

Right Availability in RAC environment

Playing with Oracle clusterware infrastructure components



SPREKER : Jos van den Oord
E-MAIL : Jos.van.den.Oord@transfer-solutions.com
DATUM : 29 juni 2011



TRANSFER

Who am I

■ Jos van den Oord

- Working with Oracle products since 1998
- Working at Transfer-Solutions since 2010

■ Interests

■ Databases

- Right Availability
- Security, Performance

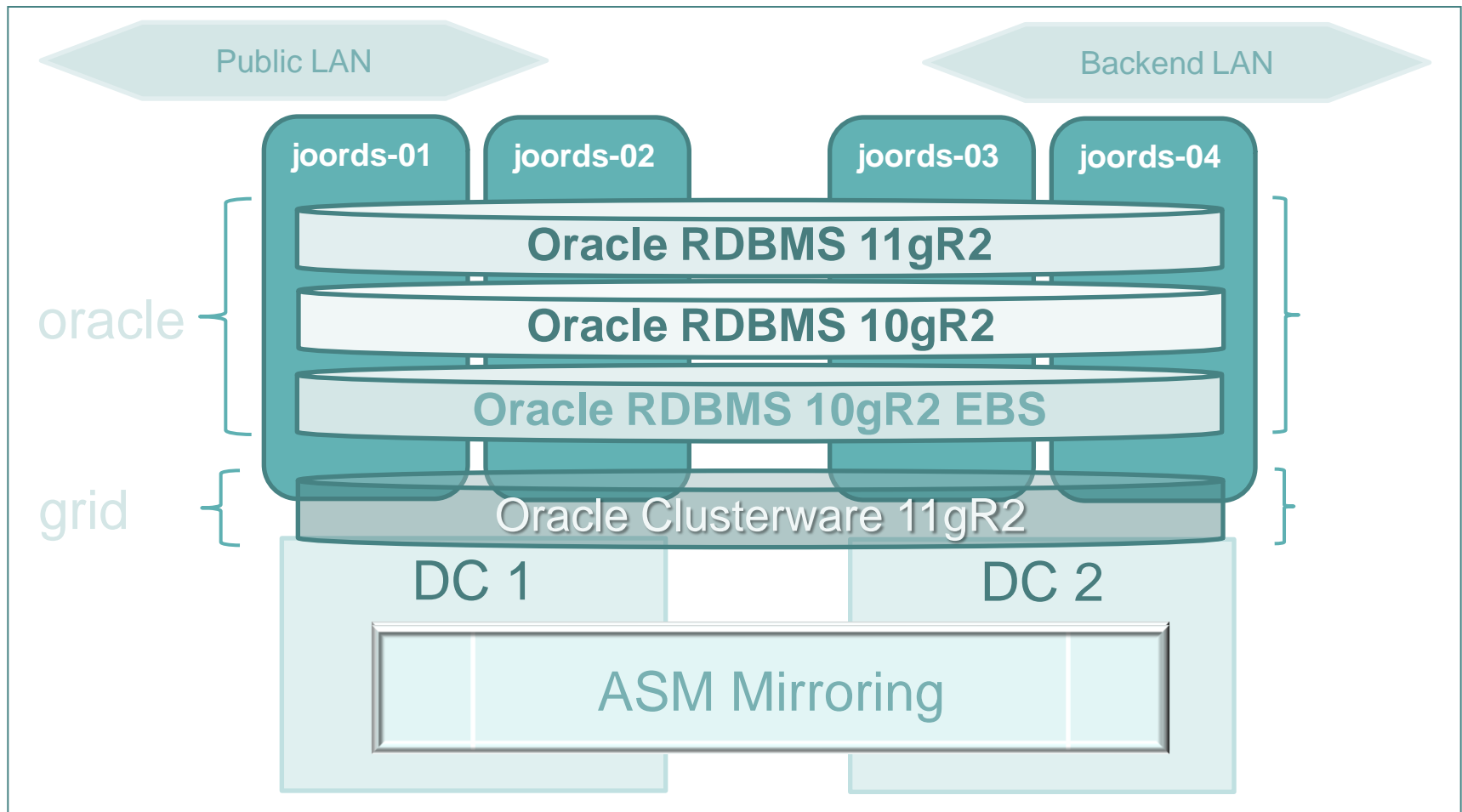
■ Blog : joordsblog.vandenoord.eu

■ Email : jos.van.den.oord@transfer-solutions.com

Agenda

- The RAC environment
- Right Availability for an application concept
- Configuration HP Data Protector on cluster
- Create Right Availability application on cluster
- Scheduling methods on RAC
- SCAN listener configurations
- Votingdisk problem on ASM
- ASM spfile - ASM mirroring

The RAC environment



High Availability for an application concept

- Point of interest
 - Objectives
 - Framework clusterware architecture
 - What is needed
 - Clusterware options
 - Clusterware resources
 - Action Script
 - Clusterware framework
 - Summary

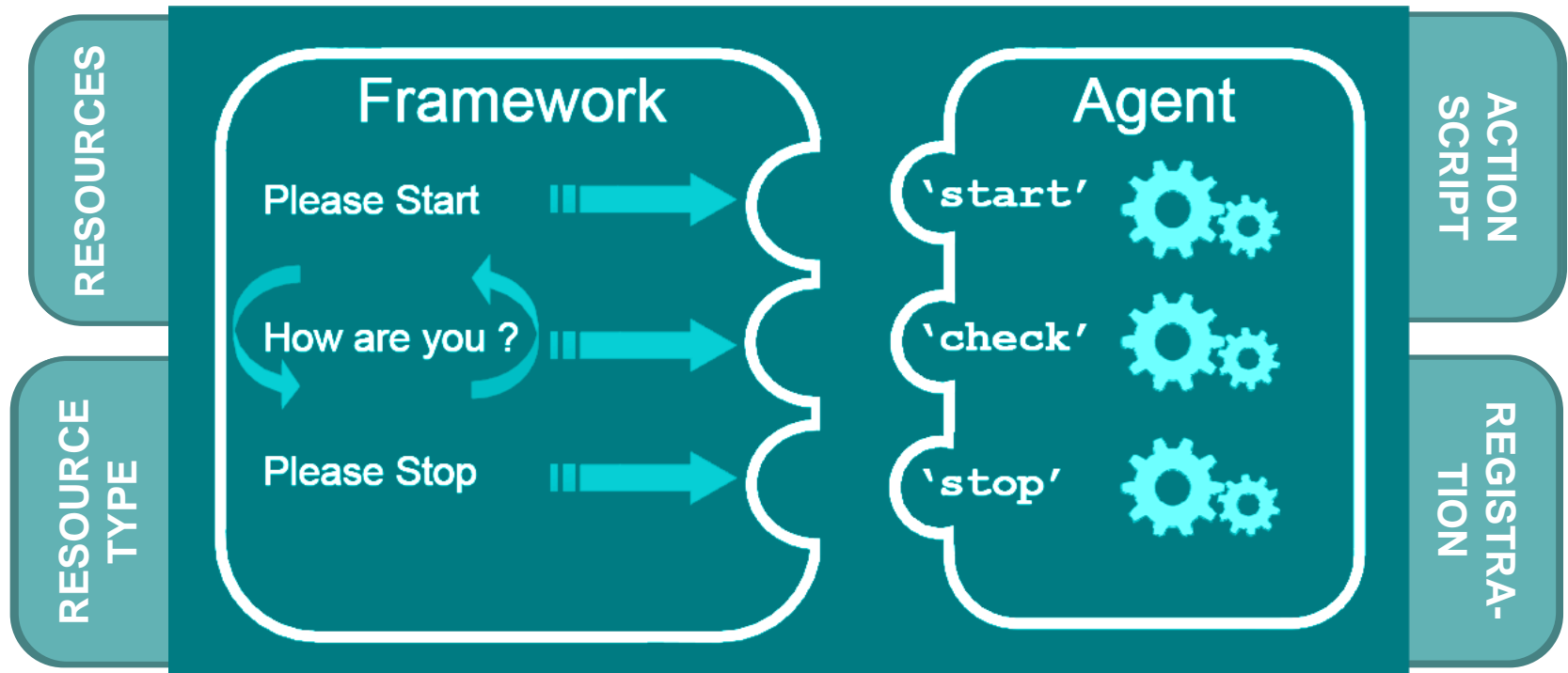
High Availability for an application concept

■ Objectives

- Restart on application failure
- Relocate on node failure
- Uses Oracle Clusterware
- Brings the same level of protection for Application components as Oracle's database components benefits from.

High Availability for an application concept

■ Framework clusterware architecture



High Availability for an application concept

■ What is needed

■ Network Location

- Clients need a node independent way of connecting to an Application

■ Configuration Files

- Applications typically need configuration files stored on shared file system

■ Dependencies

- Applications depend on these configuration files.
- They also depend on the location independent network.
- Or they can depend form other application resources.

High Availability for an application concept

■ Clusterware options

■ Application HA Framework

- Extends Oracle Clusterware Right Availability protection to your application

■ Oracle HA API

- Interface to allow you to change, at run time, how Oracle Clusterware manages your application

■ VIP resource

- Provides Application VIP's

■ ACFS or OCFS

- Provides a common location for files visible across all nodes

High Availability for an application concept

■ Clusterware resources

- A resource is an Oracle Clusterware managed application
- Profile Attributes for the resource are stored in the Oracle Cluster Registry
- Oracle RAC Database is made up from a series of resources
- Other applications can be treated as resources
- The API (C code – A control language)
 - Register, unregister, query, start, stop resources
 - Control their behavior from a program

High Availability for an application concept

■ Action Script

- Must be able to accept 5 parameter values called **Entry Points**: start, stop, check, clean and abort

```
case $1 in
'start')
    # start the application and check status
'stop')
    # stop the application
'clean')
    # kill application and cleanup
'check')
    # validate the status of the application
esac
```

- An Control Agent resource

High Availability for an application concept

- Clusterware framework
 - Contains the application modelling logic
 - Invokes the application aware agents, performs resource recovery
 - When a node goes down, the framework recovers the application by relocating it to a live node
 - Hosts in the OCR cache

High Availability for an application concept

■ Possible Resource States

State	Description	Action
ONLINE	The resource is running	-
OFFLINE	The resource is NOT running	-
UNKNOWN	An attempt to stop the resource has failed. Oracle Clusterware does not actively monitor resources that are in this state.	<code>crsctl stop resource command</code> <code>crsctl start resource command</code>
INTERMEDIATE	1: Oracle Clusterware cannot determine the state 2: Resource is not functional (Database [no]mount status)	<code>crsctl stop resource command</code> <code>crsctl start resource command</code>

High Availability for an application concept

■ Summary

■ Oracle Clusterware

- Brings Protection to 3rd Party Applications
- Restart / Relocate

■ Virtual IP's

- Used by clients to locate application irrespective of location
- Relocate with the Application as a 'package'

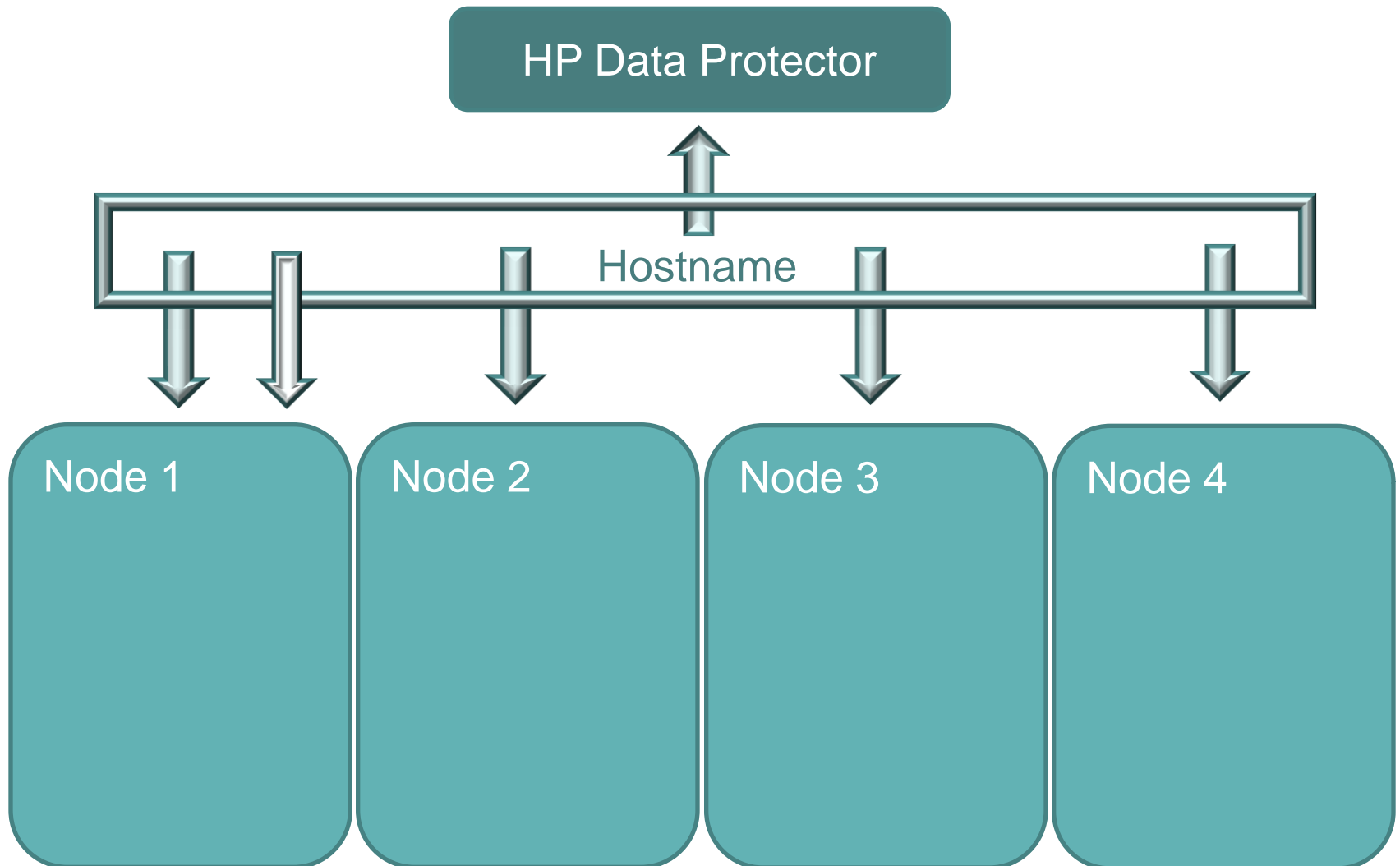
■ Protection

- Brings the same level of protection Oracle provides to it's own processes to 3rd Party Applications

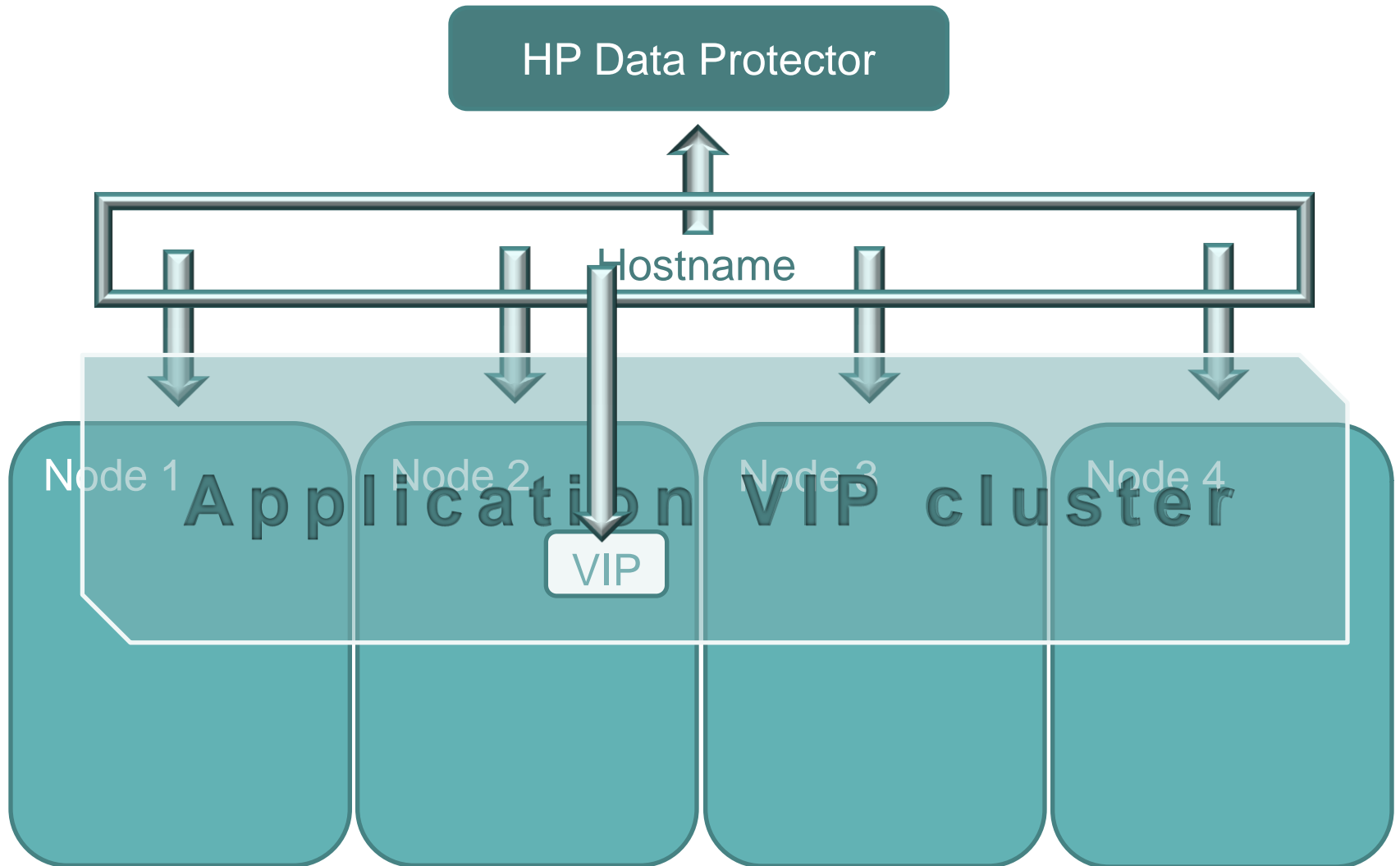
Configuration HP Data Protector

- Point of interest
 - Application virtual IP for the Oracle cluster
 - Using the backend network configuration
 - Using DB Services
 - Central Oracle password file for DB's

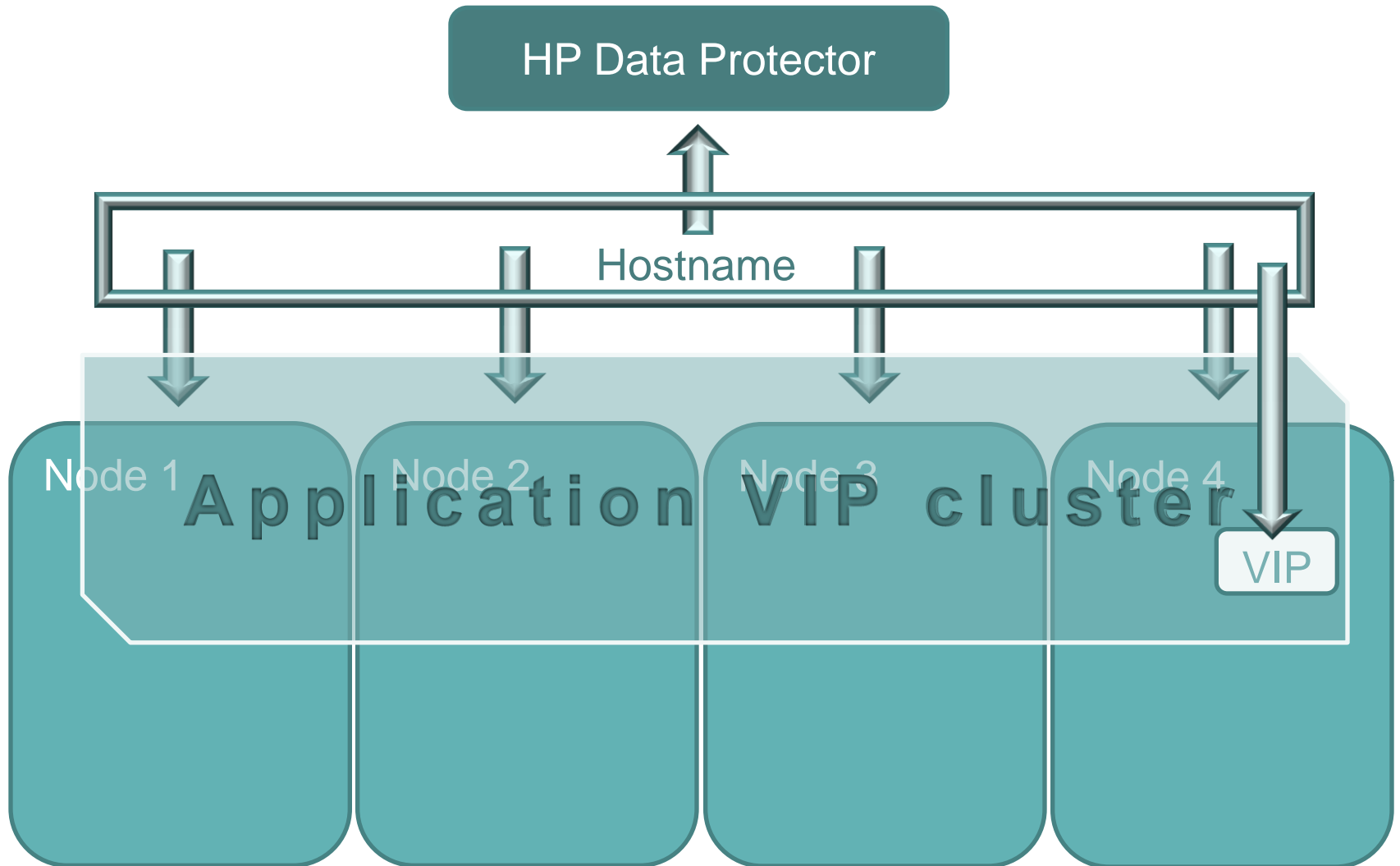
Configuration HP Data Protector



Configuration HP Data Protector



Configuration HP Data Protector



Configuration HP Data Protector

- VIP resources existed since Oracle Database 10gR1
 - Only used to failover the VIP to another node so that a client got an 'instant' NAK returned when it tried to connect to the Virtual IP. They still exist and operate in the same way in higher Releases.

Configuration HP Data Protector

■ Application VIPs

- New resource in Oracle Database since 10gR2
- Created as functional VIPs which can be used to connect to an application regardless of the node it is running on
- VIP is a dependent resource of the user registered application
- There can be many VIPs, only one VIP is allowed per User Application

Configuration HP Data Protector

■ Validation Cluster Network

- `crsctl stat res -p |grep -ie .network -ie subnet |grep -ie name -ie subnet`

■ Add application VIP on Backend Network

- `srvctl add vip -n joords_01 -k 2 -A
172.30.4.80/255.255.252.0/bond0`

■ Create joords_vip_bk in OCR

- `appvipcfg create -network=2 -ip=172.30.4.80 -
vipname=joords_vip_bk -user=root`

■ Validation Cluster Registration

- `crsctl stat res joords_vip_bk -p`

Configuration HP Data Protector

■ Validation Cluster Network

```
[root@joords1 ~]# crsctl stat res joords_vip_bk -p
NAME=joords_vip_bk
TYPE=app.appvip.type
ACL=owner:root:rwx,pgrp:root:r-x,other::r--,user:root:r-x
ACTIVE_PLACEMENT=0
AGENT_FILENAME=%CRS_HOME%/bin/orarootagent%CRS_EXE_SUFFIX%
AUTO_START=restore
CARDINALITY=1
CHECK_INTERVAL=1
DEGREE=1
DESCRIPTION=hp_dataprotect HA Application VIP
RESTART_ATTEMPTS=0
SCRIPT_TIMEOUT=60
SERVER_POOLS=*
START_DEPENDENCIES=hard(ora.net2.network) pullup(ora.net2.network)
STOP_DEPENDENCIES=hard(ora.net2.network)
USR_ORA_VIP=172.30.4.80
VERSION=11.2.0.1.0
```

Configuration HP Data Protector

■ Grant to grid os user

- `crsctl setperm resource joords_vip_bk -u user:grid:r-x`

■ Validate as GRID user

- `crsctl status resource joords_vip_bk -t`
- `ping -c4 joords_vip_bk`
- `crsctl relocate resource joords_vip_bk [-n node 1,2,3,4]`
- `ping -c4 joords_vip_bk`
- ssh connection test joords_vip_bk
- `ifconfig -a`

Configuration HP Data Protector

- Working with Services
 - Register DB services with `srvctl`
- Create RMAN user with `sysdba`
 - Connection error with RMAN as `sysdba` using Oracle service
 - Using symbolic links to a central password file, Problem solved.
 - Grant `sysdba` to RMAN, execute only once

Create HA application on Clusterware

- Example “Apache Tomcat”
 - The action script
 - The action script has dependencies with the VIP
 - The Cluster registration statement
 - Managing Apache Tomcat Application

Create HA application on Clusterware

■ Example “Apache Tomcat”

```
case $1 in
'start')
    $APACHE_TOMCAT_HOME/bin/startup.sh ; RET=$?
    ;;
'stop')
    $APACHE_TOMCAT_HOME/bin/shutdown.sh ; RET=$?
    ;;
'clean')
    $APACHE_TOMCAT_HOME/bin/shutdown.sh ; RET=$?
    ;;
'check')
    # download a simple, small image from the http server
    $APACHE_TOMCAT/usr/bin/wget -q --delete-after $CHECKURL
; RET=$?
...

```

Create HA application on Clusterware

- Dependencies and cluster registration
 - Depending on the resource of an Application VIP
 - Other resource dependencies.
 - Registration by configuration file or command line

Create HA application on Clusterware

- Dependencies and cluster registration
 - Using a configuration file

```
[root@joords-01 ]# $GRID_HOME/bin/crsctl add resource tomcat  
/ -type cluster_resource /  
-file /mnt/acfs/oracle/grid/tomcat.config
```

File : tomcat.config

```
ACTION_SCRIPT=/mnt/acfs/oracle/grid/tomcat.scr  
PLACEMENT=restricted  
RESTART_ATTEMP=2  
HOSTING_MEMBERS=joords-01 joords-02 joords-03 joords-04  
CHECK_INTERVAL=30  
START_DEPENDENCIES='hard(httpd-vip), pullup(httpd-vip)'  
STOP_DEPENDENCIES='hard(httpd-vip)'  
DESCRIPTION="Apache Tomcat HA Application"
```

Create HA application on Clusterware

- Dependencies and cluster registration
 - The command line

```
[root@joords-01 ] # $GRID_HOME/bin/crsctl add resource
tomcat /
/ -type cluster_resource /
-attr "ACTION_SCRIPT= /mnt/acfs/oracle/grid/tomcat.scr,
CHECK_INTERVAL=30, START_DEPENDENCIES='hard(httpd-vip)
pullup(httpd-vip)', STOP_DEPENDENCIES='hard(http-vip) ',
HOSTING_MEMBERS= joords-01 joords-02 joords-03 joords-04"
```

Create HA application on Clusterware

- Dependencies and cluster registration
 - crsctl status resource tomcat -p

```
NAME=tomcat
TYPE=cluster_resource
...
ACTION_SCRIPT= /mnt/acfs/oracle/grid/tomcat.scr
AGENT_FILENAME=%CRS_HOME%/bin/scriptagent
AUTO_START=restore
CARDINALITY=1
...
HOSTING_MEMBERS=joords=01 joords=02 joords=03 joords=04
. .
START_DEPENDENCIES=hard(httpd-vip) pullup(httpd-vip)
STOP_DEPENDENCIES=hard(httpd-vip)
...
```

Create HA application on Clusterware

■ Managing the Application

```
[oracle@joord-01 ]# start resource tomcat
```

```
[oracle@joord-01 ]# crsctl status resource tomcat
```

```
NAME=tomcat  
TYPE=cluster_resource  
TARGET=ONLINE  
STATE=ONLINE on joords-01
```

```
[oracle@joord-01 ]# $ crsctl relocate resource tomcat
```

```
CRS-2673: Attempting to stop 'tomcat' on 'joords-01'  
CRS-2677: Stop of 'tomcat' on 'joord-01' succeeded  
CRS-2673: Attempting to stop 'tomcat' on 'joords-01'  
CRS-2677: Stop of 'tomcat' on 'joord-01' succeeded  
CRS-2672: Attempting to start 'tomcat' on 'joords-02'  
CRS-2676: Start of 'tomcat' on 'joord-02' succeeded  
CRS-2672: Attempting to start 'tomcat' on 'joords-02'  
CRS-2676: Start of 'tomcat' on 'joords-02' succeeded
```

```
[oracle@joord-01 ]# $ crsctl stop resource tomcat
```

Scheduling methods on RAC

- Point of interest
 - Crontabs, or windows tasks
 - DBMS_JOB
 - DBMS_SCHEDULER

Scheduling methods on RAC

- Crontab, or window tasks
 - All crontabs are the same on every node
 - Create an active/passive node activity
 - Using an action script with cluster resource
 - Add validation “active node” to the scripting
 - Active node then “DO YOUR THING” else “EXIT”
- Result:
 - HA scheduling, default shell scripts

Scheduling methods on RAC

■ Crontab, or window tasks example

■ Action script : crontab_check.sh

```
#!/bin/bash
export M_NAME=/oravar/grid/joords_cron/crontab_check.sh
export GRID_OWNER=grid
export CRS_HOME=/software/oracle/11.2.0.2/bin
case $1 in
'start')
    hostname -a > ${M_NAME};          RET=0
    ;;
'stop')
    echo "STOP" > ${M_NAME};          RET=0
    ;;
'check')
    M_VALUE=`cat ${M_NAME}`;          NDEF=`grep -c ${M_VALUE}
${CRS_HOME}/olsnodes`
    if [ $NDEF -eq 1 ] ; then ; RET=0 ; else ; RET=1 ;fi
    ;;
esac
if [ $RET -eq 0 ]; then ; exit 0 ; else ; exit 1 ;fi
```

Scheduling methods on RAC

■ Crontab, or window tasks example

■ Action configuration file : crontab_check.config

```
PLACEMENT=restricted
HOSTING_MEMBERS=joords-01 joords-02 joords-03 joords-04
CHECK_INTERVAL=30
CARDINALITY=1
ACTIVE_PLACEMENT=0
AUTO_START=always
DEGREE=1
DESCRIPTION="Cron active node resource"
RESTART_ATTEMPTS=1
ACTION_SCRIPT=/oravar/grid/joords_cron/crontab_check.sh
```

Scheduling methods on RAC

■ Crontab, or window tasks example

■ Registration

```
[root@joords-01 ]# crsctl add resource CRONTAB_CHECK /  
-type cluster_resource
```

Scheduling methods on RAC

■ DBMS_JOB

- Supports multi-instance execution of jobs

- NOT instance aware, Node affinity

 - EXECUTE DBMS_JOB.INSTANCE(job=>123, **instance=>1**);

 - No instance parameter, default instance where job is created

 - Node affinity also with Advanced Queuing

■ Result:

- A scheduling method, actions by node failure

Scheduling methods on RAC

■ DBMS_SCHEDULER

- Fully instance aware
- Distribute your workload on services, instances
- Need Job classes
 - Control the usage of resources through the resource consumer groups
 - Managing on services
 - Transparant Application Failover

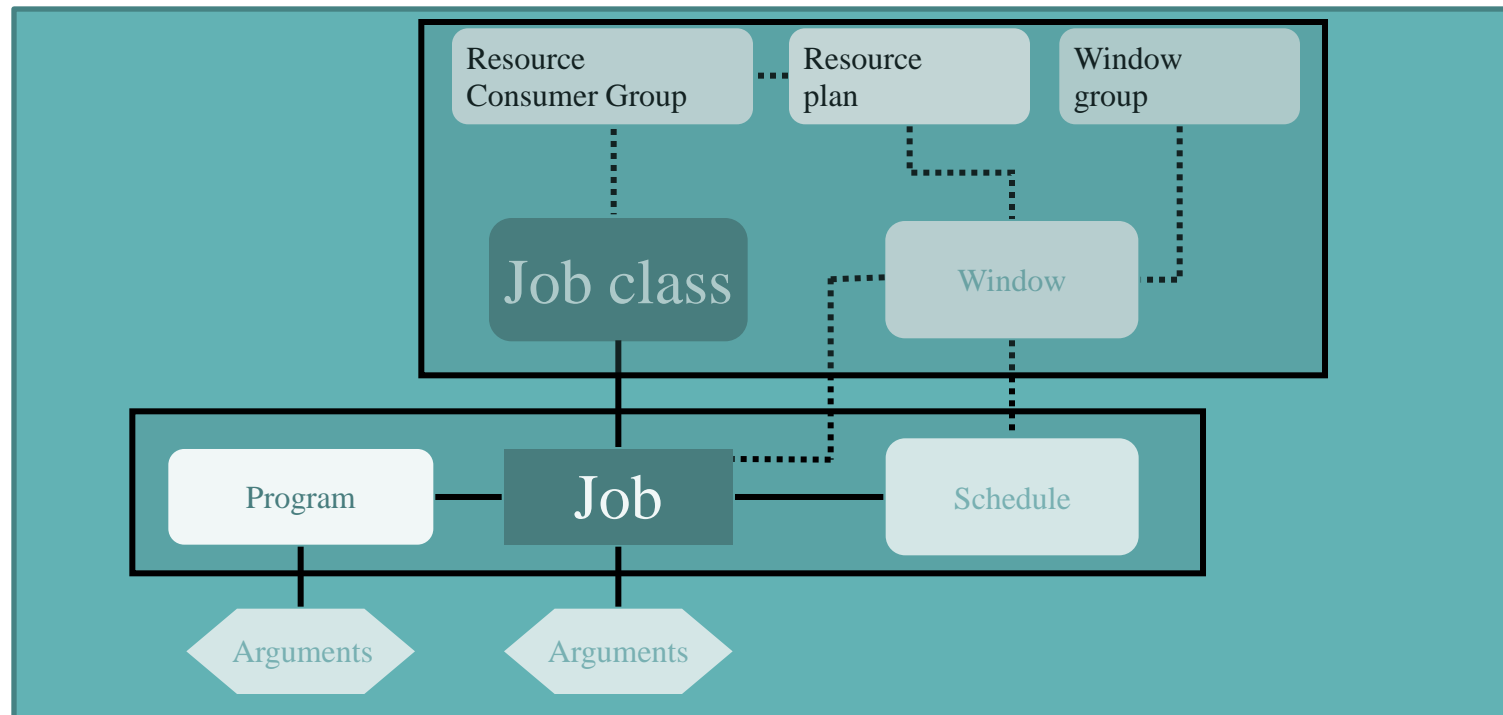
■ Result:

- HA scheduling method with load-balancing and node failover functionality

Scheduling methods on RAC

■ DBMS_SCHEDULER

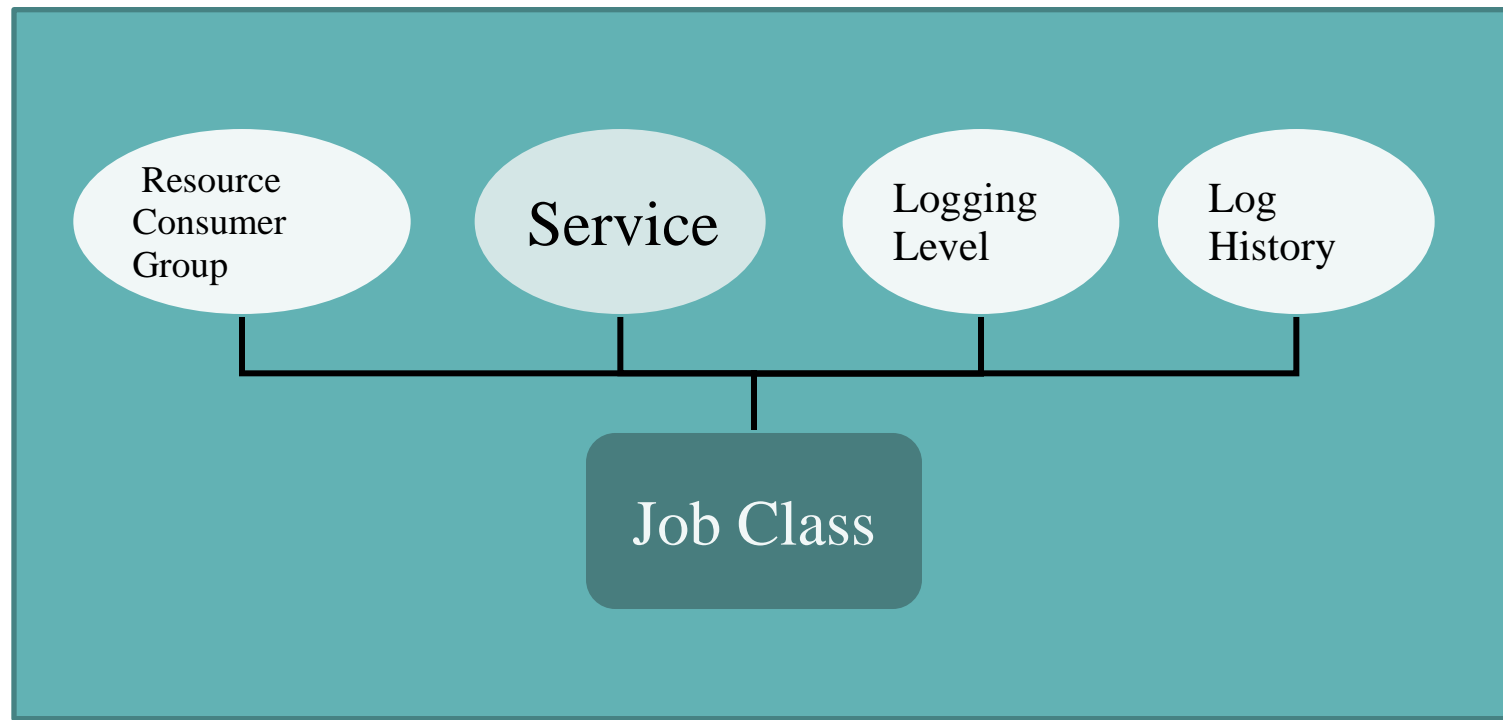
■ Overview



Scheduling methods on RAC

■ DBMS_SCHEDULER

■ Overview



Scheduling methods on RAC

■ DBMS_SCHEDULER

■ Create the services

```
The Services : OLTP_SERVICE, BATCH_SERVICE, REPORT_SERVICE

[oracle@joords-01 ]# srvctl add service -d ORCL-s OLTP_SERVICE      -r ORCL1,
ORCL2 -a ORCL3
[oracle@joords-01 ]# srvctl add service -d ORCL-s BATCH_SERVICE    -r ORCL3
-a ORCL1,ORCL2
[oracle@joords-01 ]# srvctl add service -d ORCL -s REPORT_SERVICE -r ORCL3
-a ORCL1,ORCL2

[oracle@joords-01 ]# srvctl status service -d ORCL
Service OLTP_SERVICE      is running on instance(s) ORCL1 ORCL2
Service BATCH_SERVICE     is running on instance(s) ORCL3
Service REPORT_SERVICE    is running on instance(s) ORCL3

[oracle@joords-01 ]# srvctl config service -d ORCL
OLTP_SERVICE PREF   : ORCL1 ORCL2 AVAIL: ORCL3
BATCH_SERVICE PREF : ORCL3 AVAIL: ORCL1 ORCL2
REPORT_SERVICE PREF: ORCL3 AVAIL: ORCL1 ORCL2
```

Scheduling methods on RAC

■ DBMS_SCHEDULER

- Create the job classes and assign a service to them

```
BEGIN
  DBMS_SCHEDULER.create_job_class(
    job_class_name => 'OLTP_JOB_CLASS',
    service        => 'OLTP_SERVICE');
  DBMS_SCHEDULER.create_job_class(
    job_class_name => 'BATCH_JOB_CLASS',
    service        => 'BATCH_SERVICE');
  DBMS_SCHEDULER.create_job_class(
    job_class_name => 'REPORT_JOB_CLASS',
    service        => 'REPORT_SERVICE');
END;
/
```

- `dba_scheduler_job_classes`

Scheduling methods on RAC

■ DBMS_SCHEDULER

- Create a job and assign a job class to it.
- Distribute your workload on services, instances

```
BEGIN
  sys.dbms_scheduler.create_job(
    job_name => "OLTP_JOB",
    program_name => 'my_oltp_procedure',
    start_date => systimestamp',
    job_class => 'OLTP_JOB_CLASS',
    comments => 'This job will run on ORCL1 and ORCL2
                instances',
    auto_drop => FALSE,
    enabled => TRUE);

END;
/
```

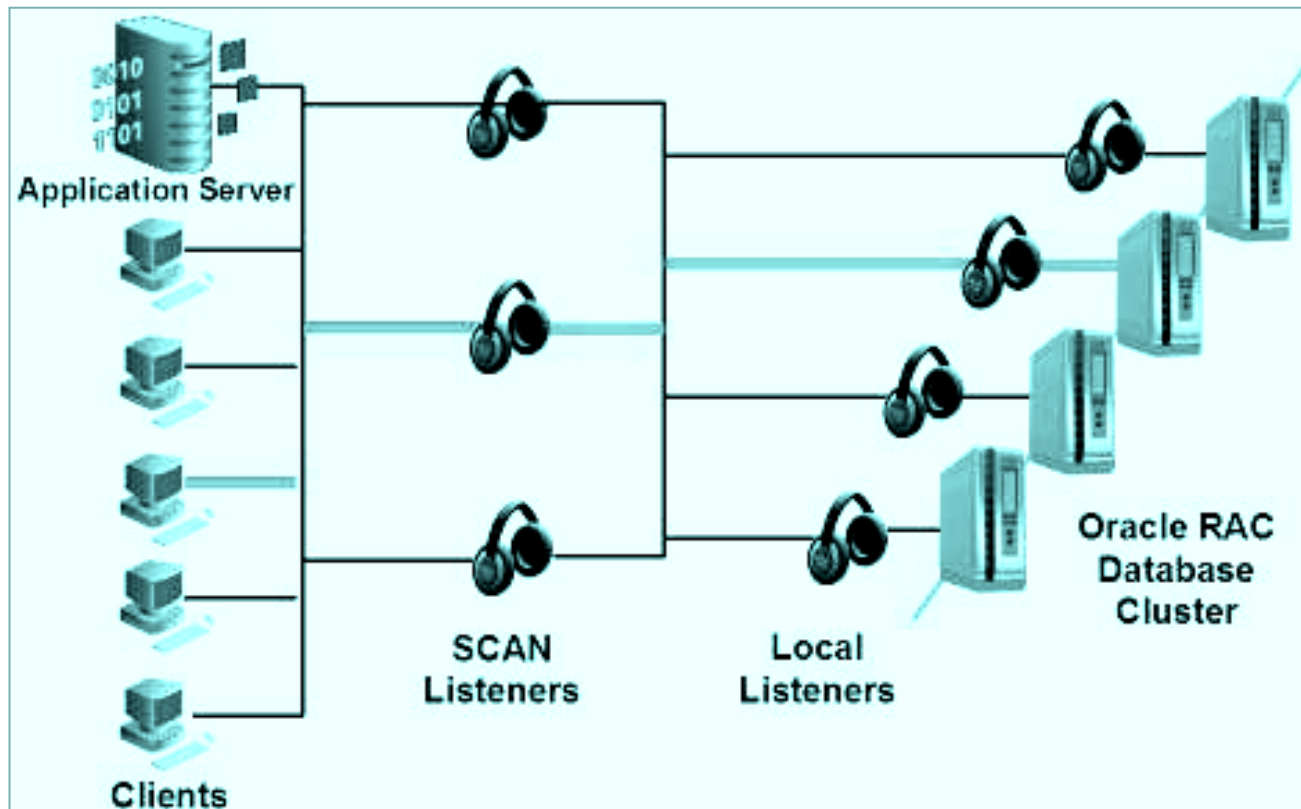
- `dba_scheduler_jobs` , `dba_scheduler_running_jobs`.

SCAN Listener configurations

- Point of interest
 - SCAN configuration on Cluster
 - SCAN Compatibility Matrix

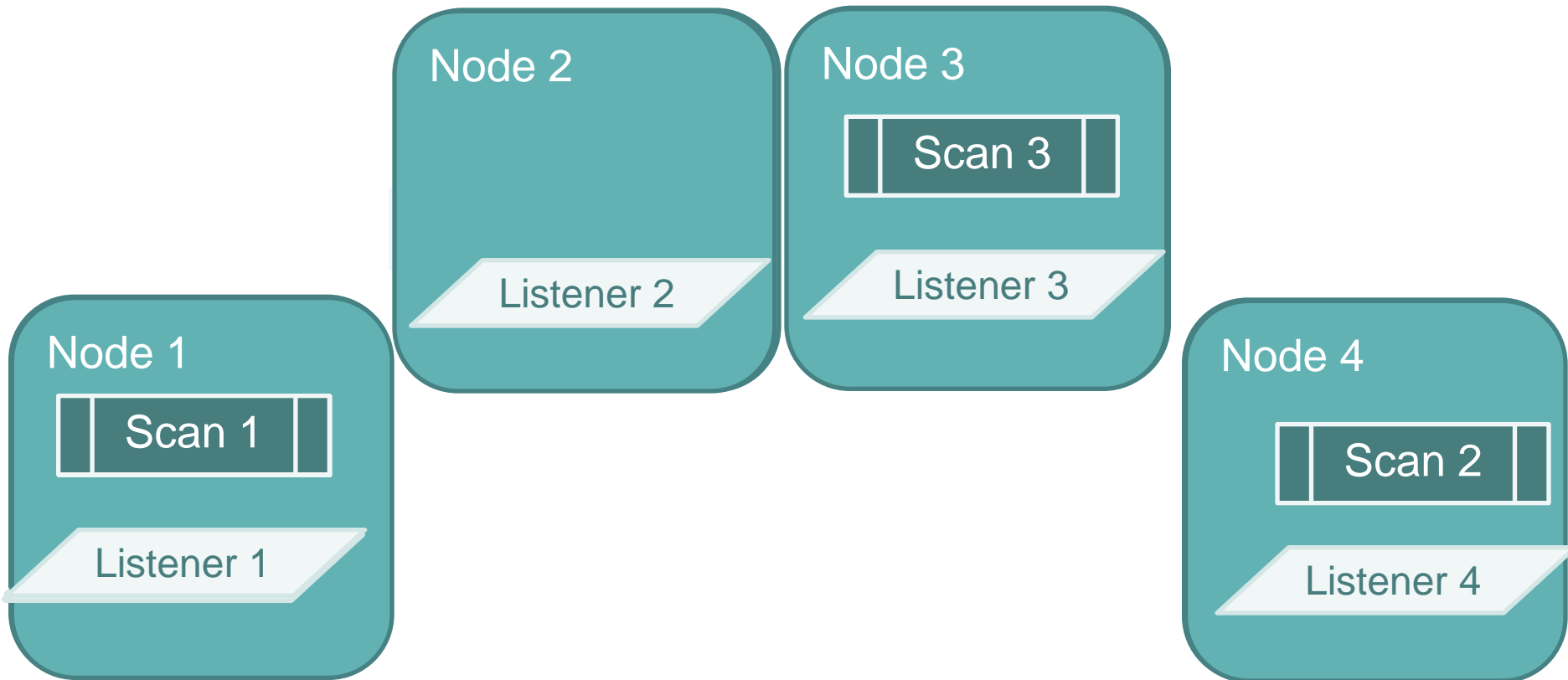
SCAN Listener configurations

- SCAN configuration on Cluster



SCAN Listener configurations

■ Overview SCAN relocation



SCAN Listener configurations

■ SCAN configuration on Cluster

- Define the SCAN in your corporate DNS (Domain Name Service)

- joords-scan.joords.nl

- IN A 133.22.67.194

- IN A 133.22.67.193

- IN A 133.22.67.192

- Use the Grid Naming Service (GNS) [optional]

- We don't use this

- Created during cluster configuration

- Default installation on port 1521 and default listener names.

SCAN Listener configurations

■ SCAN configuration on Cluster

```
[grid@joords-01 ~]# srvctl config scan_listener  
SCAN Listener LISTENER_SCAN1 exists. Port: TCP:1521  
SCAN Listener LISTENER_SCAN2 exists. Port: TCP:1521  
SCAN Listener LISTENER_SCAN3 exists. Port: TCP:1521
```

```
[grid@joords-01 ~]# srvctl config scan  
SCAN name: joords-scan, Network:  
1/133.22.67.0/255.255.255.0/  
SCAN VIP name: scan1, IP: /joords-  
scan.joords.nl/133.22.67.192  
SCAN VIP name: scan2, IP: /joords-  
scan.joords.nl/133.22.67.193  
SCAN VIP name: scan3, IP: /joords-  
scan.joords.nl/133.22.67.194
```

(SCAN) Listener configurations

■ SCAN configuration on Cluster

■ Database configuration using SCAN

```
local_listener → (DESCRIPTION=(ADDRESS_LIST=  
    (ADDRESS=(PROTOCOL=TCP)  
    (HOST=133.22.67.111) (PORT=1521))))  
remote_listener → joords1-scan.joords.nl:1521
```

■ TNSNAMES.ORA

```
Serv_DB = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)  
    (HOST = joords1-scan.joords.nl) (PORT = 1521))  
    (CONNECT_DATA = (SERVER = DEDICATED)  
    (SERVICE_NAME = MyDBservice)))
```

SCAN Listener configurations

■ Compatibility Matrix

Oracle Client Version	Oracle Database Version	Comment
Oracle Database 11g Release 2	Oracle Database 11g Release 2	No change required.
Oracle Database 11g Release 2	Pre- Oracle Database 11g Release 2	Add the SCAN VIPs as hosts to the REMOTE_LISTENER parameter.
Pre- Oracle Database 11g Release 2	Oracle Database 11g Release 2	Change the client TNSNAMES.ora to include the SCAN VIPs (* see below). IF the database was upgraded using the DBUA from a pre-11g Rel. 2 database, the DBUA will configure the REMOTE_LISTENER parameter to point to the node-VIPs as well as the SCAN.
Pre- Oracle Database 11g Release 2	Pre- Oracle Database 11g Release 2	If you want to make use of SCAN (recommended): add the SCAN VIPs as hosts to the REMOTE_LISTENER parameter. AND Change the client TNSNAMES.ora to include the SCAN VIPs (* see below). Otherwise, no change required.

SCAN Listener configurations

■ Compatibility Matrix < 11gR2 versions

■ TNSNAMES.ORA for Oracle Client

```
serv_DB =(DESCRIPTION=  
  (ADDRESS_LIST= (LOAD_BALANCE=on) (FAILOVER=ON)  
    (ADDRESS=(PROTOCOL=tcp) (HOST=133.22.67.192) (PORT=1521))  
    (ADDRESS=(PROTOCOL=tcp) (HOST=133.22.67.193) (PORT=1521))  
    (ADDRESS=(PROTOCOL=tcp) (HOST=133.22.67.194) (PORT=1521)))  
  (CONNECT_DATA=(SERVICE_NAME= MyDBservice)))
```

■ TNSNAMES.ORA for remote_listener DB

```
clst_listener =(DESCRIPTION=  
  (ADDRESS_LIST=  
    (ADDRESS=(PROTOCOL=tcp) (HOST=133.22.67.192) (PORT=1521))  
    (ADDRESS=(PROTOCOL=tcp) (HOST=133.22.67.193) (PORT=1521))  
    (ADDRESS=(PROTOCOL=tcp) (HOST=133.22.67.194) (PORT=1521)))
```

SCAN Listener configurations

- Compatibility Matrix
- Local listener works for local DB's
- SCAN listener for Load balancing

Votingdisk issue on ASM

- Mirroring in ASM
 - External, Normal, High
 - Normal redundancy by voting file, 3 voting files are places, each on one ASM disk.
 - ASM diskgroup (normal) consist of two ASM disks
 - New quorum failgroup for voting disk

Votingdisk issue on ASM

■ The Problem

```
[grid@joords-01 ]# crsctl query css votedisk
1. ONLINE 5d7ab463aede4f91bfe985df041dc86b (ORCL:OCRVOTE1) [OCRVOTE]
2. ONLINE 78ab65495fe84f19bfcc05e1e8bcec83 (ORCL:OCRVOTE2) [OCRVOTE]
Located 2 voting disk(s).
```

```
SQLPLUS@ASM1>select name, path, label, header_status from v$asm_disk
```

NAME	PATH	LABEL	HEADER_STATU
OCRVOTE1	ORCL:OCRVOTE1	OCRVOTE1	MEMBER
OCRVOTE2	ORCL:OCRVOTE2	OCRVOTE2	MEMBER
	ORCL:OCRVOTE3		

Votingdisk issue on ASM

■ Problem

■ OCRVOTING has the correct labeling

```
[root@joords-01 ]# kfed read /dev/oracleasm/disks/OCRVOTE1A | egrep  
'dsknum|dskname|grpname|fgname|hdrsts|mntstmp.hi|mntstmp.lo'  
[root@joords-01 ]# kfed read /dev/oracleasm/disks/OCRVOTE1A | egrep  
'dsknum|dskname|grpname|fgname|hdrsts|mntstmp.hi|mntstmp.lo'  
[root@joords-01 ]# kfed read /dev/oracleasm/disks/OCRVOTE1A | egrep  
'dsknum|dskname|grpname|fgname|hdrsts|mntstmp.hi|mntstmp.lo'
```

Column hdrsts tells you:

MEMBER and is now in use

FORMER used to be member and is available, safe to delete now

Votingdisk issue on ASM

■ The Solution

- Clean the disk header of ocrvoting1c disk
- Add disk ocrvoting1c to the ocrvote diskgroup as “quorum failgroup”

```
[root@joords-01 ]# dd if=/dev/zero  
of=/dev/oracleasm/disks/OCR Voting1C bs=1024 count=100  
  
SQLPLUS@ASM> ALTER DISKGROUP OCRVOTE  
ADD QUORUM FAILGROUP OCRVOTE3_EVA3 DISK 'ORCL:OCR VOTE3' NAME  
OCR VOTE3;
```

ASM spfile - ASM mirroring

■ ASM spfile

■ Pointer of the spfile in the OLR.ORA

```
strings from file \etc/oracle/olr.loc/  
    ` $ORACLE_HOME/cdata/<hostname>.olr  
.....  
spfile="+OCRVOTE/joords-  
clst/asmparameterfile/registry.253.743691783"  
.....
```

■ asmcmd copy the spfile ASM

```
[grid@joords-01 ]# asmcmd  
asmcmd> spget  
+OCRVOTE/joords-clst/asmparameterfile/registry.253.743691783  
asmcmd> spcopy -u +OCRVOTE/joords-  
clst/asmparameterfile/registry.253.743691783 +DATA11G/joords-  
clst/asmparameterfile/spfileASM.ora  
asmcmd> spget  
+DATA11G/joords-clst/asmparameterfile/spfileASM.ora
```

ASM spfile - ASM mirroring

■ ASM spfile

■ Check the spfile location

```
Strings from file `/etc/oracle/olr.loc/'  
.....  
SPFile="+DATA11G/joords-clst/asmparameterfile/spfileASM.ora "  
.....
```

■ Move to the new ASM spfile

```
Every node in the cluster execute :  
[root@joords-01 ~]# /software/oracle/11.2.0.2/bin/crsctl stop has  
[root@joords-01 ~]# /software/oracle/11.2.0.2/bin/crsctl start has  
sqlplus / as sysasm  
SQL@ASM> show parameter sp  
spfile +DATA11G/joords-clst/asmparameterfile/spfileasm.ora  
SQL@ASM>
```

ASM spfile - ASM mirroring

■ ASM spfile

■ Find all spfiles in asmcmd

```
ASMCMD> find / *spfile*  
+DATA10G/ORA10G/spfileORA10G.ora  
+DATA11G/ORA11G/spfileORA11G.ora  
+DATA11G/joords-clst/asmparameterfile/spfileASM.ora
```

ASM spfile - ASM mirroring

■ ASM mirroring

■ Diskgroups when ASM mirrors the AU's

The crucial aspect of administering the diskgroups when ASM mirrors the AUs, is having enough free space to re-copy the AUs when a loss occurs. This is not a concern when external redundancy is used, but it is for normal and high redundancy, and requires that the ASM administrator be aware of the free space needs

■ Tracking free space within a diskgroup

■ View V\$ASM_DISKGROUP

TOTAL_MB => indicates the total space in the diskgroup

FREE_MB => indicates the total free space in the diskgroup

REQUIRED_MIRROR_FREE_MB => indicates free space required in the diskgroup to restore redundancy by copying allocation units.

USABLE_FILE_MB => indicates how much of the FREE_MB may be safely used whilst leaving enough free space for mirror copy recovery in the case of disk failure

TYPE => indicates the redundancy attribute of the diskgroup.

OFFLINE_DISKS => indicates how many disks are offline in the diskgroup



**Vragen
Antwoorden**

CONSULTING | MANAGED SERVICES | EDUCATION

WWW.TRANSFER-SOLUTIONS.COM

TRANSFER