern und Korruptionen.

Sollte die Produktions-Datenbank aus irgendwelchen planmäßigen oder außerplanmäßigen Gründen ausfallen, kann eine der angeschlossenen Standby Datenbanken direkt die Produktionsaufgaben übernehmen. Somit werden Ausfallzeiten minimiert und Datenverlust verhindert.

Data Guard ist Bestandteil der Oracle Enterprise Edition. Beginnend mit Oracle 11g kann die Physical Standby Datenbank bei gleichzeitigem Apply lesend für Abfragen geöffnet werden. Dies erfordert die Lizenzierung der Active Data Guard Option. Data Guard kann mit anderen Oracle Hochverfügbarkeits-Funktionalitäten wie Real Application Clusters (RAC), Flashback Database und Recovery Manager (RMAN) kombiniert werden.

2.1 Data Guard Architektur

Eine Data Guard Umgebung besteht immer aus einer Produktions-Datenbank, auch Primär-Datenbank genannt, und bis zu neun daran angeschlossenen Standby Datenbanken (ab Oracle 11.2 bis zu 30 Standby Datenbanken). Die Standby Datenbanken werden anfangs aus einem Backup der Primär-Datenbank erstellt. Bei Verwendung des Recovery-Manager (RMAN) musste bis Oracle 10g hierzu ein Backup der Primär-Datenbank in einem Staging-Bereich bereitgestellt werden. Aus diesem Backup heraus wurde dann die Standby Datenbank aufgebaut. Beginnend mit Oracle 11g kann das Aufbauen der Standby Datenbank mithilfe von RMAN-Befehlen direkt aus der laufenden Produktions-Datenbank heraus erfolgen, ohne das Backup in einem Staging-Bereich ablegen zu müssen. Dies bedeutet eine erhebliche Zeit- und Speicherplatz-Ersparnis. Das Primär System kann eine Single-Instance oder RAC Datenbank sein. Unabhängig davon kann die Standby Datenbank als Single-Instance oder als RAC betrieben werden.

Änderungen, die auf dem Primär-System stattfinden, werden durch die Log-Transport Services auf die Standby Datenbank übertragen. Beginnend mit Oracle 11g erfolgt der Transport standardmäßig mit dem LNS (Log Network Server), wahlweise synchron oder asynchron. Der Archiver wird nur noch zum Auflösen von Gaps (fehlende archivierte Redo Logs) verwendet.

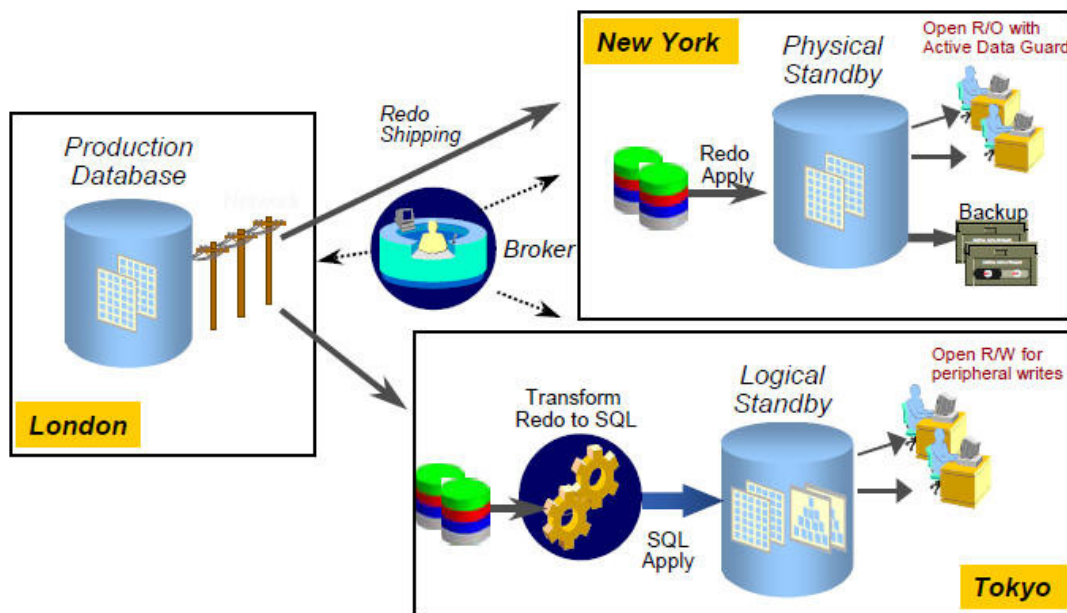


Abb. 2: Oracle Data Guard Architektur

Zur Erleichterung der Administration einer Data Guard Umgebung kann die Broker-Architektur eingesetzt werden. Ein zusätzlicher Prozess, der Data Guard Broker (DMON), überwacht regelmäßig

das Gesamt-System. Der Data Guard Broker kann über zwei APIs angesprochen werden. Das sind das Command-Line-Interface `dgmgrl` und die graphische Oberfläche, die in Enterprise Manager Grid Control integriert ist.

Transaktionen, die in der Primärdatenbank stattfinden, werden im Current Online Redo Log protokolliert. Dieses Transaktions-Protokoll wird dann kontinuierlich synchron oder asynchron nach jedem Commit via SQL*Net auf die Standby Seite übertragen.

Dort nimmt der RFS (Remote File Server-Prozess) die Informationen entgegen und stellt sie für das anschließende Apply, also das Einspielen in die Standby Datenbank, zur Verfügung.

Das Apply wiederum kann kontinuierlich direkt nach Eintreffen der Informationen (Real Time Apply), nach Auslösen eines Log-Switches auf dem Primär-System oder aber gezielt zeitlich verzögert (Delayed Apply) erfolgen.

Bei einer kontinuierlichen synchronen Übertragung können sogenannte Zero Data Loss-Umgebungen aufgebaut werden, in denen garantiert wird, dass beim Totalausfall der Primär-Datenbank keine abgeschlossenen Transaktionen verloren gehen.

Die jeweilige Konfiguration hängt von den Anforderungen an das System ab.

Möchte man die Standby Datenbank dazu einsetzen, Systemausfälle durch menschliche Fehler (versehentliches Löschen von Daten) zu vermeiden, so kann man das Apply optional durch ein sogenanntes Delay verzögern. Seit Oracle 10g Release 1 kann alternativ die Standby Datenbank mit der Flashback Database Technologie auf einen Zeitpunkt in der Vergangenheit zurückgesetzt werden.

Oracle unterscheidet zwei Arten von Standby Datenbank, die sich im wesentlichen in der Art des Einspielens der Änderungen auf der Standby Datenbank, also im Apply, unterscheiden.

2.2 Physical Standby Datenbank (Redo Apply)

Bei der Physical Standby Datenbank ist der MRP (Managed Recovery Process) für das Apply verantwortlich. D.h., die Änderungen, die auf der Primär-Datenbank stattgefunden haben, werden durch ein Recovery in der Physical Standby Datenbank nachgezogen. Daher ist die Physical Standby Datenbank blockweise 1:1 identisch mit der Primär-Datenbank und erlaubt zum Einen optimalen Disaster-Schutz, zum Anderen kann die Physical Standby Datenbank für Backups herangezogen werden und somit das Primärsystem entlasten. Die Physical Standby Datenbank bietet somit optimalen Disaster-Schutz.

Ein besonderer Betriebsmodus der Physical Standby Datenbank ist Active Data Guard. Active Data Guard erlaubt den gleichzeitigen lesenden Zugriff auf die Physical Standby Datenbank während das Apply stattfindet. Active Data Guard ist eine separate Option zur Oracle Enterprise Edition. Active Data Guard erlaubt den Betrieb von sogenannten Lesefarmen z.B. bei Web-Anwendungen, bei gleichzeitigen Desasterschutz. Mit Hilfe von Active Data Guard können ab Oracle 11.2 korrupte Blöcke der Primär-Datenbank automatisch im laufendem Betrieb durch intakte Blöcke der Standby Datenbank ersetzt und so die Verfügbarkeit der Primär Datenbank vergrößert werden.

2.3 Logical Standby Datenbank (SQL Apply)

Der grundlegende Unterschied zwischen der Physical und der Logical Standby Datenbank liegt im Apply Mechanismus. Während bei der Physical Standby Datenbank das Apply in einer blockweise identischen Datenbank durch ein Recovery stattfindet, findet bei der Logical Standby Datenbank ein SQL Apply statt. Die Logical Standby Datenbank ist somit nicht mehr blockweise 1:1 identisch. Sie steht während des Apply den Anwendern lesend und unter bestimmten Voraussetzungen auch schreibend zur Verfügung. Für das Apply werden mit der LogMiner-Technologie die Inhalte der Redo Logs analysiert und aus diesen Informationen SQL-Statements rekonstruiert. Diese SQL-Statements werden dann in der offenen Datenbank angewandt.

Dadurch kann die Logical Standby Datenbank während des Apply für Reportingaufgaben genutzt werden. Ferner erlaubt das PL/SQL-Paket DBMS_LOGSTDBY eine Manipulation des Apply-Mechanismus (z.B. Definition von Filtern).

Auch wenn zunehmend mehr Datentypen unterstützt werden, gibt es Beschränkungen. Daher kann die Logical Standby Datenbank nur unter bestimmten Voraussetzungen auch als Disaster-Schutz eingesetzt werden. Die Einschränkungen bei den unterstützten Datentypen entsprechen denen des Oracle LogMiners und sind in der Data Guard Dokumentation im Appendix C beschrieben.

3. Oracle Recovery Manager (RMAN)

Oracle Recovery Manager (RMAN) ist DAS Backup/Recovery Werkzeug der Oracle Datenbank. RMAN wurde mit Oracle 8.0 eingeführt und seitdem kontinuierlich erweitert. RMAN kann über zwei APIs bedient werden, dem Command-Line-Interface `rman` und über die webbasierte graphische Oberfläche des Enterprise Managers. RMAN kann in allen möglichen Oracle Datenbank Architekturen eingesetzt werden, egal ob Single Instance oder Real Application Clusters Umgebungen, egal, ob die Datenfiles im Filesystem, in raw devices oder in ASM (Automatic Storage Management) liegen. Mit RMAN können Online und Offline Backups durchgeführt werden. Es können sogenannte "Savesets" oder "image copies" der Datenbank erstellt werden. Ferner besteht die Möglichkeit Komplet-Backups, Teil-Backups oder inkrementelle Datenbank Backups zu erstellen. Ein Recovery kann vollständig oder teilweise erfolgen, auf Datenfile-Basis oder Blocklevel basiert.

Der RMAN führt automatisch optimierte Backups durch, bei denen nicht oder nicht mehr verwendete Datenblöcke nicht mitgesichert werden (Null block compression, unused block compression). Ferner wird das Backup der Undo Tablespaces dahingehend optimiert, dass UNDO-Blöcke, die Informationen über committete Transaktionen enthalten, die für ein Recovery nicht mehr benötigt werden, ebenfalls aus dem Backup ausgeschlossen werden. Dadurch ergeben sich Platz und Zeitersparnisse beim Durchführen eines Backups und/oder Recovery. Zur Beschleunigung des Backups kann der RMAN Backups über mehrere sogenannte Channel parallelisieren.

Die Informationen über die durchgeführten Datenbank Backups werden immer im Controlfile der Datenbank protokolliert. Zusätzlich kann ein sogenannter Recovery Catalog, eine separate Datenbank zur Verwaltung aller RMAN Backups aller Datenbanken, eingerichtet werden. In diesem können dann auch RMAN Skripte hinterlegt werden, die die Durchführung von Datenbank Backup/Recovery Maßnahmen erleichtern können.

Beginnend mit Oracle 11g können innerhalb des Catalogs mit sogenannten "virtual catalogs" Untergruppen von Datenbanken abgebildet werden, die jeweils von eigenen Administratoren gesichert werden.

Die von RMAN erstellten Backups können entweder diskbasiert im Filesystem des Servers, in der Fast Recovery Area (FRA) im Filesystem oder in ASM, oder direkt auf Tape erfolgen. Für eine direkte Sicherung auf Band werden entweder third party Plugins verwendet oder Oracle Secure Backup. RMAN Backup können komprimiert und/oder verschlüsselt erstellt werden.

In Verbindung mit dem Enterprise Manager kann auch ein automatischer Datenbank Backup Job definiert werden, der dafür sorgt, dass die Datenbank automatisch einmal täglich in die Fast Recovery Area gesichert wird.

Die Verwendung des RMAN bei der Sicherung von Oracle Datenbanken erleichtert die Backup-Prozeduren und stellt sicher, dass alle für ein Backup notwendigen Aktionen durchgeführt werden und dass das Backup vollständig ist (also alle Datenbank-Dateien der Datenbank enthalten sind, egal wo diese Dateien abgelegt sind).

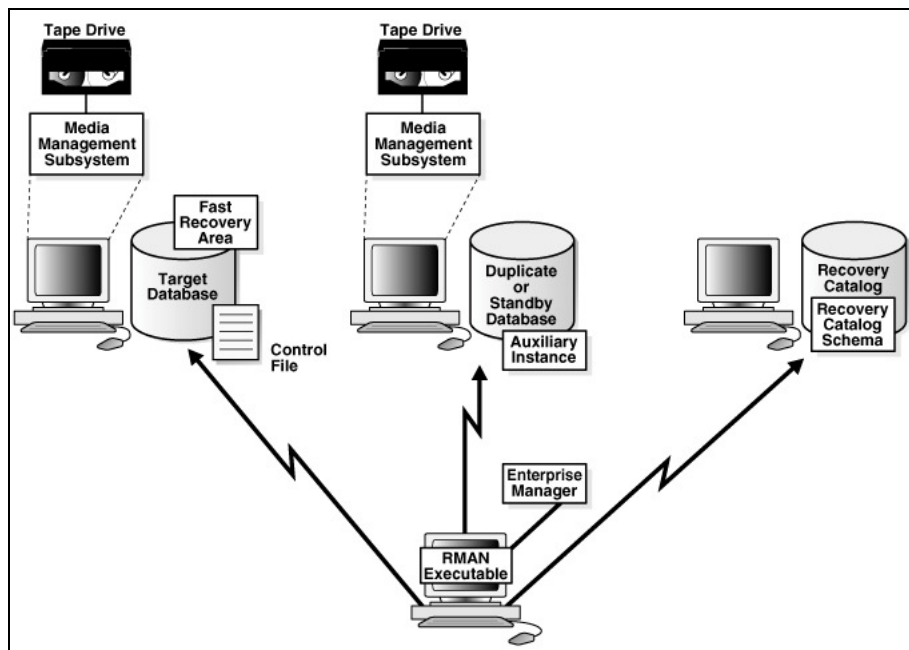


Abb. 3: Oracle Recovery Manager Architektur Beispiel

Abbildung 3 zeigt schematisch eine Recovery Manager Architektur. Die zu sichernde Datenbank wird "Target Database" genannt. Eine Hilfsdatenbank oder "Auxiliary Database" kommt beim Klonen der Target Datenbank oder bei der Erstellung einer Standby Datenbank zum Einsatz. Der "Recovery Catalog" ist in einer separaten Oracle Datenbank abgelegt und enthält alle Informationen über durchgeführte Datenbank Backups. In der Fast Recovery Area können die Archivelogs der Datenbank und diskbasierte Backups abgelegt werden. Bei Verwendung von Media Managern kann eine direkte Sicherung auf Band erfolgen. Die Bedienung erfolgt entweder über das rman CLI oder den Enterprise Manager.

In Zusammenhang mit einer Data Guard Umgebung wird mit einer Recovery Catalog Datenbank gearbeitet.

Grundsätzliche RMAN Einstellungen können mit dem `configure` Kommando eingestellt werden und gelten dann als default für alle zukünftigen Backups. Über das `configure` Kommando können auch Aufbewahrungsregeln für in der Fast Recovery Area abgelegte Backups und Archivelogs definiert werden. Dies kommt im Zusammenhang mit Data Guard zum Tragen.

4. Fast Recovery Area (FRA)

Die Fast Recovery Area, die bis Oracle 10g als Flash Recovery Area bezeichnet wurde, ist ein von der Oracle Datenbank verwaltetes Verzeichnis, das entweder im Filesystem oder in einer ASM disk group liegt. Die FRA stellt eine zentrale Stelle für alle Backup/Recovery relevanten Dateien dar. In der FRA werden bei Verwendung der Flashback Database Funktionalität die Flashback Logs gespeichert. Ferner können die Archivelogs und Backups oder Image Copies der Datenbank in diesem Bereich abgelegt werden. Die FRA wird durch folgende Datenbank Parameter definiert:

<code>DB_RECOVERY_FILE_DEST</code>	ASM Diskgroup oder Verzeichnis
<code>DB_RECOVERY_FILE_DEST_SIZE</code>	Platz, den die Datenbank für die FRA belegen darf

5. RMAN im Einsatz in Data Guard Umgebungen

RMAN kann in Data Guard Umgebungen in mehreren Situationen zum Einsatz kommen und Data Guard "aware" konfiguriert werden.

Zur Erstellung einer Standby Datenbank wird immer ein Backup der Primär Datenbank benötigt. Dieses kann als User Managed Backup oder aber sehr elegant als RMAN Backup bereitgestellt werden.

Der Einsatz einer Standby Datenbank entbindet den DBA nicht davon ein solides Backup/Recovery Konzept für das Produktionssystem zu erstellen. Bei Verwendung einer Physical Standby Datenbank können Backup Tätigkeiten auch auf die Standby Seite verlagert werden. Der RMAN unterstützt dies.

Bei Verwendung der FRA für ArchiveLogs und Backups ist die Datenbank in der Lage, im Fall einer sogenannten "space pressure" in der Fast Recovery Area bei korrekter Konfiguration der RMAN Aufbewahrungsregeln automatisch auf der Standby Seite nicht mehr benötigte ArchiveLogs zu löschen und so den Platz in der FRA wieder freizugeben.

In Verbindung mit Active Data Guard kann RMAN im Fall eines korrupten Datenblockes in der Primary für ein Recovery automatisch einen intakten Datenblock von der Standby Datenbank anfordern, anstelle auf die Backups zugreifen zu müssen.

Ebenfalls in Verbindung mit Active Data Guard kann RMAN auf der Standby Datenbank auch schnelle inkrementelle Backups mit "Block Change Tracking" durchführen.

5.1 Anlegen einer Standby Datenbank

Beim erstmaligen Anlegen einer Standby Datenbank kann der RMAN dank des `duplicate` Kommandos schon seit Oracle 9i eingesetzt werden. Bis Oracle 10g war es notwendig, dies in zwei Schritten durchzuführen. Zunächst wurde ein Backup der Produktionsdatenbank, ein spezielles Backup des Controlfiles und ein Backup der archivierten Redo Logs erstellt. Anschließend wurde die Standby Instanz mit einem auf diese Instanz zugeschnittenen `spfile` in den `nomount` Status hochgefahren. In einem zweiten Schritt wurde nun aus den Daten des Backups eine Standby Datenbank aufgebaut.

Die folgenden Listings zeigen beispielhaft die RMAN Befehlssequenzen für Oracle 9i und 10g. Wichtig ist, dass das Backup für den Aufbau der Standby Datenbank auf der Standby Seite in genau demselben Verzeichnis zur Verfügung steht, in das es auf der Produktionsseite gesichert wurde.

Backup erstellen auf der Produktionsseite:

```
rman target / nocatalog
```

```
RMAN> run
{
allocate channel c1 type disk;
backup database format '/tmp/oracle/backup/db_%U.sav filesperset 1' ;
backup current controlfile for standby format '/tmp/oracle/backup/ctl.sav' ;
sql "alter system archive log current ";
backup archivelog all format '/tmp/oracle/backup/arch.sav' ;
}
```

Die Standby Instanz mit bereitgestellten `spfile`, Password File, konfigurierten SQL*Net und gestartetem Listener in den `nomount`-Status bringen und anschließend Standby Datenbank aufbauen. Es werden zwei Connects benötigt: ein Connect zur Produktionsdatenbank (`target`) und ein Connect zur Standby Instanz (`auxiliary`). Das Kommando kann auf der Produktionsseite aufgerufen werden, daher reicht für `target /`.

```
rman target / auxiliary sys/<SYS_PW>@<standby_instance>
```

```
RMAN> run
{
allocate auxiliary channel c1 type disk;
duplicate target database for standby dorecover nofilenamecheck;
}
```

Beginnend mit Oracle Version 11g kann man jetzt eine Standby Datenbank direkt aus der laufenden Produktionsdatenbank heraus aufbauen, ohne ein Backup auf der Platte ablegen zu müssen. Dies spart Plattenplatz und Zeit beim Aufbau der Standby Datenbank. Die notwendigen Anpassungen am SPFILE können ebenfalls über das RMAN Kommando zum Anlegen der Standby Datenbank definiert werden. Somit reicht ab Oracle 11g das Bereitstellen einer mit `nomount` hochgefahren Standby Instanz, die SQL*Net seitig erreichbar ist und deren `spfile` nur den Parameter `dbname` enthält. Das Password File muss zuvor auf die Standbyseite **kopiert** werden.

Das folgende Listing zeigt die ab Oracle 11g zur Verfügung stehenden Möglichkeiten mit RMAN:

```
rman target / auxiliary sys/<SYS_PW>@<standby_instance>
```

```
run {
  DUPLICATE TARGET DATABASE
  FOR STANDBY
  FROM ACTIVE DATABASE
  DORECOVER
  SPFILE PARAMETER_VALUE_CONVERT '/u01/', '/stby/'
  SET "DB_UNIQUE_NAME"="STBY"
  SET SGA_TARGET="200M"
  SET LOG_FILE_NAME_CONVERT='/u01', '/stby'
  SET DB_FILE_NAME_CONVERT = '/u01', '/stby'
  NOFILENAMECHECK;
```

Neue Kommandos im Oracle 11g RMAN sind die Klauseln "from active database" und "spfile parameter_value_convert". Ersteres erlaubt das direkte Anlegen der Standby Datenbank über SQL*Net ohne temporäre Zwischenablage eines Backup im Filesystem und letzteres ermöglicht spile Parameter Anpassungen ohne vorherige Erstellung und Editieren einer Parameterdatei.

5.2 Datenbanksicherung mit RMAN

In Data Guard Umgebungen muss bei der Nutzung des RMAN über das initiale Erstellen der Standby Datenbank hinaus, immer ein Recovery Catalog verwendet werden. Da die Primär und Standby Datenbank eindeutig identifizierbar sein müssen, muss der Parameter `DB_UNIQUE_NAME` entsprechend eindeutig auf allen Seiten gesetzt sein. Befinden sich die Datenbanken auf separaten Systemen, so wird ein Backup auf Platte immer als lokal betrachtet und steht zunächst auch nur auf dem jeweiligen Rechner für ein Restore zur Verfügung. Bei Sicherungen auf Band steht die Sicherung automatisch allen Seiten zur Verfügung. In Verbindung mit der Physical Standby Datenbank können Sicherungen der Primär Datenbank für ein Restore auf der Standby Datenbank verwendet werden und umgekehrt. Daher ist es möglich in Data Guard Umgebungen die Backup Aktivitäten auf die Standbyseite zu verlagern. Für den Recovery Fall muss bei einer Sicherung ins Filesystem dann aber berücksichtigt werden, dass bei Verlust einer Datendatei auf dem Primär-System, das Backup davon zunächst aus dem Filesystem der Standby Datenbank auf das Primärsystem übertragen und registriert werden muss (z.B. mit ftp), bevor die Datei dort restored und recoverd werden kann. In

höchstverfügbaren Systemen mit kleinen RTO (recovery time objectives) wird man daher auf beiden Seiten RMAN Sicherungen durchführen.

Zur leichteren Handhabung des RMAN können eine Reihe von Voreinstellungen für den Betrieb des RMAN definiert werden (Standardsicherungsverzeichnis, Aufbewahrungsregeln für Backups, optimiertes Backup, automatisches Controlfile Backup, ...).

Mit Hilfe des `CONFIGURE` Kommandos können diese grundsätzliche Einstellungen für die jeweilige Datenbank oder für beide Datenbanken vorgenommen werden.

Normalerweise legt man mit `CONFIGURE` die Einstellungen für die Datenbank fest, an die man sich mit `TARGET` mit dem RMAN verbunden hat.

Alternativ kann man mit `CONFIGURE ... FOR DB_UNIQUE_NAME` Eigenschaften für eine bestimmte Datenbank definieren, auch wenn man nicht mit dieser verbunden ist.

Entsprechend listet

```
RMAN> SHOW DEVICE TYPE FOR DB_UNIQUE_NAME dgprod3;
```

die Eigenschaften der Datenbank mit dem `DB_UNIQUE_NAME dgprod3` auf.

Mit

```
RMAN> SHOW ALL FOR DB_UNIQUE_NAME dgprod3;
```

werden entsprechend alle eingestellten Eigenschaften der Datenbank mit dem `DB_UNIQUE_NAME dgprod3` aufgelistet.

Mit

```
RMAN> SHOW ALL FOR DB_UNIQUE_NAME ALL;
```

werden entsprechend die eingestellten Eigenschaften aller zur Data Guard Konfiguration gehörenden Datenbanken aufgelistet.

Die Durchführung eines Backups der Standby Datenbank unterscheidet sich nicht von der Durchführung eines Backups auf der Primär Datenbank. Das RMAN Backup kann auch durchgeführt werden, wenn die Standby Datenbank im Managed Recovery Mode ist:

```
RMAN> BACKUP DATABASE;
```

```
RMAN> BACKUP ARCHIVELOG ALL;
```

führt ein vollständiges Backup der Standby Datenbank durch.

Werden nur auf einer Seite Datenbank Backups durchgeführt und sollen die archivierten Redo Logs automatisch unter bestimmten Voraussetzungen gelöscht werden, so kann man durch Einstellen der "archived redo log deletion policy" entsprechende Einstellungen vornehmen.

5.3 Automatisches Löschen von archivierten Redo Logs einrichten

RMAN kann so eingerichtet werden, dass unter bestimmten Voraussetzungen nicht mehr benötigte archivierte Redo Logs automatisch gelöscht werden.

Folgende Voraussetzungen müssen **alle** erfüllt sein:

1) Vor Oracle 11g muss entweder die Standby Datenbank im `log_archive_dest_n` Parameter als `MANDATORY` markiert sein (was grundsätzlich nicht empfohlen wird) oder für alle beteiligten Datenbanken muss der folgende Parameter gesetzt sein:

```
_log_deletion_policy="ALL"
```

Beginnend ab Oracle 11g gilt diese Einschränkung nicht mehr.

2) Die Fast Recovery Area muss konfiguriert und nativ verwendet werden:

```
SQL> alter system set DB_RECOVERY_FILE_DEST='/u01/fra';
SQL> alter system set DB_RECOVERY_FILE_DEST_SIZE=80G;
SQL> alter system set LOG_ARCHIVE_DEST_1='LOCATION=USE_DB_RECOVERY_FILE_DEST';
```

3) Für die Primär und Standby Datenbank muss eine Deletion Policy definiert werden:

```
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON STANDBY;
```

4) das archivierte Redo Log muss auf der Standby Datenbank eingespielt worden sein (applied)

5) das archivierte Redo Log muss gemäß Retention Policy obsolet sein

6) es muss eine sogenannte "Space Pressure" in der Fast Recovery Area vorliegen.

Besonderes Augenmerk liegt in dieser Liste auf Punkt 5. Eine Retention Policy kann nur dann erfüllt werden, wenn auch tatsächlich Backups **stattgefunden** haben. d.h. wenn ein archiviertes Redo Log zum Löschen freigegeben werden kann, wenn dieses archivierte Redo Log zweimal gesichert wurde, dann müssen auch zwei Sicherungen vorliegen, bevor das archivierte Redo Log automatisch gelöscht werden kann.

Ferner findet ein automatisches Löschen nur dann statt, wenn auch tatsächlich Space Pressure (weniger als 15% verbleibender Platz) in der FRA besteht. Solange die FRA über genügend Platz verfügt, erfolgt kein automatisches Löschen.

Wieviel Platz automatisch freigegeben werden kann (SPACE_RECLAIMABLE), man mit folgender Abfrage ermitteln:

```
SQL> SELECT * FROM V$RECOVERY_FILE_DEST;
```

NAME

```
-----
SPACE_LIMIT SPACE_USED SPACE_RECLAIMABLE NUMBER_OF_FILES
-----
/opt/oracle/product/1010/dbs/ora_flash_area
1073741824 913181184 2124288 11
```

Die Spalte SPACE_RECLAIMABLE wird erst ab einem Füllgrad von 85% der FRA gefüllt.

Die nicht mehr benötigten und für ein Löschen zur Verfügung stehenden Dateien kann man mit folgender Abfrage ermitteln:

```
RMAN> report obsolete;
```

```
RMAN retention policy will be applied to the command
```

```
RMAN retention policy is set to redundancy 3
```

```
Report of obsolete backups and copies
```

Type	Key	Completion Time	Filename/Handle
Archive Log	51	15-JUN-05	/opt/oracle/product/1010/dbs/ora_flash_area/DB10G/archivelog/2005_06_15/o1_mf_1_110_1c0mj1b2_.arc
Archive Log	52	15-JUN-05	/opt/oracle/product/1010/dbs/ora_flash_area/DB10G/archivelog/2005_06_15/o1_mf_1_111_1c0mj8co_.arc
Archive Log	53	15-JUN-05	/opt/oracle/product/1010/dbs/ora_flash_area/DB10G/archivelog/2005_06_15/o1_mf_1_112_1c0mj9tp_.arc
Archive Log	54	15-JUN-05	/opt/oracle/product/1010/dbs/ora_flash_area/DB10G/archivelog/2005_06_15/o1_mf_1_113_1c0mjh6j_.arc
Archive Log	55	15-JUN-05	/opt/oracle/product/1010/dbs/ora_flash_area/DB10G/archivelog/2005_06_15/o1_mf_1_114_1c0mjn90_.arc

5.4 Automatisches Reparieren von korrupten Datenblöcken mit RMAN und Active Data Guard

Eine der Erweiterungen von Data Guard in Oracle 11g R2 ist, dass in Verbindung mit Active Data Guard, die Primär Datenbank im Fall einer Blockkorruption automatisch einen intakten Block von der Standby Datenbank anfordern kann. Folgende Voraussetzung müssen dafür erfüllt sein:

- eine Physical Standby Datenbank muss im Realtime Query Mode (Active Data Guard) betrieben werden
- `log_archive_config` muss die beteiligten Datenbanken unter `DG_CONFIG` aufgelistet haben
- `log_archive_config` muss einen Hinweis auf `db_unique_name` enthalten

Entdeckt die Primär Datenbank bei der Abarbeitung einer Abfrage einen korrupten Block, so versucht sie unter den gegebenen Voraussetzungen zunächst einen intakten Block von der Standby Datenbank anzufordern. Daher kommt es in diese Situationen nur zu einer kurzen Verzögerung und nicht zum ORA-1578 Fehler.

Wenn ein korrupter Block identifiziert wurde, kann man mit RMAN mit einem "block media recovery" mit dem Kommando `RECOVER ... BLOCK` den defekten Block reparieren. Beginnend ab Oracle 11g R2 geht RMAN folgendermaßen vor, um den defekten Block zu reparieren: zuerst wird in der Realtime Query Standby Datenbank (Active Data Guard) nach dem Block gesucht, danach in den Flashback Logs und erst danach im letzten Full- oder Level 0 inkrementellen Backup.

5.5 Block Change Tracking mit RMAN und Active Data Guard

Neben den Vollbackups und Imagekopien der Datenbank, erlaubt RMAN auch das Erstellen von inkrementellen Backups. Auf Basis eines Level 0 Vollbackups werden beim nächsten Level 1 Backup nur die geänderten Datenblöcke seit der letzten Vollsicherung (kumulativ) oder seit der letzten Level 1 Sicherung (differenziell) gesichert. Damit für eine inkrementelle Sicherung nicht immer alle Datenbankblöcke durchgescannt werden müssen, kann man optional das sogenannte "Block Change Tracking" einschalten. Es wird dann im mit `DB_CREATE_FILE_DEST` definierten Verzeichnis eine Datei gepflegt, in der die geänderten Blöcke aufgelistet werden.

Das "Block Change Tracking" wird über ein `ALTER DATABASE` Kommando eingeschaltet.

```
ALTER DATABASE ENABLE BLOCK CHANGE TRACKING;
```

Die Datei wird dabei in dem durch `DB_CREATE_FILE_DEST` definierten Verzeichnis angelegt. Alternativ kann man den Dateinamen auch explizit vorgeben:

```
ALTER DATABASE ENABLE BLOCK CHANGE TRACKING  
USING FILE '/mydir/rman_change_track.f' REUSE;
```

Beginnend mit Oracle 11g und in Verbindung mit Active Data Guard, kann nun auch auf der Physical Standby Datenbank das "Block Change Tracking" aktiviert werden, und somit die Backup Zeit für ein inkrementelles Backup der Standby Seite verringert werden.

6. Zusammenfassung

Data Guard und RMAN wurden vor dem Hintergrund einer Oracle Datenbank Architektur designed. Beide zusammen angewandt bieten die Möglichkeit kritische Datenbanksysteme zuverlässig und integriert administrieren zu können. Die Kombination beider Features erlaubt es unterschiedliche Datenbankverfügbarkeits-Anforderungen bedienen zu können. Data Guard und RMAN sind Features der Enterprise Edition. RMAN steht auch in der Standard Edition zur Verfügung. Die im Papier

ebenfalls beschriebene Funktion Active Data Guard ist eine kostenpflichtige Option zur Oracle Enterprise Edition.

Relevante Support Notes:

1116484.1 Master Note for Oracle Recovery Manager

728053.1 Configure RMAN to purge archivelogs after applied on standby

464668.1 Maintenance Of Archivelogs on Standby Databases

848716.1 RMAN Effectively in A Dataguard Environment

315098.1 How is the space pressure managed in the Flash recovery Area - An Example

Kontaktadresse:

Claudia Hüffer

Oracle Deutschland B.V. & Co. KG

Kühnehöfe 5

D-22761 Hamburg

Telefon: +49(0) 40 89091-135
Fax: +49(0) 40 89091-250
E-Mail: claudia.hueffer@oracle.com
Internet: <http://www.oracle.com>