

Exadata Database Machine Resource Management – teile und herrsche!

**Konrad Häfeli
Trivadis AG
CH-3014 Bern**

Schlüsselworte:

Oracle Database Machine, Exadata, Ressource Manager, IORM, DBRM

Einleitung

Die Exadata Database Machine zeichnet sich durch enorme Performance aus. Dies auch im OLTP Bereich, was sie auch als optimale Konsolidierungsplattform einsetzen lässt. Sobald ich aber mehrere Datenbanken auf dem System oder mehrere Applikationsschemas in einer Datenbank habe, habe ich konkurrenzierende Vorgänge welche früher oder später zu Engpässen führen können. Wie stelle ich nun sicher, dass meine wichtigste Anwendung immer minimale Systemressourcen bekommt?

Den Database Resource Manager gibt es schon seit einiger Zeit. Mit Exadata wurde der I/O Resource Manager eingeführt welcher auf Storagezellenebene die verfügbare Bandbreite den verschiedenen Anwendergruppen zuteilen lässt und somit eine Priorisierung der I/O Requests macht. In diesem Vortrag werden Konzepte sowie die scriptmässige Umsetzung von Inter-Database und Intra-Database Konfigurationen anhand eines Kundenprojektes aufgezeigt.

Über die Evaluation der Oracle Exadata Database Machine als strategische DB Server Platform ist schon einiges geschrieben worden, dies soll nicht Gegenstand dieses Berichtes sein. Kurz aufgelistet waren bei meinem Kunden (grosses Unternehmen in der Telco Branche) folgende Beweggründe ausschlaggebend:

- Ersatz der Storage Systeme geplant
- Bestehende Standby System waren nicht gleichwertig der Primary Systeme
- Standby Systeme waren nicht mehr ausreichend
- Einsatz von Oracle 11gR2 geplant
- Bestehende Backup Infrastruktur und Abläufe zu komplex
- Energieverbrauch nicht mehr angemessen an heutige Standards
- Mehrere Platformkonsolidierungen standen an
- Lizenz-Kostenreduktion gewünscht

Auch das Standard-Setup von Oracle ACS (Advanced Customer Services) ist bekannt und die Migration einer Datenbank mit Oracle Golden Gate (Near Zero Downtime Migration) auf die „jungfräuliche“ Database Machine habe ich an der letzten DOAG Konferenz in einem Vortrag¹ präsentiert. Ich gehe im Folgenden auf die Überlegungen und Konzepte beim Einsatz als Konsolidierungsplattform ein.

IST Situation, Zielplattform

Das zu konsolidierende Mengengerüst bestehe aus ca. 50 Datenbanken unterschiedlicher Grösse (100GB bis mehrere Terabyte), verschiedener Versionen, hauptsächlich Oracle 10.2.0.4, meist single Instanz, aber auch RAC Datenbanken. Die Infrastruktur wird applikationsbezogen von zwei Teams betrieben, welche im DB Bereich eng zusammenarbeiten. Die Verfügbarkeit wird meist mittels Failover-Cluster-Konfiguration gemacht, wobei da ein Stretched Veritas Cluster über zwei nahegelegene (3km) Datacenter im Einsatz steht. Die bestehende RAC Konfiguration ist mittels Dataguard zu einem MAA (Maximum Availability Architecture) System zusammen gefasst.

Das Sizing der Exadata Platform wurde mittels Trivadis Configuration Manager (TVD-ConfManTM) unterstützt. Über alle bestehenden Systeme wurden Metriken (KPIs) definiert und gemessen und danach mit den performancerelevanten Zahlen auf der Exadata verglichen. Dabei stellte sich heraus, dass eine Half-Rack Konfiguration ausreicht um die bestehende Last zu bewältigen sowie für die geplanten neuen Datenbanken genügend Reserve bereit zu stellen. Durch die Verfügbarkeitsanforderungen im Telco Bereich (24*365) war auch eine MAA Konfiguration (je ein Half-Rack in zwei Datacenter 80km getrennt) nötig.

Konzeptüberlegungen und Bewertung

Durch das Zusammenfassen aller Datenbank-Infrastrukturen und deren Betrieb in einem Team (neue, horizontale Organisation) können die applikationsbezogenen Unterteilungen der Datenbanken ausser Betracht gelassen werden. Es stellt sich aber dennoch die Frage ob eine der beiden Extremvarianten überhaupt untersucht werden soll:

- 1:1 Migration, d.h. für jede alte DB eine Neue auf der Exadata

¹ http://www.trivadis.com/uploads/tx_cabagdownloadarea/Konrad_Haefeli_OracleGoldenGate_DB-Migration-auf-Exadata-mit-Fallback_NOV2010.pdf

- „all in one“ DB, alle produktiven DB-Schemas werden in eine Datenbank migriert/konsolidiert

Bei der ersten Variante werden keine Synergien und Möglichkeiten der verfügbaren RAC Infrastruktur genutzt:

- Servicehoch- (höchst) Verfügbarkeit
- Scaleout-Möglichkeit
- Reduktion der Ressourcen im Bereich Memory, Background-Prozesse und Anzahl Tablespace

Die zweite Variante gibt Problempunkte im Bereich Abhängigkeiten bzw. Abgrenzung der einzelnen bestehenden Applikationsschemas:

- Z.T. sind viele globale Synonyme im Einsatz (erst noch mit gleichen Namen über die verschiedenen Applikations-DBs hinweg)
- Überprivilegierte User
- Changemanagement unterschiedlich
- Klumpenrisiko bzw. Seiteneffekte untereinander

Es kristallisiert sich also ein Kompromiss heraus, bei dem die Anzahl der Datenbanken (massiv) reduziert werden soll, dies aber in Anbetracht der Gemeinsamkeiten bzw. Unterschiede.

Folgende Definition wurde als generelle Grundlage für die Konsolidierung auf (Exadata) RAC Infrastruktur gemacht:

1. *Alle Applikationen verbinden sich nur noch über Datenbank-Service Namen und haben keinen Bezug mehr zur physischen DB resp. zum DB Namen*
2. *Die Applikationen sind fähig (einerseits mittels Oracle Net Konfiguration, andererseits im Handling der Oracle Verbindungen) einen kurzfristigen Verbindungsabbruch zu überstehen und einen neuen Verbindungsaufbau zu machen*

Die erste Anforderung lässt eine Abstraktion des Datenzugriffs zu, sodass es keine Rolle spielt wo sich diese physisch befinden. Die Zweite ist nötig, damit die Verschiebung der DB-Services über die RAC-Knoten (und evtl. MAA Knoten) hinweg keine Applikations-Abstürze hervorruft.

Konzeptbeschreibung:

Auf den vier Knoten des Exadata-Half-Rack sollen die bestehenden Daten themenspezifisch auf möglichst wenigen Datenbanken konsolidiert werden. Die Zugriffe darauf werden über DB-Service Namen abstrahiert. Alle Themen-DBs haben gestartete Instanzen auf jedem Knoten, wobei die Applikationsservices einer DB immer nur auf einem Knoten laufen und mittels Servicefailover hochverfügbar gehalten werden. Die Services werden so auf die Knoten verteilt, damit eine ausgeglichene Last auf dem System ist.

Vorteile:

Durch die Zusammenfassung der Datenbanken können bestehende Datenbanklinks weggelassen werden. Ein spezifischer DB-Service ist hochverfügbar, da ein Failover auf eine schon gestartete Instanz erfolgen kann. Eine mögliche Cachefusion-Problematik wird umgangen, da die jeweiligen Verbindungen auf dieselben Daten nicht über verschiedenen Instanzen hinweg gemacht wird (per

default kein scaleout, da die meisten Applikationen nicht „RAC-aware“ sind). Eine gute Lastverteilung ist auf der Granularität DB-Service sehr einfach möglich.

Die folgende Abbildung 1 zeigt zusammengefasst die redundanten Daten in den Exadata Storage Cells und darüber die vier Datenbank-Knoten mit der Oracle Grid-Infrastruktur (Cluster-Schicht), den ASM Instanzen (pro Server) und den verschiedenen thematisch konsolidierten Datenbanken (TH[a-d]). In der Grafik sind es 4 Datenbanken welche jeweils eine Instanz (I[1-4]) pro Knoten haben und auf denen die Services (S[a-d]) der Datenbanken verteilt sind.

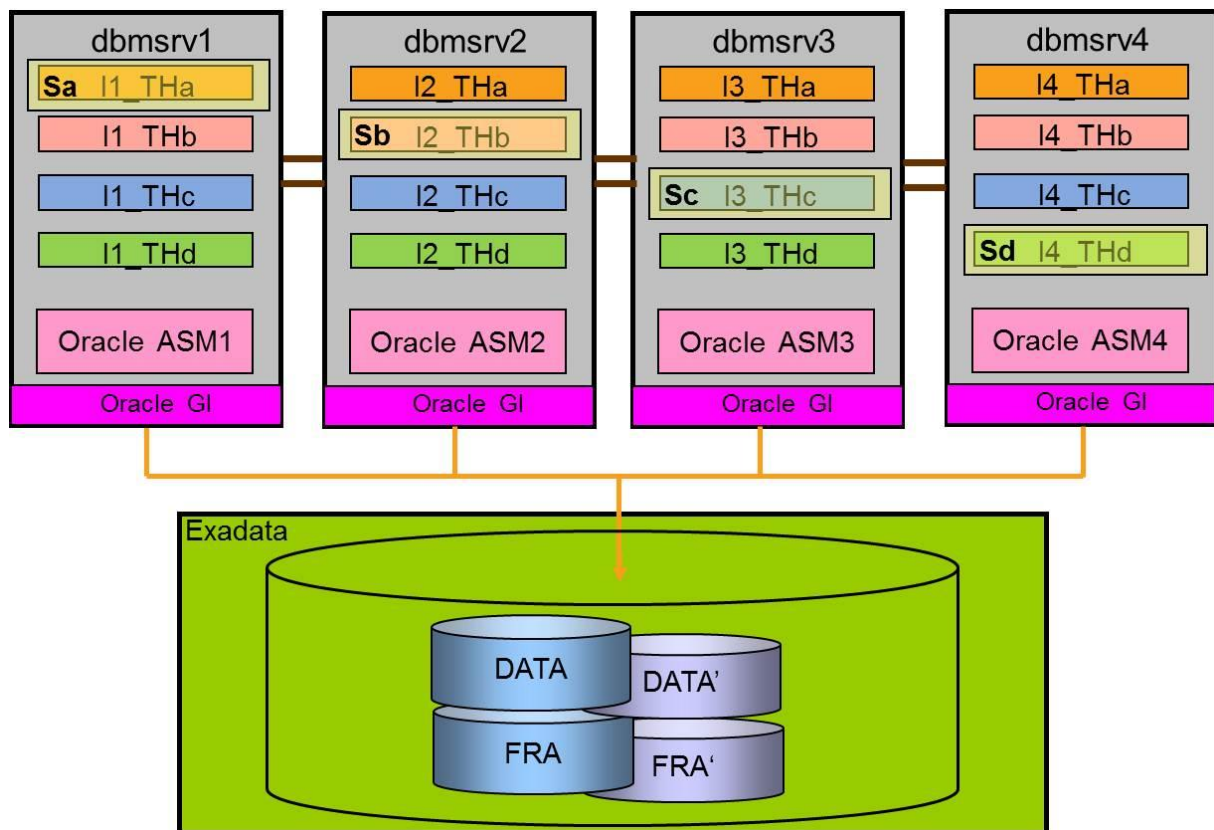


Abbildung 1: Konzeptgrafik Exadata Database Konsolidierung auf DB-Services

Mit dem Einsatz einer Exadata Plattform ist die beim alten Konzept mittels Stretched Cluster sichergestellte Disaster-Redundanz nicht gewährleistet. Aus diesem Grund wird beim neuen Konzept eine Maximum Availability Architecture (MAA) definiert. Diese stellt mittels Dataguard (Redologtransport und Apply) eine datenbankseitige Absicherung dar, welche im Fehlerfall die definierten Services hochfahren und bereitstellen kann. Ebenfalls ist mit Active Dataguard die Möglichkeit gegeben „readonly“ Services auf den Standby Datenbanken anzubieten, welche das transaktionsorientierte (Primary-)System entlasten können.

Die Abbildung 2 stellt die beiden Datacenter dar, welche im Wechsel oder sogar gemischt die Datenbanken im Primary und Standby-Modus beherbergen. Eine Überwachung kann optional aus einem dritten Datacenter gemacht werden, in dem ein sogenannter Dataguard-Observer konfiguriert wird, der den Failover der Primary-Site auf die Standby Site im Fehlerfall selbständig durchführt.

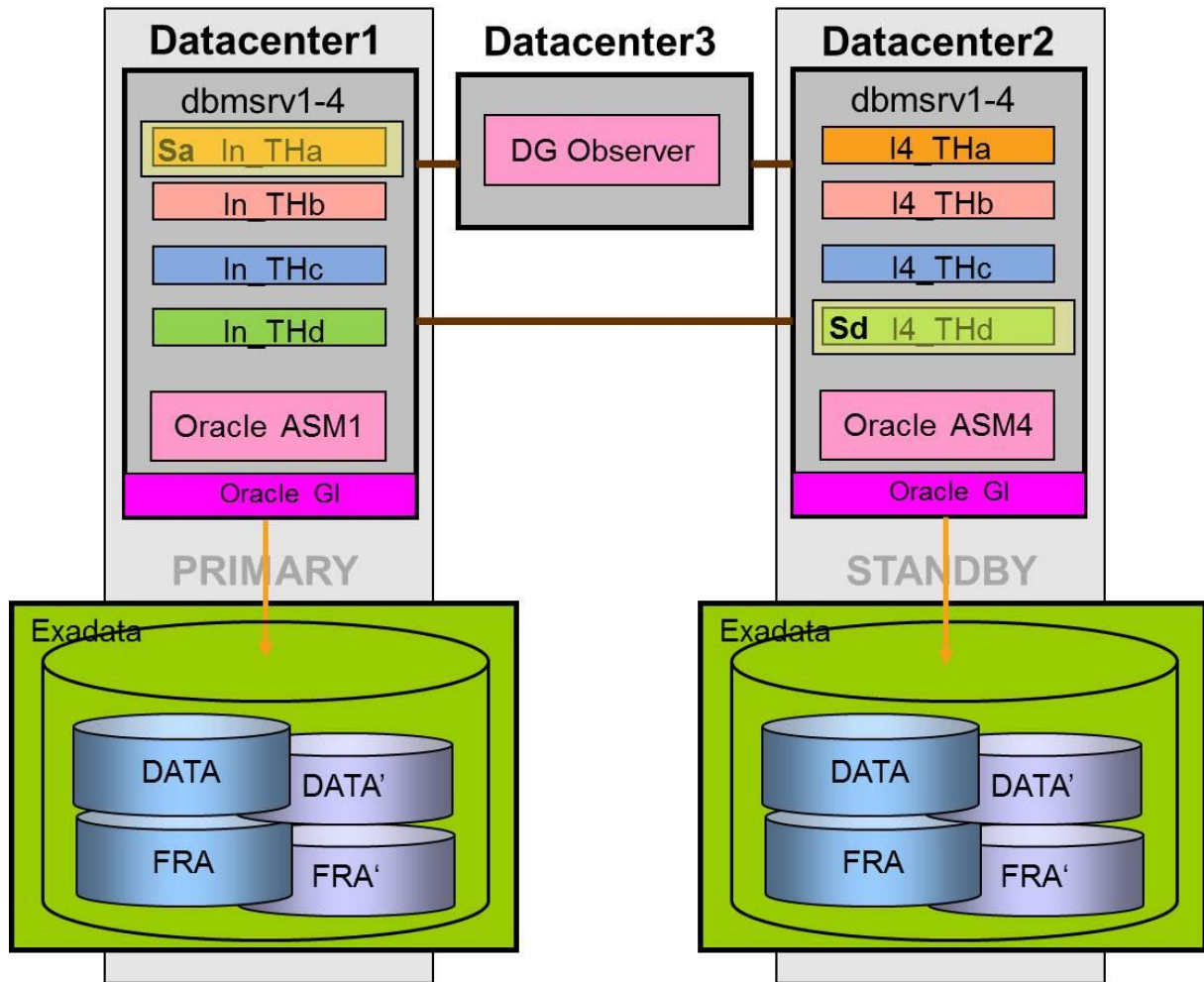


Abbildung 2: Konzeptgrafik Exadata MAA Konfiguration

Umsetzung der Konsolidierung

Die Zusammenfassung der Datenbanken sollte nach folgenden Kriterien erfolgen:

- Datenzugehörigkeit
 - Gleiche Applikationsfamilie
 - Gemeinsame Daten (Ersatz von DB-Links)
- Explizite Datentrennung
 - Trennung wegen nötiger Unabhängigkeit
 - Klumpenrisiko
- Security
 - Sensitive Daten
 - Schema braucht hohe Privilegien
- Verfügbarkeit

- Multi Instanz DB
- Single Instanz DB
- Dataguard
- DG Protection Level
- RDBMS Version
 - Major/Minor Releases
 - Patch/Patchset

Für die Umsetzung der Konsolidierung musste zuerst ein Katalog aufgestellt werden, der die Kriterien der Zusammenfassung der Datenbanken auflistet und pro DB bewertet. Diese zuerst als einfache Matrix gedachte Liste, stellte sich dann aber doch als recht aufwändig heraus, weil die verschiedenen Kriterien sich zum Teil ausschlossen, andererseits die verschiedensten Applikations-, Plattform-, Service-, und andere Manager dazu befragt werden mussten. Dass bei den meisten Verantwortlichen mehr Risiko als Vorteil in der Zusammenlegung der Datenbank gesehen wurde, liegt auf der Hand. Die Koordination der verschiedenen Aktivitäten über einen grösseren Teil der komplex verstrickten Funktion eines Telco-Unternehmens war aber doch recht schwierig und bedurfte eines einflussreichen Projekt-Sponsors.

Dieser umfangreiche Konsolidierungs-Fragebogen hatte folgende Grobstruktur:

- Technische Information pro Datenbank (Name, Version, Grösse, Ressourcenverbrauch,...)
- Applikatorische Informationen pro Datenbank (Name, Art der Applikation, Art des Datenbankzugriffs, Anforderungen, an Verfügbarkeit und Sicherheit,...)
- Betriebliche Informationen (Backup, Monitoring, Datenfilesystemtyp, Batch-Jobs,...)
- Verantwortlichkeiten (Manager, Team, DBA, Lieferant,...)
- Test/Integrations Umgebung (Setupart, Testdaten, Skalierung, ...)
- Informationen zu Migrations Szenario (Versionen, Anpassungen , Kosten für Anpassungen, Konsolidierungsfähigkeit,...)

Die Auswertung er gesammelten Daten, wobei nicht für alle Projekte innert nützlicher Zeit alle Informationen beschafft werden konnte, ermöglichte folgende Aufteilung und Priorisierung:

- Implementation aller neuen Datenbank-Ressourcenanfragen auf der neuen Plattform
- Migration von ein paar unkritische Datenbanken als „quick wins“, für die Lieferung von nötigen Erfahrungswerten in der Migration und Betrieb
- Identifikation der zeitlich kritischen Datenbanken und Behandlung mit Projektleiterfokus, damit auch diese dann termingerecht konsolidiert sind

Erfahrungen

Der hohe Innovations-Rhythmus in der Unternehmung machte die schnelle Umsetzung der Konsolidierungsplattform nötig. Folgende Aufgaben mussten zuerst angegangen werden:

- Namenskonzepte für Datenbanken, Instanzen und Services

- Entwicklung von Templatescripts für die Erstellung von standardisierten Datenbanken
- Entwicklung von Scripts für die Konfiguration der Datenbanken und Services im Cluster
- Konzept und Implementation des Ressourcen-Managements (Exadata IORM, DBRM)
- Implementation des Backupkonzepts
- Implementation des Monitoringkonzepts

Die Angst der involvierten Applikations-Verantwortlichen und Betreiber vor dem Zusammenlegen ihrer Daten in einer Datenbank war um einiges grösser als erwartet. Der Widerstand gegen die Konsolidierung hatte zu einem grossen Teil damit zu tun, dass Mehraufwand durch Abgabe von bestehenden Privilegien befürchtet wurde und Reduktion der Verfügbarkeit durch Abhängigkeiten von weniger stabilen Systemen. Durch ein moderates Auslegen des Konsolidierungsdrucks, indem die heikelsten Datenbanken eins-zu-eins migriert wurden, konnten die Anwender von der guten Funktionalität der Plattform überzeugt werden und sind so eventuell in einer späteren Phase bereit das Datenbankschema mit anderen zusammen zu legen.

Erst der produktive Betrieb hat die Nützlichkeit der MAA Konfiguration wirklich aufgezeigt. Durch Switchover von einem Datacenter auf das Andere, konnten grosse Systempatches ohne Risiko implementiert werden. Auch bei den aufgetretenen technischen Fehlern (Diskausfall, Motherboard Austausch) welche nicht immer ohne Probleme gelöst werden konnten, war das zweite System von gutem Nutzen. Durch die Übernahme der Services war der nötige Hardware-Austausch ohne Last durchführbar.

Wichtig beim Betrieb einer Exadata-Plattform ist eine enge Zusammenarbeit mit dem Oracle Support, welcher sich speziell bei Exadata-Service-Requests sehr engagiert und zeitnahe Lösungen bringt. Denn eines ist klar die Plattform, Software wie Hardware, braucht ein intensives Lifecycle-Management und dazu gehört das beheben von unvorhergesehenen Problemen dazu.

Die Kostenfrage wird natürlich bei so einer Konsolidierung immer wieder in den Vordergrund gestellt. Die Erfahrungen im Bereich Lizenzen waren sehr gut, wobei da natürlich auch noch gute Verhandlungen mit Oracle als Early-Adopter der Exadata-Technologie geführt werden konnten. Der Betriebsaufwand, nach den nicht zu vernachlässigenden Setup-Anstrengungen, kann gegenüber der herkömmlichen Plattform als definitiv kleiner ausgewiesen werden. Ist dem DBA das Betriebssystem bzw. Netzwerk nicht unbekannt und hat er keine Abneigung unter Anweisung solche Aufgaben auch durchzuführen, kann dieser eine Exadata-Plattform nach dem Setup unterhalten.

Es gibt noch viel Interessantes im Bereich Funktion und Anwendung beim Einsatz der Exadata Database Machine als konsolidierter Datenbankserver zu erzählen, der Autor steht gerne für Fragen zur Verfügung.

Kontaktadresse:

Konrad Häfeli

Trvadis AG
Papiermühlestrasse 73
CH-3014 Bern

Telefon: +41(0)31-928 09 60
Fax: +41(0)31-928 09 64
E-Mail: konrad.haefeli@trivadis.com
Internet: www.trivadis.com