

Automatic Workload Management Planboard June 2010

Bernhard de Cock Buning

Who am I ?

- Bernhard de Cock Buning
- 4+ year working as Consultant/Member of GRID-IT.
- 12 year experience with Oracle Products.
- bdcbuning@grid-it.nl

- www.grid-it.nl / www.rachelp.nl
- Technical Reviewer of Pro Oracle Database 11g Rac. Release date 22 july 2010.(<http://apress.com/book/view/1430229586>)

Agenda

- What is Automatic Workload Management
- Automatic Workload Management Options
- Services
- Client connections
- Load Balancing Advisor
- Instance Caging
- Customer case
- Conclusion.

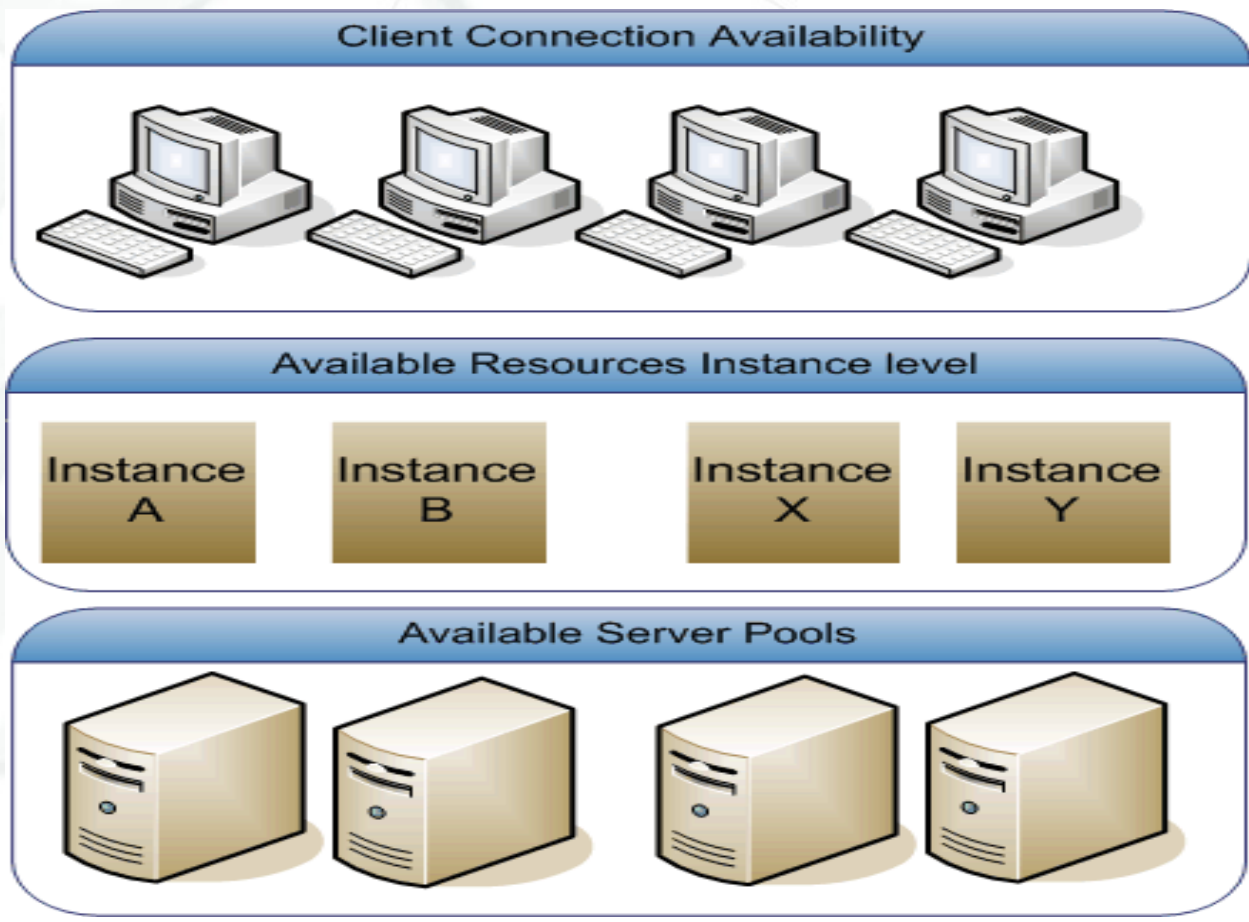
What is Automatic Workload Management

- Automatic Workload Management enables you to manage the distribution of workloads to provide optimal performance for users and applications.
- This includes providing the highest availability for database connections, rapid failure recovery, and balancing workloads optimally across the active configuration.

Automatic Workload Management

- With the introduction of Oracle 11g Release 2 (11.2.0.1) we have all kinds of automatic workload management options (both new and old) available in Real Application Cluster environments.
- Some of these options are also available in Single Instance and Data Guard environments with/or without RAC
- Oracle restart
- What are the available options ?
- What can you use, what not ?
- How to configure and manage the possibilities ?

Automatic Workload Management



Automatic Workload Management

Automatic Workload Management options:

- Client side connect time load balancing
- Client side connect time failover
- Run Time Connection failover (TAF)
- Fast Connection Failover
- Load Balancing Advisor
- Automatic Workload Repository
- Fast Application Notification
- Services
- High Availability Framework
- Single Client Access Name
- Serverpools
- EZ connect
- Database Resident Connection Pooling
- Instance Caging
- Resource Manager

Automatic Workload Management Services

- As of Oracle 10g, Oracle introduced their slogan 'everything will change to services'.
- A service is a logical name, to group similar type of users/type of work together.
- Management of a Cluster environment with Real Application Cluster database instances can be made most simplified when you use services.
- Many different attributes for the services.

The question will **NOT** be, will I use services, but which attributes do we define for our services.

Oracle Workload Management Services

- Oracle 11gR2 without services is like a car without a steering wheel.
- Creation of a service trough, Enterprise Manager, DBMS_SERVICE (deprecated), srvctl
- Admin managed Database / Policy Managed Databases

```
srvctl add service -d <db_unique_name> -s <service_name>
{-r "<preferred_list>" [-a "<available_list>"]}
[-P {BASIC | NONE | PRECONNECT}]
| -g <server_pool> [-c {UNIFORM | SINGLETON}] }
[-k <net_num>] [-l [PRIMARY][,PHYSICAL_STANDBY][,LOGICAL_STANDBY]
[,SNAPSHOT_STANDBY]] [-y {AUTOMATIC | MANUAL}] [-q {TRUE|FALSE}]
[-x {TRUE|FALSE}] [-j {SHORT|LONG}] [-B {NONE|SERVICE_TIME|THROUGHPUT}]
[-e {NONE|SESSION|SELECT}] [-m {NONE|BASIC}] [-z <failover_retries>]
[-w <failover_delay>]
```

Oracle Workload Management Services

Cluster Database: RAC

- [Home](#)
- [Performance](#)
- [Availability](#)
- [Server](#)
- [Schema](#)
- [Data Movement](#)
- [Software and Support](#)
- [Topology](#)

[High Availability Console](#)

Backup/Recovery

Setup

- [Backup Settings](#)
- [Recovery Settings](#)
- [Recovery Catalog Settings](#)

Manage

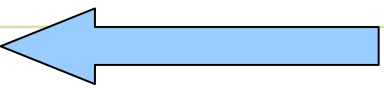
- [Schedule Backup](#)
- [Manage Current Backups](#)
- [Backup Reports](#)
- [Manage Restore Points](#)
- [Perform Recovery](#)
- [View and Manage Transactions](#)

Oracle Secure Backup

- [Assign and Manage](#)

Services

[Cluster Managed Database Services](#)



Automatic Workload Management

Client side connect time load balancing

- The idea is to eliminate login storms.
- Sessions are created on a round robin basis.
- Example without SCAN:

```
sales =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST=server1-vip) (PORT = 1521))  
    (ADDRESS = (PROTOCOL = TCP) (HOST=server2-vip) (PORT = 1521))  
    (LOAD_BALANCE = yes)  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = prod)))
```

Automatic Workload Management

Client side connect time load balancing

- When using SCAN, you must have more than 1 IP address defined.
- Oracle recommends to define 3 ip addresses in DNS.
- Example with SCAN:

```
sales =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = TCP) (HOST=cluster1-scan) (PORT = 1521))  
  (CONNECT_DATA =  
    (SERVER = DEDICATED)  
    (SERVICE_NAME = prod)))
```

Automatic Workload Management

Client side connect time failover

- Enable failover at the time a session tries to connect to a instance when it may fail unexpectedly.
- If the first connection attempt fails, then the client will automatically try to connect to the next server in the list.
- Example without SCAN:

```
sales =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST=server1-vip) (PORT = 1521))  
    (ADDRESS = (PROTOCOL = TCP) (HOST=server2-vip) (PORT = 1521))  
    (FAILOVER = yes)  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = prod)))
```

Automatic Workload Management

Client side connect time failover

- Using Failover with SCAN is something you get when using SCAN in all cases.
- SCAN gives the list of IP and when the first is not available, the second one is validate.
- Address list can be useful in Data Guard environments to point to another scan name.
- Example with SCAN:

```
sales =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = TCP) (HOST=cluster1-scan) (PORT = 1521))  
  (ADDRESS = (PROTOCOL = TCP) (HOST=cluster2-drscan) (PORT = 1521))  
  (CONNECT_DATA =  
    (SERVER = DEDICATED)  
    (SERVICE_NAME = prod)))
```

Automatic Workload Management EZconnect

- The EZconnect method was already available for some Oracle releases but until now it was not very useful in RAC environments.
- Starting with Oracle 11gR2 Ezconnect will make it possible to create a session without needing a client tnsnames.ora configuration file, or defining the total connect string.
- 11g Release 2 default is NAMES.DIRECTORY_PATH=(tnsnames, ldap, ezconnect), modify the sqlnet.ora file to modify the order of appearance.

- An example:

```
[oracle@server1 ~]$ sqlplus system@"cluster1-scan/rac\  
[oracle@server1 ~]$ sqlplus sys@"cluster1-scan/rac\  
          as  
          sysdba
```

Automatic Workload Management

Database Resident Connection Pooling

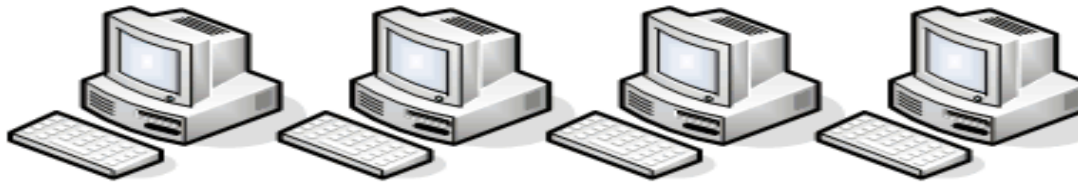
- Pooled server model closely follows the “dedicated” model.
- Removes the overhead of dedicating a server session to every connection. (SYS_DEFAULT_CONNECTION_POOL)
- For PHP/OCI applications.
- Connection Pooling like middle-tier but now in the database.
- An example:

```
SQL> execute dbms_connection_pool.start_pool;
```

```
$ sqlplus sys@\"cluster1-scan/rac:pooled\" as sysdba
```

```
SQL> execute dbms_connection_pool.stop_pool;
```

Client Connection Availability



Available Resources Instance level

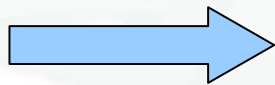
Instance
A

Instance
B

Instance
X

Instance
Y

Available Server Pools



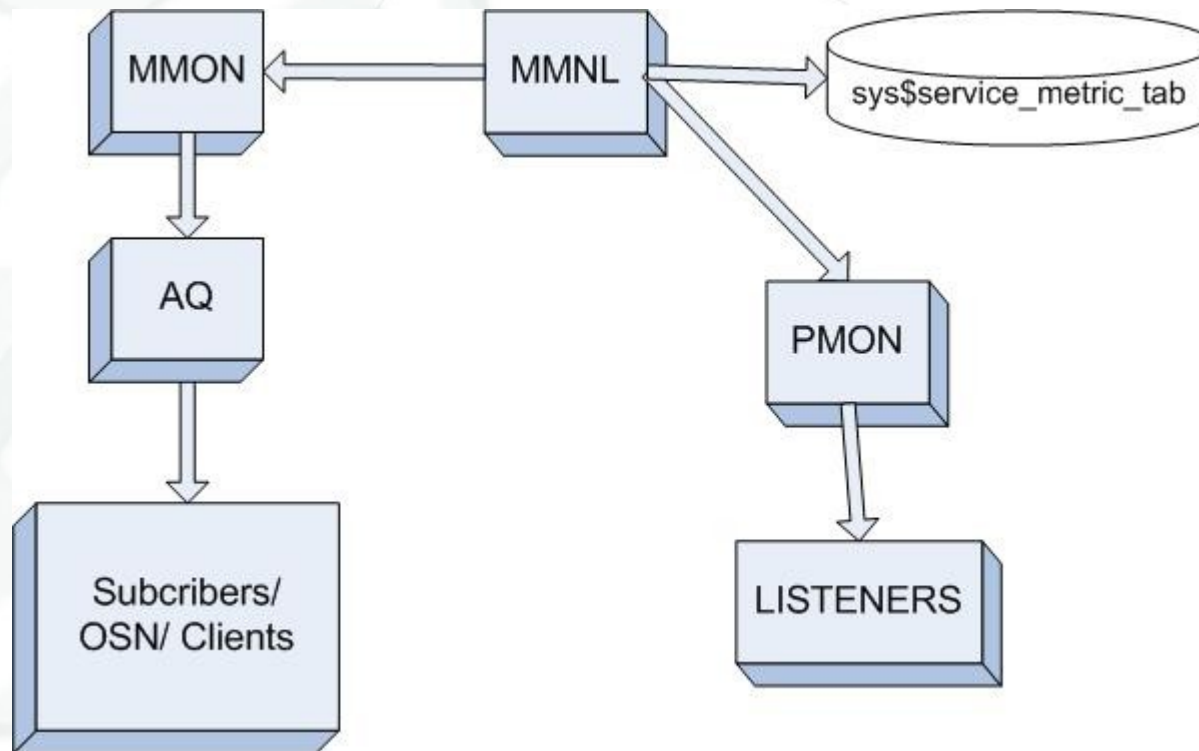
Automatic Workload Management Load Balancing Advisor

- The load balancing advisor is Introduced in Oracle 10g.
- Before Oracle 10g Release 2 only based upon the number of sessions. Now it is possible to make use of the load balancing advisor, which will collect information for each instance and forward the result to each listener on the host.
- The listener will redirect incoming connections to other instances if needed based on the information received from the load balancing advisor.
- FAN is used to inform connection pools/Listeners

Automatic Workload Management Load Balancing Advisor

- Connection Load Balancing: CLB_GOAL
 - Short – uses Load Balancing Advisory, when Load Balancing Advisory is enabled with Service Goals.
 - Long – Balances the number of connections per instance using session count per service.
- Service Goals:
 - None
 - Service Time - Best response time – elapsed time for work done in the service plus available bandwidth.
 - Throughput - better throughput – rate that work is completed plus available bandwidth.

Automatic Workload Management Load Balancing Advisor



Automatic Workload Management

Instance Caging

- Method of managing CPU allocation .
- Useful in consolidated server environment.
- Multiple instance on a single server “fight” for CPU resources
- Requires Resource Manager and `cpu_count` parameter.

- Over-Provisioning: For Non-critical database instance
- Partitioning: For production systems
- Not a solution to save license costs.

Automatic Workload Management Instance Caging

- One off needed on top of 10.2.0.4 or apply 10.2.0.5 patchset.
- 6309691 Dynamic changes to CPU_COUNT limited to 3 * initial value
- 6874858 Poor performance with Resource Manager when RMAN is running
- 7117117 Resource manager does not scale wtr to cpu_count
- 7510766 Resource Manager may over throttle
- 7504003 STPRD: Session waiting on 'RESMGR: CPU quantum'
- 5879698 Performance degradation when using PX with resource plans
- 7607661 DRBM Diagnostic dump needs to include debugger stack trace

Automatic Workload Management Instance Caging

- One off needed on top of 11gR1
- 7117117 Resource manager does not scale wtr to cpu_count
- 7510766 Resource Manager may over throttle
- 7504003 STPRD: Session waiting on 'RESMGR: CPU quantum'
- 7607661 DRBM Diagnostic dump needs to include debugger stack trace

Automatic Workload Management Instance Caging

- CPU_COUNT modification impact:
 - # of DBWR processes
 - # of LMS processes for RAC
 - # LGWR processes
 - Parallel Recovery
 - utlrp.sql
- DBA_CPU_USAGE_STATISTICS display modification in CPU. Will be collected after `dbms_stats.gather_database_stats`

Automatic Workload Management Instance Caging

- Enable resource manager
- Default several plans are available.
- Set resource_manager_plan parameter.
- Make sure MGMT_P# / CPU_# is defined.

```
SQL> show parameter resource_manager_plan
```

NAME	TYPE	VALUE
resource_manager_plan	string	MIXED_WORKLOAD_PLAN

Automatic Workload Management Instance Caging

- Monitor the Instance Caging.
- Use EMGC performance page.
- Query the Resource Management views.

```
SQL> select name from gv$rsrc_plan where is_top_plan = 'TRUE'  
and cpu_managed = 'ON';
```

```
SQL> select username,RESOURCE_CONSUMER_GROUP from gv$session  
where username is not null;
```

```
SQL> select name, consumed_cpu_time, cpu_wait_time from  
gv$rsrc_consumer_group;
```

Automatic Workload Management Customer Case

- Real Application Cluster 11gR2 on Linux
- Clients 10g, 11gR1.
- Workload OLTP and Batch.

Requirement of the customer:

- Connections need to failover in case of an failure.
- Connections need to go to a server where it can be dealt with the best.
- File handling on one server
- Import on one server
- Batches on one server

Automatic Workload Management

Automatic Workload Management options:

- Client side connect time load balancing
- Client side connect time failover
- Run Time Connection failover (TAF)
- Fast Connection Failover
- Load Balancing Advisor
- Automatic Workload Repository
- Fast Application Notification
- Services
- High Availability Framework
- Single Client Access Name
- Serverpools
- EZ connect
- Database Resident Connection Pooling
- Instance Caging
- Resource Manager

Automatic Workload Management

Conclusion

- Lots of available options.
- Always make use of services even when not using RAC.
- Create services and no client configuration files are required when using EZconnect
- Several connection options cannot be used when not using RAC/Data Guard.
- You need 11gR2 client to make full use of all the options.
- Test for each client version if the desired configuration can be used.
- Combine the options based on business needs and proof the “result”
- Memory/CPU resource allocation more each to control for resource management in compare to I/O.

Questions?

