

# Oracle Database Failover Cluster with Grid Infrastructure 11g Release 2

**DOAG Conference 2011**

Robert Bialek

Principal Consultant  
Trivadis GmbH

November 16, 2011

BASEL    BERN    LAUSANNE    ZÜRICH    DÜSSELDORF    FRANKFURT A.M.    FREIBURG I.BR.    HAMBURG    MÜNCHEN    STUTTGART    WIEN

1

2011 © Trivadis

Oracle Database Failover Cluster with Grid Infrastructure 11g Release 2  
November 16, 2011

**trivadis**  
makes IT easier. ■ ■ ■

# Who Am I

- Principal consultant and trainer at Trivadis GmbH in Munich (DE)
  - robert.bialek@trivadis.com
- Focus: Oracle database high availability
  - Grid Infrastructure, Real Application Cluster, Data Guard, Maximum Availability Architecture, Failover Cluster
  - Architecture design
  - Review, troubleshooting, coaching
  - Backup and recovery
  - Performance tuning
  - Linux

**ORACLE**  
Certified Master

# Trivadis facts & figures



11 Trivadis locations with more than 550 employees

Financially independent and sustainably profitable

Key figures 2010

- Revenue CHF 101 / EUR 73 Mio.
- Services for more than 700 clients in over 1,800 projects
- Over 170 Service Level Agreements
- More than 5,000 training participants
- Research and development budget: CHF 5.0 / EUR 3.6 Mio.

# Why we are special

## **Customer-specific solution competence and vendor independence**

- offers substantiated techniques and skills as well as self-developed approaches
- guarantees repeatable quality and a safe execution

## **Technology competence**

- offers more than 17 years of expertise in Oracle and Microsoft
- has its own Technology Center and strives for technological excellence

## **Solution and integration expertise**

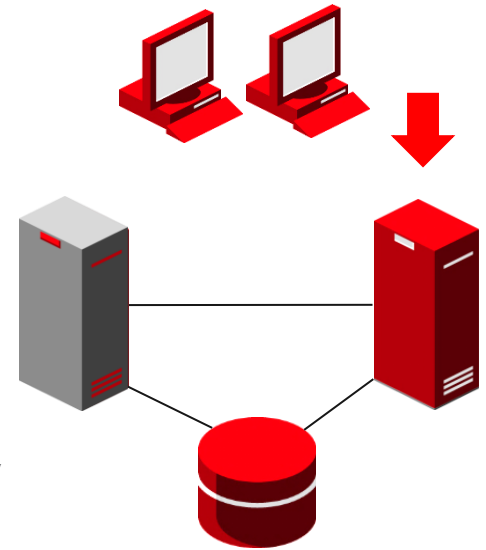
- has a wide and cross-sectoral customer basis and more than 1800 projects every year
- Combines technological expertise with an understanding of the specific business needs of the client

## **Support for the entire IT project lifecycle**

- has a modular portfolio of services for the entire IT project lifecycle
- provides the appropriate combination of solutions and services for every „level of maturity“

# Introduction

- Failover database cluster
  - Very popular **database service** high availability solution
  - Service active **only** on one node in a cluster at every point in time
  - Build-in monitoring, restart and failover functionality
- **Why** Grid Infrastructure as the cluster software stack ?
  - Stable and feature rich cluster manager – proved especially in combination with RAC
  - From 10g Release 2 onwards able to protect any kind of application
  - Includes ASM – volume manager and a file system for Oracle database files
  - Available for almost all operating systems
  - Support from one software vendor



# Introduction

- **Why** Grid Infrastructure as the cluster software stack ?
  - **Licensing**, source: *Oracle Database Licensing Information 11g Release 2 (11.2)*

*"Oracle Clusterware can be used to protect **any application** (restarting or failing over the application in the event of a failure), **free of charge**, if **one or more** of the following conditions are met:*

- *The server OS is supported by a valid Oracle Unbreakable Linux support contract.*

*...*

- *At least one of the servers in the cluster is licensed for Oracle Database (SE or EE) "*

- For active/passive failover database cluster you may benefit from the "10-day rule usage"
  - Source: <http://www.oracle.com/us/corporate/pricing/specialty-topics/index.html>, *Licensing Data Recovery Environments*
  - But only with data mirroring, **not data replication**

# Introduction

- Presented method can be used
  - For a database service high availability with **every** edition (EE, SE, ...)
  - For **every** (supported) database release
  - **Without** additional options which are subject to licensing conditions
  - Actually, to make **every application** highly available in an Oracle cluster
- For older (10.2, 11.1) Clusterware releases there is an Oracle white paper
  - <http://www.oracle.com/technetwork/database/clusterware/overview/index.html>
  - Oracle will **discontinue** to provide updated papers – see MOS article 790189.1
  - Due to many changes in the current 11.2 version **don't** use the “old” method/scripts without modifications

# AGENDA

1. Cluster Resources
2. Registering Database Resources
3. Resource Monitoring
4. Resource Placement



# Cluster Resources

- **Resource** – registered component managed by Oracle Clusterware
  - Consists of **resource attributes**

```
oracle@white:~/ [+ASM1] crsctl status resource FO1.db -p
NAME=FO1.db
TYPE=fo_db.type
ACL=owner:oracle:rwx,pgrp:oinstall:rwx,other::r--
ACTION_SCRIPT=/u00/app/oracle/local/dba/bin/crs_db.ksh
AGENT_FILENAME=%CRS_HOME%/bin/scriptagent
CHECK_INTERVAL=15
ENABLED=1
START_DEPENDENCIES=hard(ora.U01.dg,ora.U02.dg) . . .
```

- Every attribute must be defined in a **resource type**
  - Attribute name, value type, default value, required/optional
  - **Generic** predefined types: **cluster\_resource**, **local\_resource**, application
  - User defined types can also be created

# Cluster Resources

- Resource attributes
  - For some of them defaults can be used (e.g. DEGREE, CARDINALITY, ...)
  - For the others, values must be specified – depending on the configuration and requirements

## Configuration

- NAME
- TYPE
- ACL
- ACTION\_SCRIPT
- AGENT\_FILENAME
- START\_DEPENDENCIES
- STOP\_DEPENDENCIES

## Monitoring

- CHECK\_INTERVAL
- RESTART\_ATTEMPTS
- FAILURE\_INTERVAL
- FAILURE\_THRESHOLD
- UPTIME\_THRESHOLD

## Placement

- PLACEMENT
- SERVER\_POOLS
- LOAD

# Cluster Resources – Most Important Attributes

## ■ START\_DEPENDENCIES

- Set of relationships considered during resource startup/switchover/failover
- Dependency types: hard, weak, attraction, pullup, dispersion
- Modifiers: intermediate, global, concurrent, always, type, ...

```
START_DEPENDENCIES='hard(ora.U01.dg,ora.U02.dg,FO1.lsnr)  
pullup(ora.U01.dg,ora.U02.dg,FO1.lsnr)'
```

## ■ STOP\_DEPENDENCIES

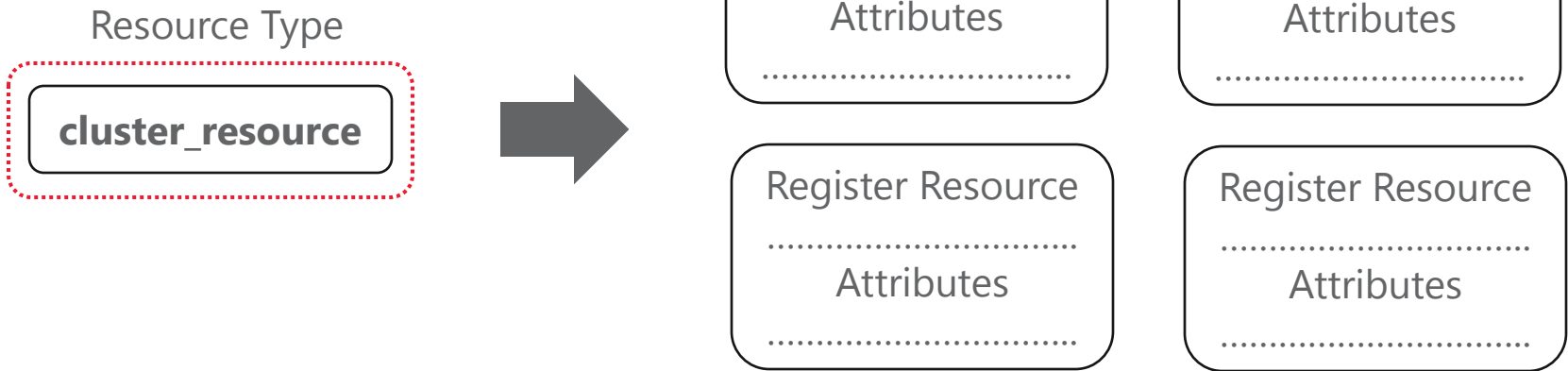
- Set of relationships considered during resource shutdown/crash
- Only hard dependency type, modifiers: intermediate, global, shutdown

```
STOP_DEPENDENCIES='hard(intermediate:ora.asm,  
shutdown:ora.U01.dg,shutdown:ora.U02.dg,FO1.vip)'
```

## ■ REQUIRED\_RESOURCES, OPTIONAL\_RESOURCES deprecated as of 11.2

# Cluster Resources – Registering

- Register and describe **always**



- Describe **once** then register

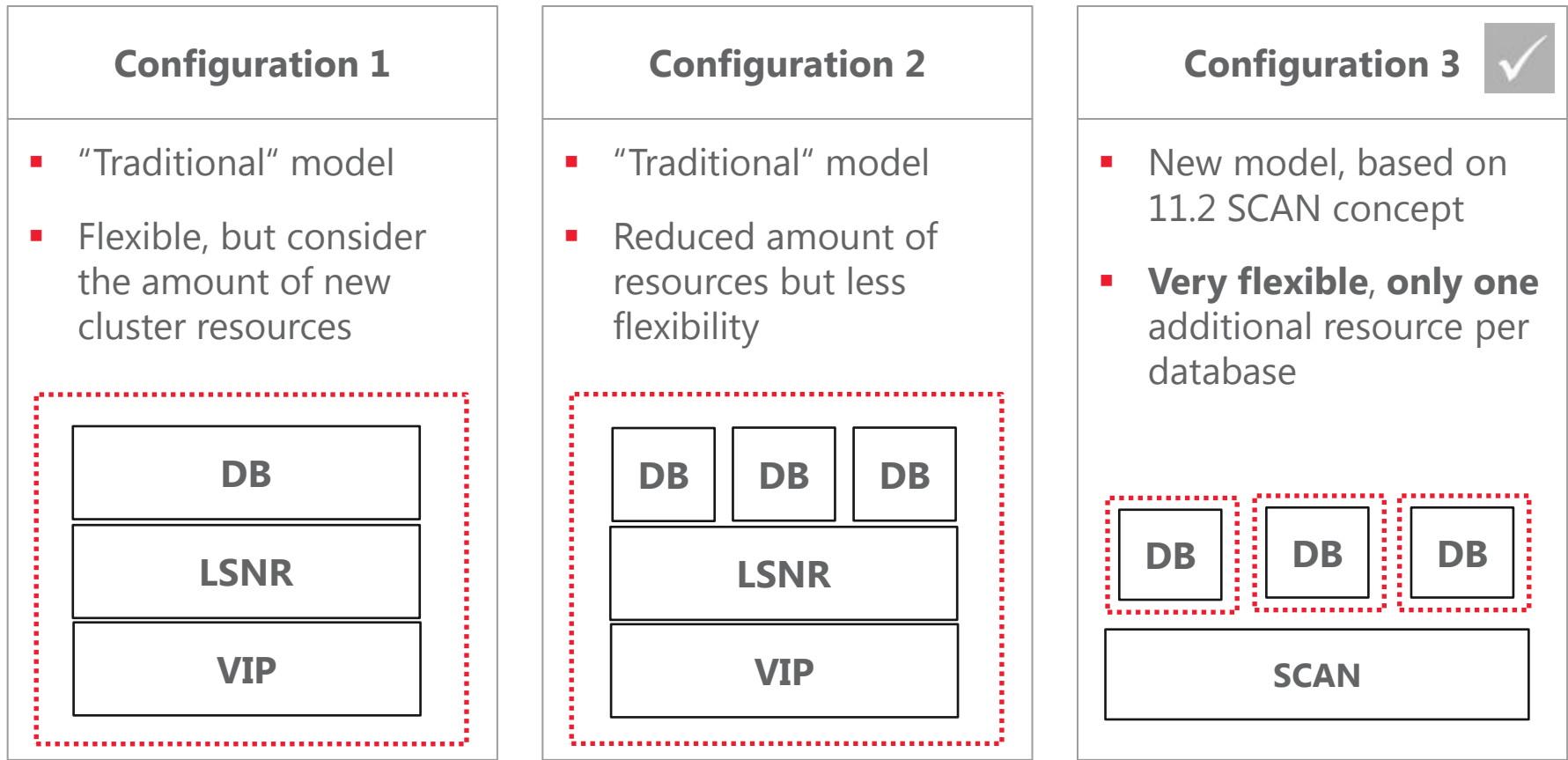


# AGENDA

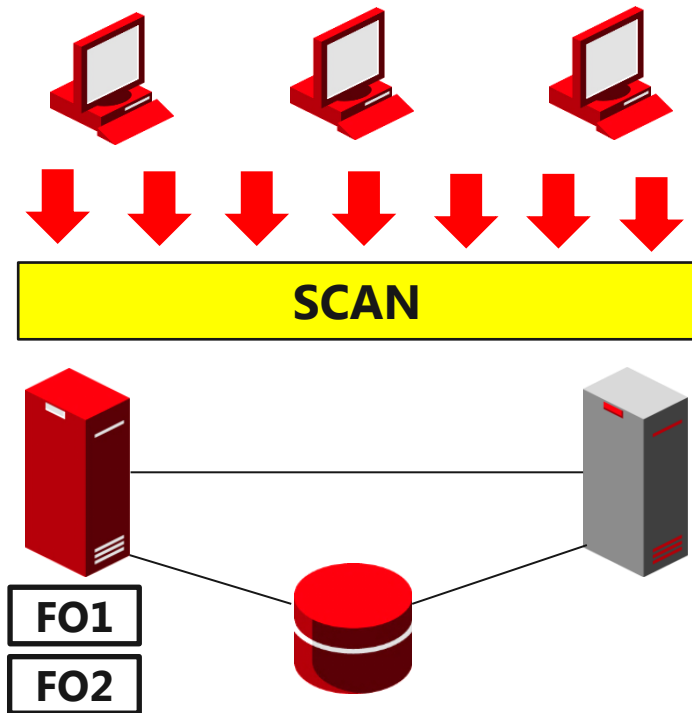
1. Cluster Resources
2. Registering Database Resources
3. Resource Monitoring
4. Resource Placement

# Registering Database Resources – Overview

- Depending on the requirements, there are many possible configurations



# Registering Database Resources – Configuration 3 in Detail



- remote\_listener – **static** value

```
remote_listener=color.example.com:1521
```

- local\_listener – **dynamic** value

```
Completed: alter database open
```

```
...
```

```
ALTER SYSTEM SET LOCAL_LISTENER='  
(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=  
(PROTOCOL=TCP) (HOST=192.168.122.12)  
(PORT=1521))))' SCOPE=MEMORY;
```

- Client configuration – **static** value

```
(ADDRESS = (PROTOCOL = TCP) (HOST = color.example.com) (PORT = 1521))  
(CONNECT_DATA =  
(SERVICE_NAME = FO1_SITE1.EXAMPLE.COM))
```

# Registering Database Resources – Agent, Action Script

- Database failover resources will be managed by the Oracle Clusterware **generic** agent called **scriptagent**
  - But not directly; it needs a user defined **action script** that is able to **start**, **stop**, **check** and **clean** a resource
  - Test it outside of Oracle Clusterware. Stability of this solution depends mainly on the **quality** of the action script!

```
AGENT_FILENAME=%CRS_HOME%/bin/scriptagent  
ACTION_SCRIPT=/u00/app/oracle/local/dba/bin/crs_db.ksh
```

- **appagent** generic script shouldn't be used anymore (deprecated in 11.2)
- Example Oracle perl action scripts

```
<GRID_INFRASTRUCTURE_HOME>/crs/demo/coldfailover/
```



# Registering Database Resources – Action Script

- Action script must follow a certain structure
  - Must return an **exit code 0** in case of a success and other in case of a failure

```
case ${Action} in
    START)
        #code to start the database instance
    STOP)
        #code to stop the database instance
    CHECK)
        #code to check the database instance
    CLEAN)
        #code to perform a clean operation
esac
```

- You can access every resource attribute value within the script

```
$_CRS_NAME #Resource name attribute
$_CRS_RESTART_ATTEMPTS #RESTART_ATTEMPTS resource attribute
```

# Registering Database Resources – Action Script

- Do we really need the generic agent **with** an action script ?
  - In 11.2 oraagent is able to directly manage failover databases – **not documented**
  - In this case database resource will be based on ora.database.type not cluster\_resource
  - Works very well, but Oracle can change the behavior of oraagent in the future **without** announcement
- The action script is a safer method
  - Can be used also for other purposes: notifications, relocating EM target, etc.
- If you need “production ready” scripts, let us know ...
  - Trivadis database toolbox **TVD-BasEnv™** delivers among other things ready-to-use cluster action scripts
  - Not only for databases!



# Registering Database Resources – Example (Configuration 3)

- **Step 1:** create a resource type (optional)

```
crsctl add type fo_db.type -basetype cluster_resource \  
-attr "ATTRIBUTE=CHECK_INTERVAL,TYPE=INT,DEFAULT_VALUE=15",\  
-attr "ATTRIBUTE=RESTART_ATTEMPTS,TYPE=INT,DEFAULT_VALUE=5",\  
-attr "ATTRIBUTE=FAILURE_THRESHOLD,TYPE=INT,DEFAULT_VALUE=2",\  
-attr "ATTRIBUTE=FAILURE_INTERVAL,TYPE=INT,DEFAULT_VALUE=3600",\  
-attr "ATTRIBUTE=UPTIME_THRESHOLD,TYPE=STRING,DEFAULT_VALUE=8h",\  
-attr "ATTRIBUTE=ACTION_SCRIPT,TYPE=STRING,DEFAULT_VALUE=  
    /.../crs_db.ksh",\  
-attr "ATTRIBUTE=START_DEPENDENCIES,TYPE=STRING,DEFAULT_VALUE=  
    'hard(ora.U01.dg,...)  
    weak(type:ora.listener.type,global:type:ora.scan_listener.type)  
    pullup(ora.U01.dg,...) ',\  
-attr "ATTRIBUTE=STOP_DEPENDENCIES,TYPE=STRING,DEFAULT_VALUE=  
    'hard(type:ora.cluster_vip_net1.type,intermediate:ora.asm,  
    shutdown:ora.U01.dg,...) ',\  
-attr "ATTRIBUTE=PLACEMENT,TYPE=STRING,DEFAULT_VALUE=balanced",\  
-attr "ATTRIBUTE=ADAPT_LOCAL_LSNR,TYPE=STRING,DEFAULT_VALUE=y"
```

# Registering Database Resources – Example (Configuration 3)

- **Step 2:** create database resources

```
crsctl add resource F01.db -type fo_db.type  
crsctl add resource F02.db -type fo_db.type
```

- **Step 3:** start database resources

```
crsctl start resource F01.db F02.db
```

```
CRS-2672: Attempting to start 'F01.db' on 'white'  
CRS-2672: Attempting to start 'F02.db' on 'white'  
CRS-2676: Start of 'F01.db' on 'white' succeeded  
CRS-2676: Start of 'F02.db' on 'white' succeeded
```

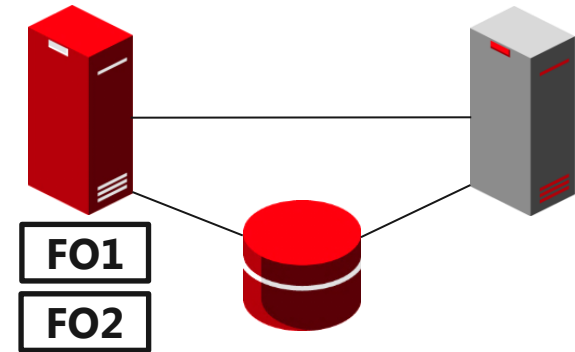
# Registering Database Resources – Demo

# AGENDA

1. Cluster Resources
2. Registering Database Resources
3. Resource Monitoring
4. Resource Placement

# Resource Monitoring

- **Resource monitoring** - one of the most important tasks of a cluster manager
  - Restarting or failing over a resource in case of a crash



- Resource restart/failover behavior can be controlled with several attributes

- **CHECK\_INTERVAL=15**
- **RESTART\_ATTEMPTS=2**
- **UPTIME\_THRESHOLD=8h**

- **FAILURE\_INTERVAL=3600**
- **FAILURE\_THRESHOLD=2**

- **RESTART\_ATTEMPTS=0** - no attempt to restart, always failover
- **FAILURE\_THRESHOLD=1** - no automatic failover

# Resource Monitoring

- To disable resource monitoring/management in a cluster
  - Useful during some maintenance tasks

```
crsctl modify resource FO1.db -attr "ENABLED=0"
```

- Resource restart/failover will be logged **only** on the OCR master node
  - To the cluster alert log file
  - Important for monitoring

```
CRS-2765:Resource 'FO1.db' has failed on server 'white'.  
CRS-2765:Resource 'FO1.db' has failed on server 'white'.  
CRS-2771:Maximum restart attempts reached for resource  
'FO1.db'; will not restart.  
...  
CRS-2768:Failure threshold exhausted by resource
```



# Resource Monitoring – Demo

# AGENDA

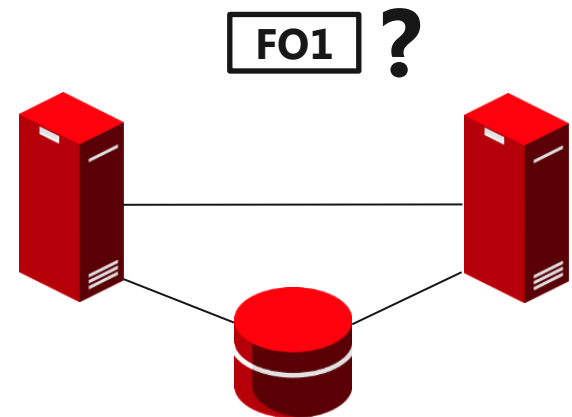
1. Cluster Resources
2. Registering Database Resources
3. Resource Monitoring
4. Resource Placement

# Resource Placement

- **Resource placement** – controls on which cluster node a resource should be started

## PLACEMENT

- **BALANCED** – less loaded servers are preferred to servers with greater loads (**LOAD** attribute)
- **FAVORED** – **preferred** are servers assigned to **SERVER\_POOLS** attribute
- **RESTRICTED** – **considers only** servers from **SERVER\_POOLS** attribute



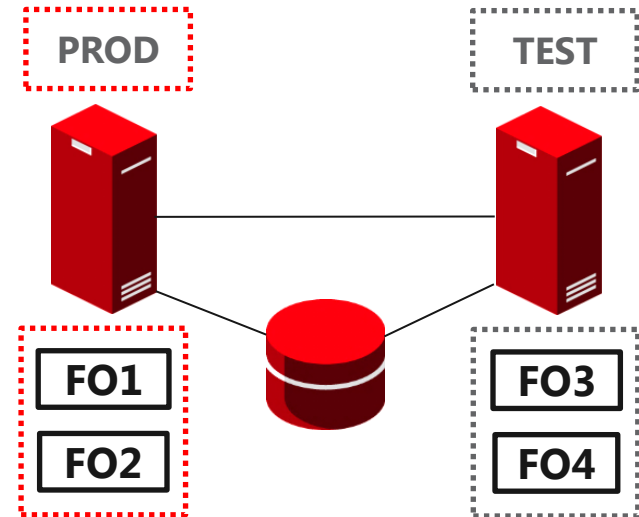
- **SERVER\_POOLS** – affinity between a resource and one or more servers

```
crsctl add srvpool fo_db.sp -attr "PARENT_POOLS=Generic, \  
> SERVER_NAMES=white"
```

# Resource Placement - Demo

# Resource Placement – Example Usage

- **PROD/TEST** cluster configuration with the following requirements
  - Production database instances are active on one server
  - Test database instances on the second one
  - In case production server crashes, automatically
    - Shutdown all test databases
    - Start all production databases on the surviving server
- How to achieve this configuration ?
  - **Cluster partitioning** with server pools – different **IMPORTANCE** attribute
  - But note: public network failure is not a reason for a server relocation between server pools



# Resource Placement – Example Usage

- **Step 1:** Create server pools

```
crsctl add srvpool PROD.sp -attr "IMPORTANCE=1, \  
> MIN_SIZE=1, MAX_SIZE=1"  
crsctl add srvpool TEST.sp -attr "IMPORTANCE=0, \  
> MIN_SIZE=1, MAX_SIZE=1"
```

- **Step 2:** Create database resources

- Production

```
crsctl add resource FO1.db -type fo_db.type \  
> -attr "SERVER_POOLS=PROD.sp, PLACEMENT=favored"
```

- Test

```
crsctl add resource FO2.db -type fo_db.type \  
> -attr "SERVER_POOLS=TEST.sp, PLACEMENT=favored"
```

# Resource Placement – Demo PROD/TEST Cluster

# Core Messages



- Oracle Grid Infrastructure (Clusterware) offers sufficient functionality to implement a failover database cluster
- Useful not only for databases
- Many ways to customize the environment, to suit your exact needs
- No additional license fees, support from one vendor
- Very good CLI tools
- EM integration possible, but not out of the box



VISIT US AT THE  
TRIVADIS STAND:

**Floor 3, No. 304**

Trivadis GmbH

Robert Bialek

Lehrer-Wirth-Str. 4  
D-81829 München

Tel. +49 89 99 27 59 30

info@trivadis.com  
www.trivadis.com

BASEL    BERN    LAUSANNE    ZÜRICH    DÜSSELDORF    FRANKFURT A.M.    FREIBURG I.BR.    HAMBURG    MÜNCHEN    STUTTGART    WIEN

33

2011 © Trivadis

Oracle Database Failover Cluster with Grid Infrastructure 11g Release 2  
November 16, 2011

**trivadis**  
makes IT easier. ■ ■ ■