ORACLE 11G AUTOMATIC MEMORY MANAGEMENT
Inderpal S. Johal, Data Softech Inc.

INTRODUCTION
Oracle has introduced Automatic Shared Memory Management in Oracle 10g and thus allows automatic tuning of five important component of SGA [Shared Global Area]. Oracle 11g has introduced Automatic Memory Management which will automates SGA as well as PGA size according to your workload by dynamically transferring the memory from SGA to PGA and vice versa.

ORACLE 10G AUTOMATIC SHARED MEMORY MANAGEMENT OVERVIEW
Starting with Oracle 9i, Oracle has made some SGA components [Shared pool, default Buffer Cache, Large Pool] as Dynamic and allows DBA to alter them without bouncing the database. The maximum size of SGA was restricted by new initialization parameter SGA_MAX_SIZE.

In Oracle 10g, the ASMM [ Automatic Shared Memory Management] has been introduced to relieve DBAs from sizing the following five main SGA components also called auto-tuned parameter :

1. Shared Pool - SHARED_POOL_SIZE
2. Large Pool - LARGE_POOL_SIZE
3. Java Pool - JAVA_POOL_SIZE
4. DB Cache - DB_CACHE_SIZE
5. Stream Pool - STREAM_POOL_SIZE

ASMM is driven by new initialization parameter SGA_TARGET and is managed by new background process MMAN [Memory Manager]. MMAN will take regular memory snapshots and evaluate and adjust the auto tuned parameters of SGA [as shown above] based on the database usage.

If we are using ASMM and If you set non zero values to any of these auto-tuned parameter, then those values are used as minimum value by ASMM.

There are still lots of SGA components that need to be adjusted manually but take the space is deducted from the total space available for SGA_TARGET itself. These parameter are also called as Manual parameters as shown below

1. DB_<KEEP|RECYCLE>_CACHE_SIZE - Dynamic
2. DB_nK_CACHE_SIZE (non default block size) - Static
3. LOG_BUFFER → Cannot changed after 10g Rel 2 - Fixed
4. Fixed SGA - Fixed

The SGA_TARGET will be limited by the SGA_MAX_SIZE value. The SGA_MAX_SIZE cannot be modified dynamically.
ORACLE 11G AUTOMATIC MEMORY MANAGEMENT

In order to implement Automatic Memory management, you need to set a new initialization parameter (MEMORY_TARGET). This parameter allows Oracle to automatically tune your instance memory by redistributing memory between SGA (System Global Area) and PGA (Program Global Area). MEMORY_TARGET parameter is dynamic and can be increased to maximum memory size specified by new initialization parameter, MEMORY_MAX_TARGET.

Even Oracle can manage your memory based on your non-zero value of MEMORY_TARGET, but still you can set some lower bound values for various caches below which database will not auto-tune that memory component.
MEMORY_TARGET W.R.T SGA_TARGET AND PGA_AGGREGATE_TARGET

Here we will check the various scenarios when MEMORY_TARGET is set to zero or non-zero values

1. When MEMORY_TARGET > 0
2. When MEMORY_TARGET =0 [Default] or not set

Scenario 1: When MEMORY_TARGET > 0

1. If you also set SGA_TARGET as well as PGA_AGGREGATE_TARGET, then
   - These values are considered as minimum values.
   - MEMORY_TARGET > (MEMORY_TARGET+SGA_TARGET) else you will get an error ORA-00838: Specified value of MEMORY_TARGET is too small, needs to be at least xxxM
   - If MEMORY_MAX_TARGET is also set, then MEMORY_TARGET can take the sum of values of SGA_TARGET and PGA_AGGREGATE_TARGET to MEMORY_MAX_TARGET.
   - If MEMORY_MAX_TARGET is not set then it will default to MEMORY_TARGET

2. If only SGA_TARGET is set but not the PGA_AGGREGATE_TARGET then
   - PGA_AGGREGATE_TARGET will be initialized to (MEMORY_TARGET–SGA_TARGET).
   - Database instance continue to auto-tunes both SGA_TARGET and PGA_AGGREGATE_TARGET.

3. If only PGA_AGGREGATE_TARGET is set but not the SGA_TARGET then
   - SGA_TARGET will be initialized to
     \[ \text{min} (\text{MEMORY\_TARGET}-\text{PGA\_AGGREGATE\_TARGET}, \text{SGA\_MAX\_SIZE} [\text{if set}]) \]
   - Database instance continue to auto-tunes both SGA_TARGET and PGA_AGGREGATE_TARGET.

4. If both SGA_TARGET as well as PGA_AGGREGATE_TARGET are not set then
   - SGA_TARGET will be initialized to
     \[ \text{Min} \left( \frac{60 \times \text{MEMORY\_TARGET}}{100}, \text{SGA\_MAX\_SIZE} \text{ (if set)} \right) \]
   - PGA_AGGREGATE_TARGET will be initialized to
     \[ \text{Max} \left( \frac{40 \times \text{MEMORY\_TARGET}}{100}, \text{MEMORY\_TARGET} - \text{SGA\_MAX\_SIZE} \text{ (if Set)} \right) \]

5. If MEMORY_MAX_TARGET is not set, then it will default to MEMORY_TARGET
**Scenario 2: When MEMORY_TARGET is not set or set to 0 [Zero-Default Value]**

1. If only SGA_TARGET is set but not the PGA_AGGREGATE_TARGET then
   - SGA components (Shared pool, Buffer Cache, Java pool, Large Pool, Stream Pool) are auto-tuned but SGA_TARGET itself is not increased or decreased automatically
   - PGA_AGGREGATE_TARGET is auto-tuned independent if whether it is explicitly set or not. Its initial value depends on the size of SGA_TARGET setting.

2. If SGA_TARGET is not set then
   - SGA components (Shared pool, Buffer Cache, Java pool, Large Pool, Stream Pool) are not auto-tuned their values are set explicitly as in Oracle 10g.

3. If SGA_MAX_SIZE is not set then it will default to MEMORY_MAX_TARGET

4. If MEMORY_MAX_TARGET is set then you can dynamically change the MEMORY_TARGET to a non-zero value, provided it does not exceed the value of MEMORY_MAX_TARGET

5. If MEMORY_MAX_TARGET is not set, then you cannot use Oracle 11g Automatic memory management.
MONITORING AND TUNING AUTOMATIC MEMORY MANAGEMENT

We can Monitor and Tune Oracle 11g AMM using
1. Oracle Enterprise Manager
2. Oracle Views with SQLPlus

MONITOR AND TUNE AMM USING EM DATABASE CONSOLE

Step 1 : Logon to Oracle 11g Enterprise Manager Database Console

Specify the Username and password and click LOGIN to Continue

** Login **

Login to Database: orcl

* User Name: sys
* Password: ********
Connect As: SYSDBA

Copyright © 1996, 2006, Oracle. All rights reserved.
Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation
Unauthorized access is strictly prohibited.
Step 2: Click on Server on the Home Page Tab as shown below in RED outline.

Oracle Enterprise Manager 11g
Database Control

Logged in As SYS

Database Instance: orcl

Latest Data Collected From Target Jul 30, 2007 4:51:36

General

Status: Up
Up Since: Jul 30, 2007 4:08:56 PM GMT
Instance Name: orcl
Version: 11.1.0.3.1
Host: gcr1
Listener: LISTENER_orcl

Health Meter

Issues: 9
Incidents: 3
Alerts: 6

Health History

Host CPU

Load: 1627389952.00
Step 3: Select **Memory Parameters** under Database Configuration. On the Memory Parameters page, you can **enable** or **disable** the Oracle 11g Automatic Memory Management as shown below in RED outline. You can also click on **ADVICE** button to get the advice.
Step 4: On the Memory Parameters Page, you can see the SGA details as shown below in RED

Database Instance: ocl >

**Memory Parameters**

<table>
<thead>
<tr>
<th>Automatic Memory Management Enabled</th>
<th>Disable</th>
<th>Total Memory Size (MB) 2048</th>
<th>Advice</th>
</tr>
</thead>
</table>

**Allocation History**

This chart shows the history of the components of the Memory.

- **PGA**
- **SGA**

**Maximum Memory Size**

The Maximum Memory Size specifies the maximum memory that the database may allocate. If you specify the Maximum Memory Size, you can later dynamically change the Total Memory Size above (provided Total Memory Size does not exceed the Maximum Memory Size).

Maximum Memory Size (MB) 2048

The database must be restarted before any changes to this value take effect.

**SGA**

The System Global Area (SGA) is a group of shared memory structures that contains data and control information for one Oracle database. The SGA is allocated in memory when an Oracle database instance is started.

**Allocation History**

This chart shows the history of the components of the SGA.

- **Shared Pool**
- **Buffer Cache**
- **Java Pool**
- **Large Pool**

**Current Allocation**

<table>
<thead>
<tr>
<th>Automatic Shared Memory Management Enabled</th>
<th>Total SGA Size (MB) 2048</th>
</tr>
</thead>
</table>

**SGA Component Current Allocation (MB)**

- **Shared Pool**: 432
- **Buffer Cache**: 1552
- **Large Pool**: 16
- **Java Pool**: 16
- **Other**: 32

76% 16% 1% 1% 1%
Step 5: On the Memory Parameters Page, you can see the PGA details as shown below in RED. To get more details about the PGA usage, click on PGC Memory Usage Details as shown in Red and you can get complete graphical representation of the usage.
MONITOR AND TUNE AMM USING ORACLE V$VIEW'S

The dynamic performance view V$MEMORY_DYNAMIC_COMPONENTS shows the current sizes of all dynamically tuned memory components, including the total sizes of the SGA and instance PGA.

SQL> select component, current_size, min_size, max_size  
from V$MEMORY_DYNAMIC_COMPONENTS;

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>CURRENT_SIZE</th>
<th>MIN_SIZE</th>
<th>MAX_SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>shared pool</td>
<td>452984832</td>
<td>452984832</td>
<td>452984832</td>
</tr>
<tr>
<td>large pool</td>
<td>16777216</td>
<td>0</td>
<td>16777216</td>
</tr>
<tr>
<td>java pool</td>
<td>16777216</td>
<td>16777216</td>
<td>16777216</td>
</tr>
<tr>
<td>streams pool</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SGA Target</td>
<td>2147483648</td>
<td>2147483648</td>
<td>2147483648</td>
</tr>
<tr>
<td>DEFAULT buffer cache</td>
<td>1627389952</td>
<td>1627389952</td>
<td>1627389952</td>
</tr>
<tr>
<td>KEEP buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RECYCLE buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEFAULT 2K buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEFAULT 4K buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEFAULT 8K buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEFAULT 16K buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEFAULT 32K buffer cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Shared IO Pool</td>
<td>67108864</td>
<td>67108864</td>
<td>67108864</td>
</tr>
<tr>
<td>PGA Target</td>
<td>436207616</td>
<td>436207616</td>
<td>436207616</td>
</tr>
<tr>
<td>ASM Buffer Cache</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

16 rows selected.
The view **V$MEMORY_RESIZE_OPS** has the circular history of last 800 SGA resize requests.

```
SQL> select component, parameter, initial_size, target_size, final_size, status, start_time, end_time
     from V$MEMORY_RESIZE_OPS;
```

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>PARAMETER</th>
<th>INITIAL_SIZE</th>
<th>TARGET_SIZE</th>
<th>FINAL_SIZE</th>
<th>STATUS</th>
<th>START_TIME</th>
<th>END_TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGA Target</td>
<td>sga_target</td>
<td>0</td>
<td>2147483648</td>
<td>2147483648</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT 4K buffer cache</td>
<td>db_4k_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>java pool</td>
<td>java_pool_size</td>
<td>0</td>
<td>16777216</td>
<td>16777216</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>ASM Buffer Cache</td>
<td>db_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT 2K buffer cache</td>
<td>db_2k_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT buffer cache</td>
<td>db_cache_size</td>
<td>1644167168</td>
<td>1627389952</td>
<td>1627389952</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT buffer cache</td>
<td>db_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT 8K buffer cache</td>
<td>db_8k_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>shared pool</td>
<td>shared_pool_size</td>
<td>0</td>
<td>452984832</td>
<td>452984832</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>streams pool</td>
<td>streams_pool_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>large pool</td>
<td>large_pool_size</td>
<td>0</td>
<td>16777216</td>
<td>16777216</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>large pool</td>
<td>large_pool_size</td>
<td>0</td>
<td>16777216</td>
<td>16777216</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT 16K buffer cache</td>
<td>db_16k_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>PGA Target</td>
<td>pga_aggregate_target</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>DEFAULT 32K buffer cache</td>
<td>db_32k_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>KEEP buffer cache</td>
<td>db_keep_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
<tr>
<td>RECYCLE buffer cache</td>
<td>db_recycle_cache_size</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>COMPLETE</td>
<td>30-JUL-07</td>
<td>30-JUL-07</td>
</tr>
</tbody>
</table>

The view **V$MEMORY_TARGET_ADVICE** provides tuning advice for the MEMORY_TARGET initialization parameter.