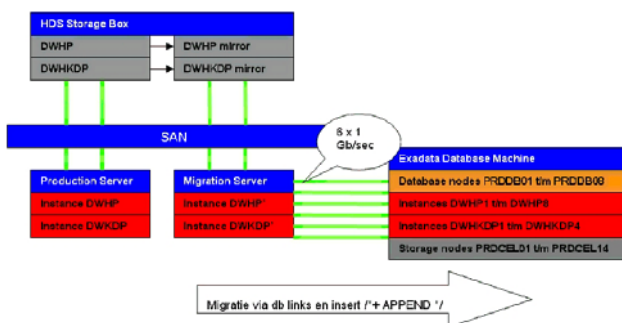


# Migreren datawarehouse naar Exadata

## Beschrijving van project binnen Atos Origin (2)

Atos Origin is door een klant gevraagd om de datawarehouse omgeving te migreren. De huidige omgeving bestaat uit 8 Oracle9i standalone databases met een totale omvang van 40 TB. Qua I/O capaciteit, performance en scalability loopt deze datawarehouse omgeving op zijn eind en de gebruikers wachten met smart op een verbetering. Als nieuwe platform is gekozen voor een Oracle 11g RAC omgeving op Exadata. In een serie van drie artikelen beschrijven we hoe dit project is aangepakt. In het eerste deel is aandacht besteed aan de voorbereiding en configuratie van de Exadata-database machine. Nu kijken we naar de migratie van de oude naar de nieuwe omgeving.

De oude omgeving bestaat uit acht Oracle9i standalone databases die op verschillende databaseservers zijn geplaatst. Het onderliggende operating system is Sun Solaris. De nieuwe omgeving bestaat uit één full rack V2 Exadata (acht databaseservers in een RAC configuratie en veertien storageservers). Het OS is Linux (OEL 5). Eerder is via een instance mapping document vastgesteld over hoeveel instances een database beschikt en op welke database nodes de instances draaien. In dit artikel wordt verder aandacht besteed aan de migratie van de twee grootste databases: database DWHP (18 TB) en database DWHKDP (13 TB). De migratie wordt uitgevoerd in een aantal stappen. Voor de migratie van de acht databases is een window beschikbaar van 48 uur.



Figuur 1

### Stap 1: klaarzetten bronomgeving

Als eerste wordt een migratieserver ingericht en de twee brondatabases worden via het afsplitsen van een mirror van productie op deze migratieserver opgestart. De migratie server is via deze zes dedicated 1 Gb lijnen verbonden met de Exadata-machine (zie de figuur 1).

### Stap 2: migratie via CTAS en INSERT /\*+ APPEND \*/

De migratie moet zo snel mogelijk plaatsvinden. De export/import methode is bij databases van deze omvang geen optie. De methode die wordt gekozen is om via een export en import de tabeldefinities over te zetten. Voorafgaand aan de migratie is een lijst gemaakt met de compressiemethode per tabelpartitie en dat zal tijdens de implementatie tevens moeten worden geïmplementeerd.

De volgende stappen worden onderscheiden:

- Stap 2a: aanmaken database links van doel naar bron database
- Stap 2b: aanmaken tabeldefinities in doeldatabase op basis van metadata export
- Stap 2c: overhalen tabeldata van bron naar doel m.b.v. insert /\*+ APPEND \*/ statement
- Stap 2d: comprimeren tabelpartities in de doel database
- Stap 2e: aanmaken indexen, constraints, packages, views etc.. in de doeldatabase.

Dankzij de zes dedicated 1 GB verbindingen en doordat de operaties allen parallel worden aangevoerd, is een maximale doorvoersnelheid van 1,8 TB per uur gehaald. Dat betekent dat de grootste database DWHP (16 TB + overhead) in minder dan negen uur is gemigreerd! Van deze 16 TB is 2/3 tabeldata, wat daadwerkelijk via de databaselink getransporteerd is naar de Exadata. Door de compressie is de database van 16 TB veel kleiner in de Exadata-omgeving.

## Compressie in Oracle databases

Oracle heeft in versie 9i basic table compression geïntroduceerd. Hiermee kan data tijdens bulkoperaties worden gecompriemd. Met behulp van basic table compression kan diskruimte worden bespaard en neemt de query performance toe. Het updaten van een tabel zal echter meer tijd kosten omdat de data dan eerst moet worden gedecomprimeerd. Na afloop van een update operatie wordt de data niet meer gecompriemd bij deze vorm van compressie. In Oracle 11g is OLTP compression geïntroduceerd. Deze vorm van compressie biedt veel voordelen ten opzichte van basic compression. Data wordt nu niet alleen bij bulk operaties maar bij alle vormen van DML gecompriemd. Daarnaast is het compressie algoritme verbeterd en duren DML operaties op gecompriemde tabellen nu net zo lang als op niet gecompriemde tabellen. Bij OLTP compressie wordt in de regel een compressie factor van 2 tot 3 bereikt. Voor deze vorm van compressie is wel de advanced compression option licentie vereist. Met de introductie van Oracle Exadata heeft Oracle tevens een nieuwe vorm van compressie geïntroduceerd, Exadata hybrid columnar compression (EHCC). Bij deze vorm van compressie wordt een ander algoritme gebruikt waarbij de data niet per rij maar per kolom wordt opgeslagen. Op deze manier zijn veel hogere compressie ratio's te behalen, variërend van factor 10 tot 15. Ook de query performance neemt toe. EHCC compressie werkt alleen bij bulk operaties zoals SQL\*Loader direct load operaties, INSERT statement met de APPEND hint, Create table as select

(CTAS) en parallel DML. EHCC compressie heeft wel een nadelig effect op de performance van UPDATE operaties. En tevens wordt de gemuteerde data naderhand niet gecompriemd weggeschreven. Er moet dus goed worden gekeken naar de aard van de tabellen om vast te stellen welke tabellen in aanmerking komen voor EHCC compressie. Voor tabellen die frequent gemuteerd worden is OLTP compressie beter geschikt. Dit tevens omdat er anders een locking probleem kan ontstaan. (Er bevindt zich nu immers veel meer data in een Oracle block, dus de kans dat 2 processen eenzelfde block met data willen is veel groter.). Binnen EHCC compressie zijn er 4 varianten: Query Low, Query High, Archive Low en Archive High. Bij de 2 Query varianten is de compressie geoptimaliseerd voor query performance en bij de 2 Archive varianten is de compressie geoptimaliseerd voor maximale ruimte besparing. Query low en high moeten worden ingezet bij data waar frequent queries op worden uitgevoerd en Archive low en high bij historische data die niet vaak meer wordt geraadpleegd. Aan EHCC compressie zijn geen extra kosten verbonden. Dit is inbegrepen bij de aanschaf van een Exadata database machine. Tot slot nog even de opmerking dat alle vormen van compressie volledig transparant zijn vanuit applicatie perspectief. Wel moet echter goed worden nagedacht welke tabellen voor welke vorm van compressie in aanmerking komen. Dit is een gezamenlijke activiteit voor de Oracle DBA en de Oracle applicatie ontwikkelaar.

## Stap 3: postmigratie acties

Na afloop van de migratie moeten nog wat postmigratieactiviteiten worden uitgevoerd. Zo wordt de database in de archive log mode geplaatst, wordt force logging aangezet en worden alle statistieken verzameld. Het verzamelen van de statistieken duurt voor de database DWHKDP zes uur en voor de database DWHP ongeveer acht uur.

Om eenmalig alle statistieken te verzamelen is het belangrijk dat dit parallel wordt gedaan, anders kan het wel eens een langdurige actie worden. Omdat er acht instances beschikbaar zijn bij de migratie is ervoor gekozen om alle acht instances ook te gebruiken om eenmalig alle statistieken te verzamelen. Parallelisme wordt gezet op 256, wat betekent

dat per instance met een parallel van 32 de statistieken worden verzameld.

```
SQL> exec DBMS_STATS.SET_PARAM('DEGREE','256');
SQL> exec DBMS_AUTO_TASK_IMMEDIATE.GATHER_OPTIMIZER_STATS;
```

## Migratieresultaten op gebied van compressie

De compressiefactor bij de databases hangt af van de gekozen compressie methode. Deze compressie methode is op tabel dan wel op (sub)partitie niveau bepaald en tijdens de migratie doorgevoerd. Met name bij database DWHKDP, waar veel archief data inzit en de EHCC Archive Low compressie (zie kader) is ingezet bij partities ouder dan zes maanden is

Database	Omvang Source DB	Omvang Target DB	Compressie factor
DWHP	16.567 GB	5.895	2,8
DWHKDP	12.288 GB	1.093 GB	11,8

De data in tabel 1 is verkregen op basis van de DBA\_SEGMENTS view. Dit bevat alleen de daadwerkelijk gevulde segmenten van de database. De databases zijn in de praktijk wat groter omdat er ook nog "lucht" in de datafiles zit en omdat de TEMPORARY en UNDO tablespaces ook onderdeel van de database vormen, maar niet zijn meegenomen in de tabel.

een spectaculaire compressie factor bereikt. In tabel 1 zijn de compressie resultaten aangegeven.

Uit tabel 1 blijkt dat de compressie aanzienlijk is, maar ook dat het veel verschilt qua database. Dit heeft te maken met de aard van de data. Met name data die niet of weinig meer wordt bekeken kan met de EHCC Archive compressie methode aanzienlijk worden verkleind. Als voorbeeld wordt de grootste tabel in de database genomen, voor het gemak noemen we deze even 'largest\_table'

In de bronomgeving blijkt deze groter dan 3 TB te zijn:

```
SELECT SUM (bytes) / (1024 * 1024 * 1024)
FROM dba_segments
WHERE segment_name = 'LARGEST_TABLE';

SUM (BYTES) / (1024*1024*1024)
-----
3810.5
```

In de doelomgeving is hier nog geen 300 GB van over!!

```
SELECT SUM (bytes) / (1024 * 1024 * 1024)
FROM dba_segments
WHERE segment_name = 'LARGEST_TABLE';

SUM (BYTES) / (1024*1024*1024)
-----
249,494751
```

Bij tabel 1 is er voor gekozen om tabelpartities <= 6 maanden te comprimeren met Query High en tabelpartities > 6 maanden met Archive Low. Als we inzoomen op partitie niveau levert dit de volgende resultaten:

```
SELECT partition_name,
       compress_for,
       ROUND (blocks * 8192 / (1024 * 1024 * 1024)) size_GB
FROM dba_tab_partitions
WHERE table_name = 'FM_NUTZUNG_EPLUS_FME' AND partition_name LIKE
'%2010%'
ORDER BY 1;
```

Partitie	Compress_for (11g only)	Size GB 11g	Size GB 9i	Compressie factor
P_2010_01	ARCHIVE LOW	6	105	17,5
P_2010_02	ARCHIVE LOW	6	99	16,5
P_2010_03	ARCHIVE LOW	7	107	15,3
P_2010_04	ARCHIVE LOW	6	109	18,2
P_2010_05	ARCHIVE LOW	7	112	16
P_2010_06	QUERY HIGH	13	125	9,6
P_2010_07	QUERY HIGH	14	128	9,1
P_2010_08	QUERY HIGH	13	125	9,6
P_2010_09	QUERY HIGH	14	128	9,1
P_2010_10	QUERY HIGH	15	132	8,8
P_2010_11	QUERY HIGH	0	0	Nvt
P_2010_12	QUERY HIGH	0	0	nvt

Tabel 2.

De oudere partities zijn via ARCHIVE LOW geoptimaliseerd voor compressie om ruimte te besparen. Dit levert een compressie factor van 17 a 18 op. De jongere partities zijn via QUERY HIGH geoptimaliseerd voor query performance. Dit levert nog steeds een force compressie factor op, maar wel aanzienlijk minder dan de archive low compressie. Van de laatste 2 partities in 2010 is nog geen data geladen. Daarom zijn deze 0 Gb.

## Migratieresultaten op gebied van performance

De performance van de database kan op verschillende manieren worden vastgesteld. Ten eerste is er de performance van de applicaties, dat wil zeggen de snelheid van de database zoals de gebruikers deze ervaren. Daarnaast is er de performance van de laadjobs. Hoewel deze uiteraard ook van belang zijn voor de totale datawarehouse oplossing heeft dit niet altijd direct invloed op de gebruikers. Maar in het geval van deze datawarehouse oplossing hebben de gebruikers er wel hinder van, omdat sommige laadjobs heel lang lopen en daarmee een negatieve invloed uitoefenen op de query performance van de gebruikers. Tot slot is er ook nog de database maintenance performance, zoals het maken van een backup en het verzamelen van database statistieken.

### Performance van de applicatie

Over het algemeen zijn de ervaringen vanuit gebruikersperspectief positief. De performance verbetering is erg afhankelijk van de applicatie architectuur. De applicaties worden gemigreerd van een standalone Oracle9i omgeving en er zijn wat zaken waar rekening mee moet worden gehouden.

Tijdens het testen blijkt bijvoorbeeld dat bestaande SQL hints een versturende factor vormen en in een aantal gevallen de database zelfs doen crashen en een ORA-0600 foutmelding veroorzaken. Het is dus zaak om de oude SQL hints te verwijderen.

## Configuratie APEX in de Exadata omgeving

Tijdens het migratie project zal tevens een applicatie worden vervangen door een nieuwe APEX applicatie. Hiervoor moet de Exadata Database Machine geschikt worden gemaakt voor APEX. Dat kan vrij eenvoudig en het is goed gedocumenteerd. Bij het aanmaken van de database met de DBCA moet de optie APEX worden aangevinkt, en ook de opties XDB en Oracle Text zijn noodzakelijk voor APEX. Daarna moeten nog wat postconfiguratie scripts in de database worden gedraaid. Deze staan hier onder weergegeven. En aansluitend kan via de browser worden aangelegd op de Apex omgeving:

```
[oracle@prdb04 ~]$ . oraenv
ORACLE_SID = [oracle] ? DWHP4
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1
is /u01/app/oracle
[oracle@prdb04 ~]$ cd $ORACLE_HOME/apex

[oracle@prdb04 apex]$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on
Fri Dec 10 14:22:11 2010

Copyright (c) 1982, 2009, Oracle. All
rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition
Release 11.2.0.1.0 - 64bit
Production
With the Partitioning, Real Application
Clusters, Automatic Storage
Management, OLAP,
Data Mining and Real Application Testing
options

SQL> @apxconf
PORT
-----
8080
Enter values below for the XDB HTTP listener
port and the password for
the Application Express ADMIN user.
Default values are in brackets [ ].
Press Enter to accept the default value.

Enter a password for the ADMIN user      []
Enter a port for the XDB HTTP listener
[8080]
...changing HTTP Port

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.
```

```
Session altered.

...changing password for ADMIN

PL/SQL procedure successfully completed.

Commit complete.

SQL> ALTER USER ANONYMOUS ACCOUNT UNLOCK;

User altered.

SQL>
#####
# PASTE below script on the SQL-prompt.
#####
DECLARE
  ACL_PATH  VARCHAR2(4000);
  ACL_ID    RAW(16);
BEGIN
  -- Look for the ACL currently assigned to
  '*' and give APEX_030200
  -- the "connect" privilege if APEX_030200
  does not have the privilege
  yet.

  SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_
  ACLS
  WHERE HOST = '*' AND LOWER_PORT IS NULL
  AND UPPER_PORT IS NULL;

  -- Before checking the privilege, ensure
  that the ACL is valid
  -- (for example, does not contain stale
  references to dropped users).
  -- If it does, the following exception
  will be raised:
  --
  -- ORA-44416: Invalid ACL: Unresolved
  principal 'APEX_030200'
  -- ORA-06512: at "XDB.DBMS_XDBZ", line ...
  --
  SELECT SYS_OP_R20(extractValue(P.RES, '/
  Resource/XMLRef')) INTO ACL_ID
  FROM XDB.XDB$ACL A, PATH_VIEW P
  WHERE extractValue(P.RES, '/Resource/
  XMLRef') = REF(A) AND
  EQUALS_PATH(P.RES, ACL_PATH) = 1;

  DBMS_XDBZ.ValidateACL(ACL_ID);
  IF DBMS_NETWORK_ACL_ADMIN.CHECK_
  PRIVILEGE(ACL_PATH, 'APEX_030200',
  'connect') IS NULL THEN

  DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE
  (ACL_PATH,
  'APEX_030200', TRUE, 'connect');
  END IF;

EXCEPTION
  -- When no ACL has been assigned to '*'.
  WHEN NO_DATA_FOUND THEN
  DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('power_
  users.xml',
  'ACL that lets power users to connect to
  everywhere',
  'APEX_030200', TRUE, 'connect');
  DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('power_
  users.xml','*');
END;
```

```
/
COMMIT;
#####

PL/SQL procedure successfully completed.

SQL> CREATE ROLE APEX_URL_DATASTORE_ROLE;

Role created.

SQL> GRANT APEX_URL_DATASTORE_ROLE to
APEX_030200;

Grant succeeded.

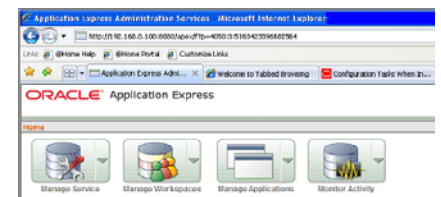
SQL> EXEC ctxsys.ctx_adm.set_parameter('file_
access_role',
'APEX_URL_DATASTORE_ROLE');

PL/SQL procedure successfully completed.

SQL> exit
```

Er kan via de browser verbinding worden gemaakt met: alle 8 database nodes, bijvoorbeeld voor prddb06 met: [http://<ip-adres node>/apex/apex\\_admin](http://<ip-adres node>/apex/apex_admin)

Nadat verbinding is gemaakt moet het wachtwoord worden gewijzigd van user admin en verschijnt aansluitend de Apex Home page:



Applicatie batchjob	Oude omgeving	Nieuwe Exadata omgeving
Job 1	7 uur	45 minuten
Job 2	29 uur	25 minuten

Tabel 3.

Ook wordt er in de applicatie bepaalde functionaliteit gebruikt (DBMS\_PIPE) die niet ondersteund wordt in een Oracle RAC omgeving. De DBMS\_PIPE functionaliteit (gebruikt voor het doorgeven van berichten tussen processen) moet worden vervangen door advanced queuing, welke wel wordt ondersteund in een RAC-omgeving.

Daarnaast is het ook van belang om de parallele kracht van de Exadata-omgeving in te schakelen. Dit heeft een grote invloed op de doorlooptijd van queries. Parallisme kan op tabelniveau worden ingesteld. Maar kan ook via SQL hints worden gestuurd. Zie onderstaand voorbeeld in verschil qua doorlooptijd:

#### Situatie 1: geen parallele uitvoering van statements

```
SQL> select count(*) from dwhcore.co_nutzung_outgoing_sms a;
Elapsed: 00:31:05.29
Costs: 2305000
```

#### Situatie 2: parallele uitvoeren op 2 instances

```
SQL> select /*+parallel(a,2,0)*/ count(*) from dwhcore.co_nutzung_
outgoing_sms a;
Elapsed: 00:01:00.26
Costs: 1278
```

#### Situatie 3: parallele uitvoeren op 8 instances met 16 slave processen per instance

```
SQL> select /*+parallel(a,8,16)*/ count(*) from dwhcore.co_nutzung_
outgoing_sms a;
Elapsed: 00:00:07.02
Costs: 19980
```

#### Performance van de applicatie batch jobs

Een tweetal applicatie batch jobs zorgden in de originele omgeving voor veel problemen. In de nieuwe Exadata-omgeving is de performance van de batchjobs spectaculair verbeterd. Zelfs zoveel dat wel meerdere malen is gecontroleerd of de batchjobs wel goed hebben gelopen en niet onverwacht waren afgebroken. Zie tabel 3 met de resultaten:

#### Performance van de backup

In de originele situatie diende altijd veel zorg en aandacht aan de backup en recovery van VLDB te worden besteed, met

```
-----
SESSION_KEY INPUT_TYPE START_TIME STATUS COMPRESSION_RATIO IN_SIZE OUT_SIZE TIME_TAKEN
-----
02 DB FULL 12/07/10 16:15 COMPLETED 1,02996295 032.44B 435.04B 03:20:43
```

09\_bijlschrift.

name de doorlooptijden en impact op performance vormden een probleem. Na afloop van de migratie is een backup test uitgevoerd van database DWHKDP. Deze database is na de migratie 1 TB groot. Als test is een online level 0 backup met RMAN gemaakt naar disk (de uiteindelijke backup configuratie wordt besproken in het derde en laatste artikel over dit project). Bij deze backup zijn per database node 2 channels gealloceerd voor de backup. De backup van de database was na 18 minuten (!) klaar:

Een dergelijk backup window is zeer zeker acceptabel als maintenance window en kan bijvoorbeeld 's nachts worden gescheduled om de performance impact te minimaliseren. De conclusie is dan ook dat de Exadata-machine een zeer acceptabele maintenance window heeft voor backup's en recoveries van zeer grote databases.

## Conclusie

Met de migratie van het datawarehouse van de Oracle9i naar de Oracle11g Exadata-omgeving is niet alleen een spectaculaire performanceverbetering doorgevoerd, maar ook een enorme ruimtebesparing. Het oude datawarehouse barstte letterlijk uit zijn voegen en met de nieuwe Exadata-architectuur is nu een datawarehouse neergezet waarmee de klant weer jaren vooruit kan en die allerlei nieuwe mogelijkheden biedt qua query windows en opslagcapaciteit. De grootste verandering (en projectuitdaging) vanuit het applicatieperspectief waren de overgang van een single instance naar een RAC-omgeving. Daarnaast zorgde de overgang van oracle9i naar oracle11g nog voor de nodige voorbereiding en aandacht.

## Referenties

Oracle Application Express Installation Guide



**Rob Lasonder en Jannes Arends** zijn als DBA werkzaam bij Atos Origin en maken deel uit van de Global Exadata Competence Group van het bedrijf.