

Storage Advantages by Oracle – Warum Redundancy Normal in ASM?

**Ralf Moosandl
Trivadis AG
Zürich**

Schlüsselworte:

ASM Automatic Storage Management Redundanz Redundancy Normal Shared Pool Usage File System Monitoring Usage Split Mirror Clone Disk Group File Template Allocation Unit AU

Einleitung

Seit dem Rollout des Automatic Storage Managements unter Oracle 10g und einer Anlaufzeit zum Sammeln von Erfahrungen, sind nun viele Anwender dabei, Datenbankfiles im ASM zu speichern, da inzwischen zahlreiche Gründe für dessen Einsatz sprechen. Bei der irreversiblen Wahl der Redundanz für das elementare Element im ASM, der Diskgruppe, wird oftmals die Redundanz External gewählt und die Verantwortung fürs Mirroring auf das Storage weitergereicht. Durch den Einsatz von Diskgruppen mit den Redundanzen Normal gewinnt man jedoch enorm an Flexibilität im Hinblick aufs Cloning und die File Space Usage.

Im Folgenden wird eine Split Mirror Technologie, basierend auf normale Redundanz in der Diskgruppe, vorgestellt. Somit hat man eine Möglichkeit für schnelles Provisioning von Datenbanken, Tablespace und RDBMS-Binaries. Besonderheiten beim Monitoring des freien Platzes in Diskgruppe und die Abschätzungen des Ressourcenverbrauchs anhand der Shared Pool Usage werden ebenfalls behandelt.

Storage Advantages by Oracle - Warum Redundancy Normal in ASM?

Was spricht für den Einsatz von ASM?

Beim Einsatz der RAC Standard Edition ist man aus Lizenzgründen an den Einsatz vom Automatic Storage Management gebunden. Mehrkosten entstehen durch den Einsatz von ASM nicht. Da man alle Datenbankfiles im ASM speichern kann, erübrigt sich der Einsatz von Cluster Filesystemen. Somit können Kosten eingespart werden.

Ein weiterer grosser Vorteil dieser Storage Option ist, dass man den Support sowohl für das RDBMS als auch für die Speicheroption ASM direkt vom Hersteller bezieht und somit wertvolle Zeit bei der Lösung von Problemen einspart. Das Zuweisen von Schuld zwischen unterschiedlichen Herstellern entfällt also. Natürlich existieren neben besagten Gründen auch technische Fürsprecher.

Technische Gründe für den Gebrauch von ASM

Durch den unternehmensweiten Einsatz reduziert sich die administrative Komplexität. Ausserdem findet die Speicherung der Datenbankfiles in optimierter Form statt, zumal beim Anlegen von Datenbankfiles optimal für den jeweiligen Filetyp gestriped wird. Somit findet das I/O Tuning frühestmöglich und für alle Einsatzgebiete der Datenbank statt.

Aus DBA Sicht wird die Anzahl der zu administrierenden Komponenten reduziert, da ja eine Diskgruppe viele Filesysteme ersetzen kann.

Das automatisch implementierte Feature „Oracle Managed Files“ erspart den Datenbankadministratoren viele unnötige Diskussionen über das optimale Filesystem für die einzelnen Filetypen und für die optimale Verzeichnisstruktur, da ja die Ordnerstruktur automatisch vorgegeben ist, aber trotzdem über die Verwendung von Aliasen werden kann.

Das Anlegen und Löschen von dem Filetype entsprechenden Ordnern erfolgt automatisch.

Ausserdem reduziert man mit der Möglichkeit der Onlinemigration auf neue Storage Hardware die Downtime. Nicht zu unterschätzen ist auch der Vorteil eines nicht vorhandenen Filesystem Interfaces, da weder Virens Scanner noch unachtsame Administratoren Files für den Zugriff durch Oracle Prozesse blockieren oder auch löschen können. In Zeiten knapper Budgets kann man die Auslastung der Ressourcen durch Konsolidierung erhöhen und von Mechanismen wie etwa File Access Control profitieren.

Wenn man in konsolidierten Umgebungen Filesystem Funktionalitäten braucht – etwa zur Ablage von klassischen Exportdumps oder Logfiles - kann man auf den in 11.2 integrierten Automatic Dynamic Volume Manager zurückgreifen.

Ausserdem kann man Block Devices so leicht administrieren wie Fileserver. Das Formatieren und die Einbindung einzelner Block Devices in den Bootprozess entfällt.

Performance, Skalierbarkeit und Zuverlässigkeit werden erhöht und Raw Disk I/O Performance erzielt. Die Storage Optionen hinsichtlich Striping, auch über Storage Arrays hinweg, können auf den Filetyp eingestellt werden. Durch den Einsatz der ASM-Lib API kann man die CPU Auslastung reduzieren und eventuell von Caching Hints profitieren. Die Limitierungen von manchen Filesystemen in heterogenen Landschaften bzgl. File- und Filesystemgrößen werden aufgehoben. Einfach, Zweifach und Dreifachspiegelung zum Schutz vor Storage Array Fehlern kann implementiert werden.

Vorteile der redundanten Datenspeicherung in Diskgruppen

Zum vollen Verständnis der Vorteile, die Diskgruppen mit der Redundanz Normal bieten, ist ein kurzer Exkurs hin zu den File Templates notwendig.

Beim Erstellen eines Datenbankfiles in einer Diskgroup wird der Filetyp ermittelt und unter Anwendung des zugehörigen File Templates Attribute, das auch die Lokation der Allocation Units, also der Filebruchstücke, beschreibt, optimal abgespeichert. Weitere, auf den Filetyp bezogene, Attribute sind die Anzahl der Kopien der AUs und auch Striping Attribute wie etwa Fine Striping oder Coarse Striping

Beim Anlegen einer Diskgruppe werden, abhängig von der gewählten Redundanz, Default Templates in den Headern der ASM Disks gespeichert. Somit können sich also die Speicherattribute bei gleichem Filetyp abhängig von der gewählten Redundanz der Diskgruppe unterscheiden.

Der Administrator der Diskgruppe hat die Möglichkeit, vorhandene Templates zu variieren oder neue Templates hinzuzufügen. Will man beispielsweise ein User-Defined Template auf alle zukünftig angelegten Datafiles anwenden, so erreicht man dies am einfachsten durch die Modifikation des Parameters `db_create_file_dest`.

```
ASM> ALTER DISKGROUP data ADD TEMPLATE mytemplate ATTRIBUTES (HIGH FINE);  
RDBMS> ALTER SYSTEM SET DB_CREATE_FILE_DEST = '+data(mytemplate)';
```

Eine Übersicht über die vorhandenen File Templates zeigt die identifizierbaren und auch variierbaren Attributsammlungen.

File Type	Default Templates
Control files	CONTROLFILE
Data files	DATAFILE
Redo log files	ONLINELOG
Archive log files	ARCHIVELOG
Temporary files	TEMPFILE
Data file backup pieces	BACKUPSET
Data file incremental backup pieces	BACKUPSET
Archive log backup piece	BACKUPSET
Data file copy	DATAFILE
Persistent initialization parameter file (SPFILE)	PARAMETERFILE
Flashback logs	FLASHBACK
Change tracking file	CHANGETRACKING
Data Pump dumpset	DUMPSET
Automatically generated control file backup	AUTOBACKUP
Cross-platform transportable data files	XTRANSPORT
Flash file	FLASHFILE
Oracle ASM Persistent initialization parameter file (SPFILE)	ASMPARAMETERFILE
Oracle ASM Persistent initialization parameter file (SPFILE) backup	ASMPARAMETERFILEBACKUP
Oracle Cluster Registry file	OCRFILE
Oracle ASM Dynamic Volume Manager volumes	n/a

*Abb. 1: File Types Supported by Automatic Storage Management
Quelle: Oracle® Database Storage Administrator's Guide 11g Release 2 (11.2)*

Der Hauptgrund für die Wahl der Redundanz Normal auf Diskgruppenebene ist die Flexibilität der File Templates bezogen auf das Mirroring denn dieses Attribut kann die Werte Mirror für die einfach Spiegelung, High für die Zweifachspiegelung oder External für den Fall dass man auf die Spiegelung verzichten will, annehmen.

Wenn man einen Blick auf den unter 11.2 ebenfalls supporteten ASM Dynamic Volume Manager wirft, erkennt man ähnliche Verhaltensweisen. Bei Diskgruppen mit Redundanz External sind die Volumes und die darin angelegten Filesysteme stets ohne Mirroring. Im Gegensatz dazu hat man bei der Konfiguration von Volumes in Diskgruppen mit normaler Redundanz die freie Wahl der Redundanz.

Weitere Gründe für den Einsatz von ASM mit Redundanz Normal sind die Möglichkeit der Datenspiegelung über Storage Systeme hinweg, auch ohne RAID Controller, oder auch die Möglichkeit der kostengünstigen Realisierung von Extended Cluster Konfigurationen. Das Feature Fast Mirror Resync zur Erhöhung der Verfügbarkeit und auch das Feature Preferred Read Failure Groups basieren auf Diskgruppen mit Redundanz.

Split Mirror

Um Diskgruppen mit Redundanz Normal für Split Mirror Techniken zu verwenden, braucht man exakt zwei Failure Groups, also zwei Sets, bestehend aus ASM Disks, mit etwa den gleichen physikalischen Eigenschaften. Auf die Erstellung von exakt zwei Failure Gruppen muss man schon beim Kreieren der Disk Group achten, da ja per default jede Disk eine eigen Failure Group ist.

Baut man nun eine Diskgroup mit folgender Syntax auf,

```

CREATE DISKGROUP U01 NORMAL REDUNDANCY
    FAILGROUP U01A DISK
        '/dev/sdc1' NAME U01A001,
        '/dev/sdc2' NAME U01A002
    FAILGROUP U01B DISK
        '/dev/sdd1' NAME U01B001,
        '/dev/sdd2' NAME U01B002
    ATTRIBUTE
        'au_size'='4M', --required
        'compatible.asm' = '11.2',
        'compatible.rdbms' = '11.2',
        'sector_size'='512';

```

so werden bei der Datenspeicherung die Allocation Units etwa gleichverteilt in den Failgroups U01A und U01B gespeichert und die zugehörigen Spiegel der AUs befinden sich jeweils in der anderen Failure Group. Insgesamt erhält man hier also in Summe zwei logische Spiegel. Die ASM Instanz berechnet die Position der AUs und der zugehörigen Kopien bei der Erstellung der Files. Im Falle des Verlusts einer Disk berechnet die ASM Instanz die neuen Koordinaten der AUs und betreibt das Rebalancing bis die voreingestellte Redundanz wieder hergestellt ist. Verliert man eine komplette Failgroup, verzögert sich das Rebalancing bis eine neue Failgroup manuell hinzugefügt wird.

In der Praxis trennt man auf der Quellseite eine Hälfte des Spiegels folgendermassen ab:

- RDBMS Instanz schliessen
- Diskgroup dismounten
- alle ASM Disks einer Failure Group entfernen
- Diskgroup mit Force Option mounten
- Optional: RDBMS Instance wieder starten
- Neue Failure Group bereitstellen und ans ASM bekannt geben
- Nachpflege der Metadaten: Löschen der fehlenden Failure Group bzw. der Disks

Nach dem Umhängen der Disks auf die Zielseite stehen folgende Tätigkeiten aus:

- Environment vorbereiten (init.ora ...)
- Ownership der Disks anpassen
- asm_diskstring anpassen
- Diskgruppe mit Force Option mounten
- Optional: RDBMS Instance wieder starten
- Neue Failure Group hinzufügen und Rebalancing monitoren
- Nachpflege der Metadaten: Löschen der fehlenden Failure Group bzw. der Disks

Bei dem vorgestellten Verfahren gibt es ein paar Punkte, die man zusätzlich beachten sollte. Hat man beispielsweise die RDBMS Binaries in der Diskgroup mit abgespeichert, kann man diese ebenfalls klonen bzw. provisionieren. Alle File Templates, die zum Speichern der Datenbankfiles oder Volumes verwendet wurden, dürfen im Vorfeld nicht auf externes Spiegeln abgeändert werden, da ja sonst die Failure Groups keine logischen Spiegel repräsentieren. Ausserdem sollte man das Abändern der DBID in Betracht ziehen, um Unstimmigkeiten im Recoverycatalog zu vermeiden. Die Verwendung der ursprünglichen Instanz- und Servicennamen stellen weitere Fehlerquellen dar. Dass die vorab vorhandene init.ora ein Verweis auf das im ASM liegende SPFile ist, sollte selbstverständlich sein. Die Anpassungen einiger Initialisierungsparameter erleichtern den zukünftigen Umgang mit der geklonten Datenbanken. Falls der Diskgroupname anschliessend geändert werden soll, ist das in 11.2 offiziell supported. Bei vorhergehenden Versionen helfen evtl. der Oracle Support und das Binary kfd weiter.

Filesystem Free Space Monitoring

Da das Abändern von Filesystem Templates bezogen auf die Redundanzeinstellungen möglich ist, sollte man in grösseren Umgebungen beim Free Space Monitoring auf Diskgruppenebene darauf achten, dass beim Evaluieren der Metrik die Redundanz der Diskgruppe, und nicht die derzeitige Einstellung in den Default Templates Beachtung finden. Verwendet man die Metrik free_mb und geht vom maximalen, dreifachen Mirroring aus, ist man stets auf der sicheren Seite.

Die Metrik "Used % of Safely Usable" ist nicht geeignet, da der Redundancy Faktor aus der Diskgruppe Verwendung findet, wie folgendes Zitat aus dem Oracle Database and Database-Related Metric Reference Manual 10g Release 2 (10.2) zeigt

Description

This metric shows the percentage of safely usable space used by a disk group. Usable free space of a disk group depends on the redundancy

Data Source

Used % of Safely Usable = $100 - (\text{usable_file_mb}/\text{usable_total_mb}) * 100$

where $\text{usable_total_mb} = \text{total_mb} - \text{required_mirror_free_mb} / \text{redundancy_factor}$

total_mb and required_mirror_free_mb are derived from the view column and redundancy factor is 1 for External Redundancy Disk Group, 2 for Normal Redundancy Disk Group, and 3 for High Redundancy Disk Group.

Shared Pool Usage

Am Beispiel der Shared Pool Usage erkennt man den Ressourcenbedarf einer ASM Instance teilweise. Die von Oracle bereitgestellten Regeln für Datafiles, Tempfiles und Redo-logs sind:

For disk groups using external redundancy,
every 100 GB of space need 1 MB of extra shared pool plus 2 MB

For disk groups using normal redundancy,
every 50 GB of space need 1 MB of extra shared pool plus 4 MB

For disk groups using high redundancy,
every 33 GB of space need 1 MB of extra shared pool plus 6 MB

Quelle: Oracle® Database Storage Administrator's Guide 11g Release 2 (11.2)

Hat man viele archivierter Redo-logs in der Flash Recovery Area so sollten diese in gleicher Weise zur Dimensionierung herangezogen werden. ADVN Volumes müssen ebenfalls in die Shared Pool Grösse eingerechnet werden.

Um den zusätzlichen Bedarf an Shared Pool in der RDBMS Instanz abschätzen zu können, kann man die View V\$SGASTAT selektieren, vgl. Doc ID 465039.1.

Praktische Messungen haben gezeigt, dass der Mehrbedarf an Shared Pool durch die Speicherung der Datenbankfiles im ASM sehr gering ist. Als Faustregel kann man für die Redundanz Normal die Datenbankgrösse um 2 Grössenordnungen reduzieren und das Ergebnis mal 5 nehmen.

Bei einer Datenbankgrösse von 100 GB ergeben sich demnach 500 B für die RDBMS Instanz.

Für Redundanz External oder High rechnet man noch 30% weg bzw. hinzu.

Zusammenfassung

Der Einsatz von ASM insbesondere mit Redundanz Normal auf Diskgruppenebene ist ratsam, da eine Split Mirror Verfahrens und somit Clone- und Provisioningmöglichkeiten für Datenbanken und RDBMS Binaries genutzt werden können und der Ressourcenbedarf nur marginal wächst.

Falls man im Automatic Storage Management die Default File Templates modifiziert, kann ein Anpassung der Filesystem Metriken notwendig werden.

Kontaktadresse:

Ralf Moosandl

Trivadis AG
Europa Strasse, 5
CH-8032 Zürich

Telefon: +41 (0) 79-571 3648
Fax: +41 (0) 12-345 6788
E-Mail ralf.moosandl@trivadis.com
Internet: www.trivadis.com