



Configuring an Application as a Clustered Resource

By Bill Pritchett, Senior Consultant, Advizex Technologies - Managed Services

Growing infrastructure costs and reduced budgets have led at times to a need for creative infrastructure design. This need is seen most often in the need for a High Availability solution within a restrictive budget. One approach to help reduce this cost is to purchase Database licenses for one server, then take advantage of getting Clusterware and ASM for free with those licenses to cluster 2 servers together. This allows you to provide a High Availability solution with half the costs both now and in the future. Let's take a look at the costs based on Oracle's License price sheet.

Example: 2 servers with Dual 8 core Intel processors

***Note: Oracle calculates Licenses at .5 per core for Intel Processors*

Oracle Pricing:

| License Type | Named User Plus | NUP Support | Processor | Proc Support |
|------------------------------------|-----------------|-------------|-----------|--------------|
| Enterprise Edition | 950 | 209.00 | 47,500 | 10,450.00 |
| Real Application Clusters | 460 | 101.20 | 23,000 | 5,060.00 |
| Real Application Clusters One Node | 200 | 44.00 | 10,000 | 2,200.00 |

Retail Cost for a Full RAC installation would be:

2(servers) X 4(Enterprise Edition) = 380,000
 2(servers X 4(Real Application Cluster) = 184,000
 Future support per year(8 EE + 8 RAC) = 124,080
 TOTAL for 2 years = 688,080

Retail Cost for a cluster with a single instance would be:

1(servers) X 4(Enterprise Edition) = 190,000
 Future support per year(4 EE) = 41,800
 TOTAL for 2 years = 231,800

2 year total savings of \$ 456,280

3 year total savings of \$ 538,560

Action Script

```
#!/bin/bash
#####
# Export Variables
#####
export ORACLE_HOME=/opt/app/oracle/product/11.2.0.3
export ORACLE_SID=wmssit
#####
# Case statement that handles input variable and based
# on variable performs either a startup, immediate shutdown,
# shutdown abort, or checks the database status
#####

case $1 in
'start')
$ORACLE_HOME/bin/sqlplus /nolog <<EOF
conn / as sysdba
startup
exit
EOF

RET=0
;;
'stop')
$ORACLE_HOME/bin/sqlplus /nolog <<EOF
conn / as sysdba
shutdown immediate
exit
EOF

RET=0
;;
'clean')

$ORACLE_HOME/bin/sqlplus /nolog <<EOF
conn / as sysdba
shutdown abort
exit
EOF

RET=0
;;
'check')
ok=`ps -ef | grep smon | grep $ORACLE_SID | grep -v grep | wc -l`
if [ $ok = 0 ]; then
    RET=1
else
    RET=0
fi
;;
*)
RET=0
;;
esac

#####
# Check return code
#####

if [ $RET -eq 0 ]; then
    exit 0
else
    exit 1
fi
```

Resource file

```
NAME=ora.mytest.db
TYPE=cluster_resource
DESCRIPTION=Oracle Database resource
ACL=owner:oracle:rwx,pgrp:oinstall:rwx,other::r--,group:dba:r-x,user:grid:r-x
ACTION_SCRIPT=/u01/app/grid/11.2.0.4/crs/HA_scripts/mytest_active_passive_cluster.sh
ACTIVE_PLACEMENT=0
AUTO_START=never
CARDINALITY=1
CHECK_INTERVAL=10
DEGREE=1
ENABLED=1
HOSTING_MEMBERS=oel6p5-rac1 oel6p5-rac2
LOGGING_LEVEL=1
START_DEPENDENCIES=hard(ora.DATA.dg,ora.ARCH.dg)
weak(type:ora.listener.type,uniform:ora.ons)
START_TIMEOUT=600
STOP_DEPENDENCIES=hard(intermediate:ora.asm,shutdown:ora.DATA.dg,shutdown:ora.AR
CH.dg,ora.net1.network)
STOP_TIMEOUT=600
UPTIME_THRESHOLD=1h
```

TNSNAMES.ORA Examples:

***NOTE: These files should contain the local listener information based on the host it resides on*

Example client TNSENTRY

```
MYSTEST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = oramystest)(PORT = 1610))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = mytest)
    )
  )
```

oramystest is the DNS CName entry that points to the scan address in DNS

NODE 1

```
oracle@oel6p5-rac1:/u01/app/oracle/product/11.2.0.4/network/admin
File Edit View Search Terminal Help
[oracle@oel6p5-rac1 admin]$ ls
samples shrept.lst tnsnames.ora
[oracle@oel6p5-rac1 admin]$ cat tnsnames.ora
# tnsnames.ora Network Configuration File: /u01/app/oracle/product/11.2.0.4/netw
ork/admin/tnsnames.ora
# Generated by Oracle configuration tools.

MYTEST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = oel6p5-rac1.localdomain)(PORT = 9011))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = mytest)
    )
  )
)

LISTENER_MYTEST =
  (ADDRESS = (PROTOCOL = TCP)(HOST = oel6p5-rac1.localdomain)(PORT = 9011))
```

NODE 2

```
[oracle@oel6p5-rac2 admin]$ cat tnsnames.ora
# tnsnames.ora Network Configuration File: /u01/app/oracle/product/11.2.0.4/netw
ork/admin/tnsnames.ora
# Generated by Oracle configuration tools.

MYTEST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = oel6p5-rac2.localdomain)(PORT = 9011))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = mytest)
    )
  )
)

LISTENER_MYTEST =
  (ADDRESS = (PROTOCOL = TCP)(HOST = oel6p5-rac2.localdomain)(PORT = 9011))
```

How to Modify Resource attributes and key attribute settings

Note: Below is an example of how to modify the resource attributes dynamically

As **ORACLE**

```
$GRID_HOME/bin/crsctl modify res ora.mytest.db -attr "STOP_DEPENDENCIES=(settings)"
```

You can also check the Clusterware Admin and Deployment Guide in Appendix G for more details

Example: (After adding a diskgroup named ACFS)

```
$GRID_HOME/bin/crsctl modify res ora.mytest.db -attr  
"STOP_DEPENDENCIES='hard(intermediate:ora.asm,shutdown:ora.DATA.dg,shutdown:ora.ARCH.dg,s  
hutdown:ora.ACFS.dg,ora.net1.network)'"
```

```
$GRID_HOME/bin/crsctl modify res ora.wmssit.db -attr  
"START_DEPENDENCIES='hard(ora.DATA.dg,ora.ARCH.dg,ora.ACFS.dg)  
weak(type:ora.listener.type,uniform:ora.ons)'"
```

```
START_DEPENDENCIES=hard(ora.DATA.dg,ora.ARCH.dg) weak(type:ora.listener.type,uni  
form:ora.ons)  
START_TIMEOUT=600  
STATE_CHANGE_TEMPLATE=  
STOP_DEPENDENCIES=hard(intermediate:ora.asm,shutdown:ora.DATA.dg,shutdown:ora.AR  
CH.dg,ora.net1.network)  
STOP_TIMEOUT=600  
UPTIME_THRESHOLD=1h  
  
[oracle@oel6p5-rac1 HA_scripts]$ $GRID_HOME/bin/crsctl modify resource ora.mytes  
t.db -attr "STOP_DEPENDENCIES='hard(intermediate:ora.asm,shutdown:ora.DATA.dg,sh  
utdown:ora.ARCH.dg,shutdown:ora.ACFS.dg,ora.net1.network)'"  
[oracle@oel6p5-rac1 HA_scripts]$ $GRID_HOME/bin/crsctl modify resource ora.mytes  
t.db -attr "START_DEPENDENCIES='hard(ora.DATA.dg,ora.ARCH.dg,ora.ACFS.dg) weak(t  
ype:ora.listener.type,uniform:ora.ons)'"  
[oracle@oel6p5-rac1 HA_scripts]$ $GRID_HOME/bin/crsctl status resource ora.mytes  
t.db -p | grep DEPEND  
START_DEPENDENCIES=hard(ora.DATA.dg,ora.ARCH.dg,ora.ACFS.dg) weak(type:ora.liste  
ner.type,uniform:ora.ons)  
STOP_DEPENDENCIES=hard(intermediate:ora.asm,shutdown:ora.DATA.dg,shutdown:ora.AR  
CH.dg,shutdown:ora.ACFS.dg,ora.net1.network)  
[oracle@oel6p5-rac1 HA_scripts]$ █
```

Key Attributes

AUTO_START: Determines how the resource is started after a system reboot. The following settings are valid

- **always:** Restarts the resource when the server restarts regardless of the state of the resource when the server stopped.
- **restore:** Restores the resource to the same state that it was in when the server stopped. Oracle Clusterware attempts to restart the resource if the value of TARGET was ONLINE before the server stopped.
- **never:** Oracle Clusterware never restarts the resource regardless of the state of the resource when the server stopped.

****Note:** An AUTO_START setting of never can be overridden if the START_DEPENDENCIES attribute contains any pullup dependencies.

If you only license one server of the cluster then you should set AUTO_START to never so that you can bring the database up on node one with the following command

```
$GRID_HOME/bin/crsctl start resource ora.mytest.db -n oel6p5-rac1
```

```
[oracle@oel6p5-rac1 HA_scripts]$ $GRID_HOME/bin/crsctl stop resource ora.mytest.db
CRS-2673: Attempting to stop 'ora.mytest.db' on 'oel6p5-rac2'
CRS-2677: Stop of 'ora.mytest.db' on 'oel6p5-rac2' succeeded
[oracle@oel6p5-rac1 HA_scripts]$ $GRID_HOME/bin/crsctl start resource ora.mytest.db -n oel6p5-rac1
CRS-2672: Attempting to start 'ora.mytest.db' on 'oel6p5-rac1'
CRS-2676: Start of 'ora.mytest.db' on 'oel6p5-rac1' succeeded
[oracle@oel6p5-rac1 HA_scripts]$ █
```

START_DEPENDENCIES: This attribute is used to set to enforce resource dependencies so that the resource only starts once all the dependencies are met. The following are types of dependencies are valid

- **hard**(**[intermediate:]****[global:]**{resource_name | type:resource_type}): Specify a hard start dependency for a resource when you want the resource to start only when a particular resource or resource of a particular type starts.
- **weak**(**[concurrent:]****[global:]****[uniform:]**{resource_name | type:resource_type}): Specify a weak start dependency for a resource when you want that resource to start despite whether named resources are running, or not. An attempt to start this resource also attempts to start any resources on which this resource depends if they are not running.
- **attraction**(**[intermediate:]**{resource_name | type:resource_type}): Use the attraction start dependency when you want this resource to run on the same server with a particular named resource or any resource of a particular type.
- **pullup**:**always**(**[intermediate:]****[global:]**{resource_name | type:resource_type}): When you specify the pullup start dependency for a resource, then this resource starts as a result of named resources starting.

Note:

Oracle recommends that resources with hard start dependencies also have pullup start dependencies.

If you specify the pullup dependency on a resource type for a resource, then, when any resource of that particular type starts, Oracle Clusterware can start this resource. This causes the AUTO_START setting of NEVER to be overridden

STOP_DEPENDENCIES: This attribute is used to enforce resource dependencies for shutting down the resource. The only type of dependency available is [hard](#)

- [hard\(\[intermediate:\]\[global:\]\[shutdown:\]{resource_name | type:resource_type}\)](#): Specify a hard stop dependency for a resource that you want to stop when named resources or resources of a particular resource type stop.

Note:

If you want the database to failover to the passive node when there are issues with the Public Network then you have to specify the network as a dependency as seen in my example file

```
STOP_DEPENDENCIES=hard(intermediate:ora.asm,shutdown:ora.DATA.dg,shutdown:ora.ARCH.dg,ora.net1.network)
```

Documents used for references

Cluster Admin Guide

[11gR2 Clusterware Administration and Deployment Guide](#)