



Oracle Databases on VMware High Availability

© 2011 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. This product is covered by one or more patents listed at <http://www.vmware.com/download/patents.html>.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

VMware, Inc.
3401 Hillview Ave
Palo Alto, CA 94304
www.vmware.com

Contents

1. Introduction	5
2. Oracle Architecture.....	5
3. Protection with VMware vSphere High Availability.....	6
4. Protection with Symantec ApplicationHA.....	7
5. Oracle RAC in Virtual Machines	9
5.1 Oracle RAC One Node.....	9
5.2 Multi-Node RAC.....	10
6. High Availability Options – Discussion.....	12
7. Summary.....	14
8. References.....	15

1. Introduction

High availability is the design and implementation of a service to create a system that can be operational during a given measurement period. It is factored into system design to protect business critical business operations from both unplanned downtime (an outage due to infrastructure failure) and planned downtime (for example, an outage required for system maintenance).

This document describes some high availability scenarios designed to protect a VMware virtualized Oracle database to minimize downtime within the same datacenter. The architectures described are based on VMware vSphere High Availability (HA), Symantec ApplicationHA (which is integrated with VMware vCenter Server™), and Oracle RAC. These scenarios are compared to help system architects decide which high availability solution for Oracle on VMware is best for their environment. The list is not exhaustive, and there are other solutions.

For background and more detail about the VMware functions and products in this paper, see the list of documents in Section 8, References. Designing a high availability Oracle database implementation is part of a system-wide strategy, but this document does not cover network and storage high availability features, or backup/recovery and disaster recovery considerations.

2. Oracle Architecture

The following describes a basic Oracle configuration where the client/user and associated database server process are on separate virtual machines connected across a network.

- The database server consists of an Oracle instance running in a virtual machine.
- A separate virtual machine runs an application server (alternatively, it could be an end-user workstation). This application, referred to as the *client*, establishes a connection to the database server via the SQL*Net driver. (SQL*Net is a software layer between Oracle and the networking software, providing communication between a client and the database server or from one database server to another.)
- The database server also runs the SQL*Net driver. The server detects the connection request from the application and creates a (dedicated) server process on behalf of the client process.
- The client executes SQL statements and commits the transaction.

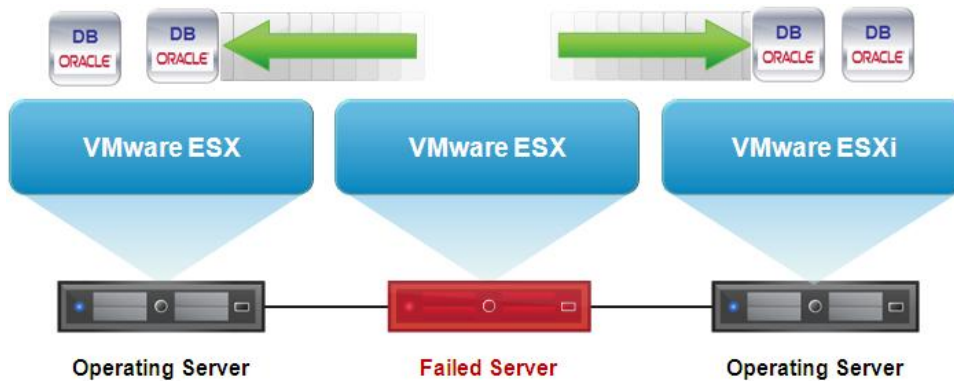
Transparent Application Failover (TAF) is a client-side feature that allows clients to reconnect to a database in the event of a failure of a database instance. This reconnect happens automatically from within the Oracle Call Interface (OCI) library (an interface that is a layer above SQL*Net). TAF can operate in one of two modes, Session Failover or Select Failover. Session Failover recreates lost connections and sessions. Select Failover replays queries that were in progress. The reconnect properties can be used in both single instance Oracle and RAC situations, so all of the scenarios described in the following sections can take advantage of this feature.

3. Protection with VMware vSphere High Availability

VMware vSphere High Availability (HA) continuously monitors all VMware ESX[®] and VMware ESXi[™] hosts in a cluster (a group of nodes running ESX hypervisor) and detects hardware failures. vCenter Server manages the nodes' resources jointly such that the cluster owns all of the CPU and memory of all nodes. The VMware HA agent placed on each node maintains a heartbeat link with the other nodes in the cluster. Each server sends heartbeat signals to the other servers in the cluster at regular intervals. If any servers lose the heartbeat signal, VMware HA reacts by restarting all virtual machines on other nodes. After a VMware HA failover and restart of a virtual machine, startup scripts/service are required to auto start the Oracle instance in the guest operating system.

Figure 1 depicts a typical scenario with multiple virtual machines that host a single-instance Oracle database with VMware HA applied (a configuration often found in existing installations). The table following the figure summarizes the features of this configuration.

Figure 1. VMware HA for Single Instance Oracle Database



Key Points	Cons
<ul style="list-style-type: none"> • Protection against ESX/ESXi server failure. • Auto restart of virtual machines. • VMware HA is easy to configure (VMware “out-of-the-box”). • A cost effective solution to provide hardware protection to many databases including non-production systems, regardless of the operating system inside the virtual machine. 	<ul style="list-style-type: none"> • No monitoring of Oracle instance inside virtual machine. • Some downtime for recovery after failure. Time required for virtual machine to restart. Operating system to boot and Oracle instance to start and complete instance recovery. • Guest operating system and Oracle patching requires downtime.

4. Protection with Symantec ApplicationHA

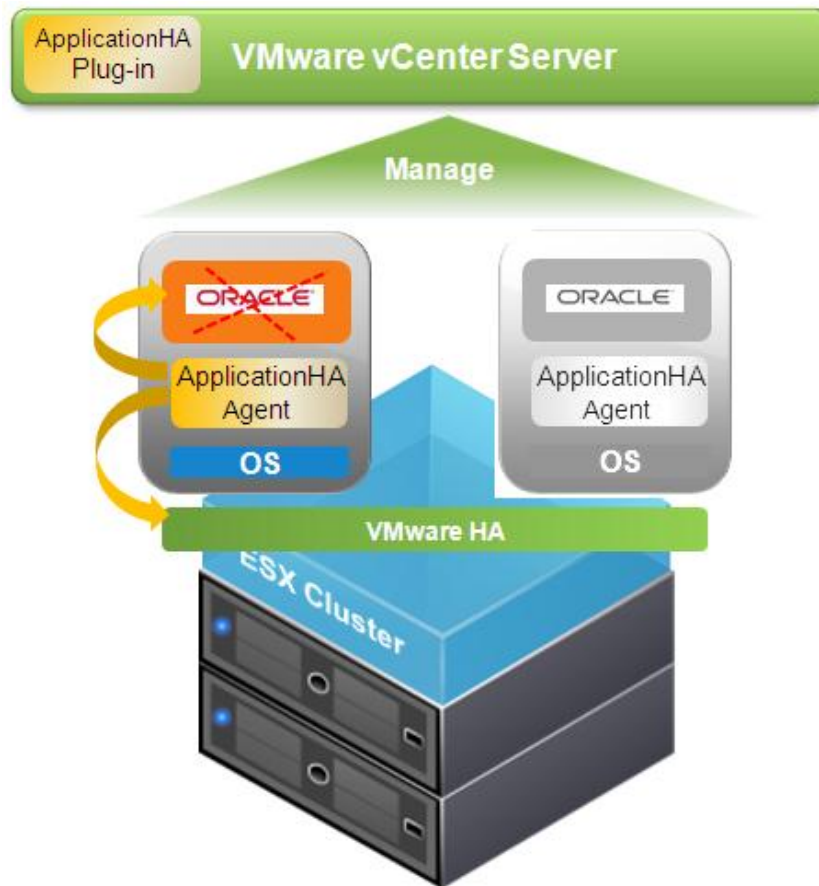
Symantec ApplicationHA is an availability solution that augments VMware HA to provide application visibility, monitoring, and control. Based on Veritas Cluster Server (VCS) technology, it utilizes a framework that provides an extendable and flexible methodology for supporting a wide range of applications. Agents interface with this framework, so the framework is commonly referred to as the *agent framework*. Agents provide the ability to perform functions such as bringing an application online or offline, or monitor it for proper operation.

There are many types of agents provided out-of-the-box by Symantec. These include agents that monitor critical resources such as file system mount points and virtual IP addresses as well as various applications. An agent is available to monitor a single instance Oracle database. The agent communicates with a central process that is installed within the guest operating system, relaying information about the state of the Oracle resource. This central process coordinates the activities of the agent so that the resource is brought online and offline as needed while dependencies are met. For example, an Oracle instance is not brought online before the storage it requires is online first.

If a failure is detected, ApplicationHA can be configured to automatically restart the Oracle application and any dependent resources. If this restart fails to rectify the application issue, it can then communicate to VMware HA to trigger a restart of the virtual machine. This is accomplished using an API that was jointly developed by VMware and Symantec.

Implementation of the ApplicationHA solution is simplified as Symantec provides a method to automatically discover the target application and any dependent resources it requires at time of configuration. Symantec further simplifies matters by providing a plug-in for VMware vSphere® that allows an administrator to view the status of an application and perform start/stop operations on it. For further information on this solution consult the paper *Virtualizing Business-critical Applications with Confidence* in the References section.

Figure 2. Logical Architecture of Symantec ApplicationHA for Virtualized Single Instance Oracle Database



Key Points	Cons
<ul style="list-style-type: none"> Builds on VMware HA to allow for application-level availability. Does not impede the functionality of VMware features such as VMware vSphere Distributed Resource Scheduler (DRS), VMware vSphere® vMotion®, VMware vSphere Distributed Power Management (DPM), and others. Ability to manage from a single pane of glass (vSphere). Oracle agent starts and stops the Oracle and dependent resources in the required order. Simple to set up and configure. 	<ul style="list-style-type: none"> In case of Oracle failure some minimal downtime is incurred for the agent to restart the Oracle instance. If VMware HA is invoked, downtime is incurred for the virtual machine to restart, the operating system to boot, and the Oracle instance to start and complete instance recovery. Guest operating system and Oracle patching requires downtime. Able to monitor only a single application per virtual machine.

5. Oracle RAC in Virtual Machines

This section discusses two Oracle Real Application Cluster (RAC) scenarios in virtual machines: RAC One Node, and the traditional multi-node RAC deployment. The implementation of Oracle RAC in virtual machines combines the uptime features of clustering with the consolidation and workload management benefits of VMware virtualization.

As per My Oracle Support document ID 249212.1, RAC 11.2.0.2 and later is supported by Oracle on VMware. For further background on Oracle RAC refer to the References section.

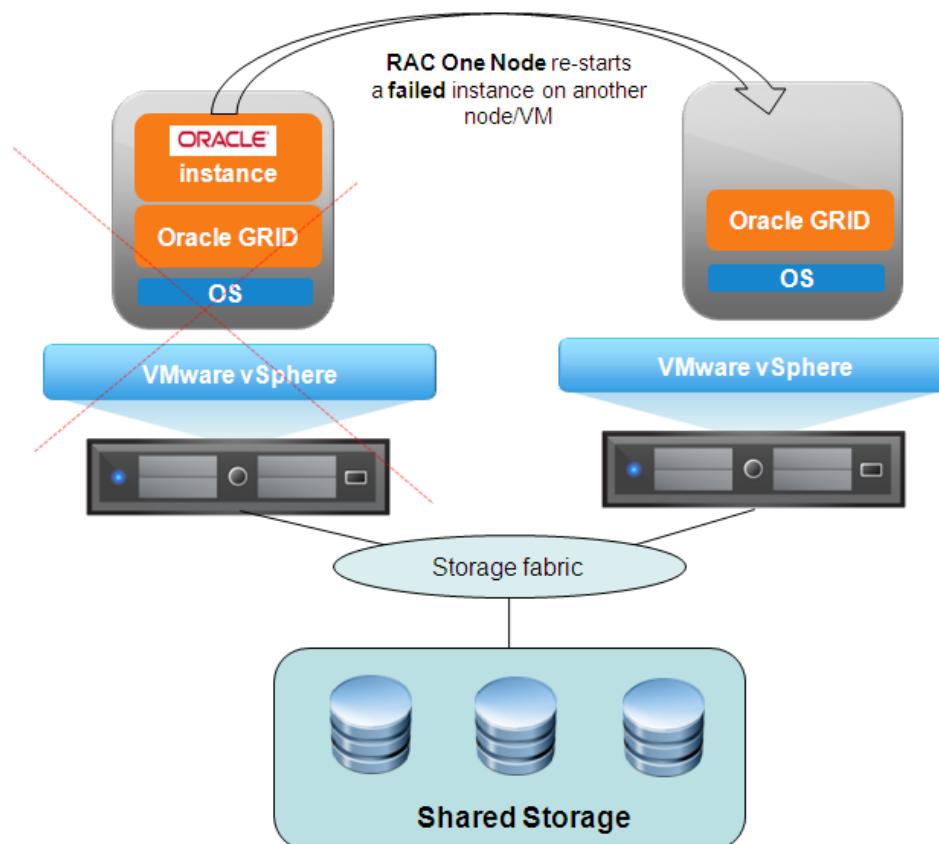
5.1 Oracle RAC One Node

With Oracle Database 11g Release 2, Oracle introduced RAC One Node. Oracle RAC One Node is a single instance of an Oracle RAC-enabled database running on one node in a cluster. A cluster is the prerequisite for installing an Oracle RAC One Node database. In a VMware environment this means the grid infrastructure is installed and running on multiple virtual machines. Oracle RAC One Node can be installed in a virtual machine in the same manner as on a physical server. If there is an ESX/ESXi host failure or a failure of the database instance, the failed database fails over to another virtual machine.

Oracle RAC One Node uses the Omotion utility to move an instance from one node to another in the cluster. This is useful when performing software maintenance on a node and minimizes the downtime of the database instance.

A logical architecture of a RAC One Node deployment in virtual machines is shown in the following figure. The table following the figure summarizes the features of this configuration.

Figure 3. Logical Architecture of Oracle RAC One Node in Virtual Machines



Key Points	Cons
<ul style="list-style-type: none"> • Failure detection in case of instance or ESX/ESXi host failure—Oracle service is relocated to a second virtual machine. • Oracle instance can be migrated to another node (using Oracle Omotion) for guest operating system and Oracle patching. • Overall a similar setup to a traditional clustering solution. • Follow <i>Disabling simultaneous write protection provided by VMFS using the multi-writer flag</i> to deploy RAC on VMFS. http://kb.vmware.com/kb/1034165 	<ul style="list-style-type: none"> • RAC skills required: Generally more complex solution, but unlike multi-node RAC an interconnect is not required. • In case of failure some downtime is incurred for the Oracle instance to start and complete instance recovery on the second node. • During Omotion, existing transactions are allowed to complete but Oracle experiences some downtime as the instance is stopped and then restarted on the target node.

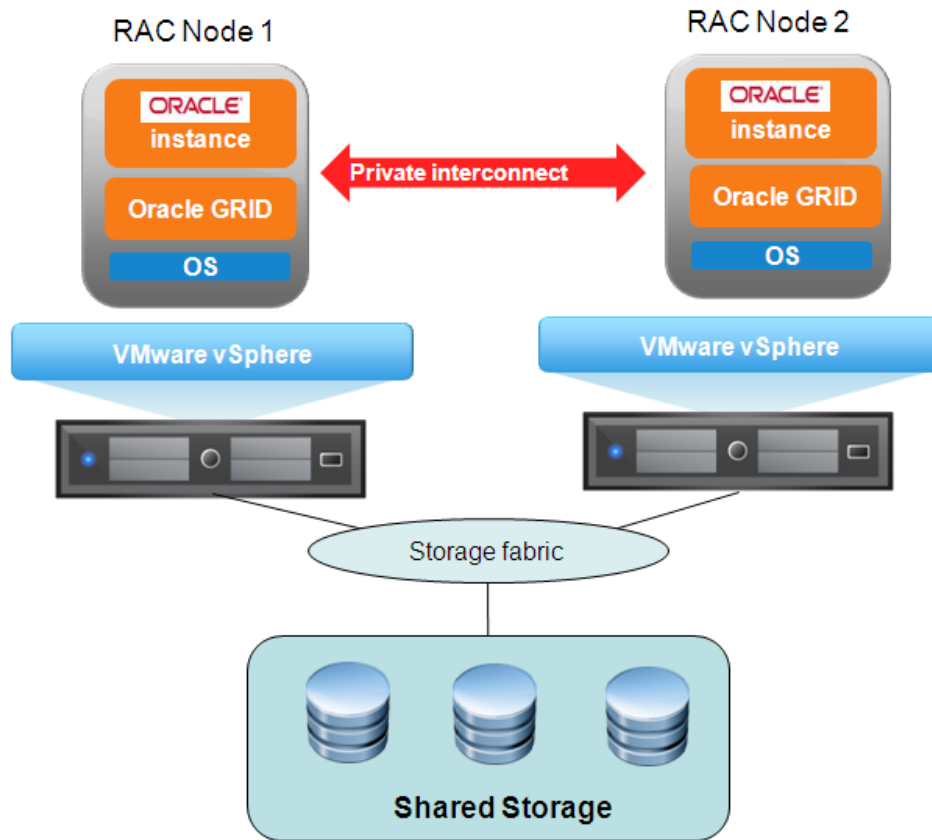
5.2 Multi-Node RAC

A multi-node RAC in the physical environment consists of several nodes (servers) connected to each other by a private interconnect. The database files are kept on a shared storage subsystem, where they are accessible to all nodes. This can also be implemented in the VMware virtual environment whereby each node corresponds to a separate virtual machine. The private interconnect is used for Cache Fusion which enables the shipping of blocks between the SGAs of nodes in a cluster.

Oracle RAC features rolling upgrades that enable some instances of the RAC installation to be available during the scheduled outage required for patch upgrades. Only the RAC instance that is currently being patched must be brought down. The other instances can continue to remain available, thus minimizing downtime for application users.

A logical architecture of a RAC deployment in virtual machines is shown in Figure 4. The table following the figure summarizes the features of this configuration.

Figure 4. Logical Architecture of Oracle RAC in Virtual Machines



Key Points	Cons
<ul style="list-style-type: none"> • Protection against ESX/ESXi host failure. • In case of a node failure, the Oracle instance is still available on the remaining nodes. • Guest operating system and Oracle patching requires no downtime. • Follow similar deployment techniques as for RAC on physical. Follow <i>Disabling simultaneous write protection provided by VMFS using the multi-writer flag</i> to deploy RAC on VMFS. http://kb.vmware.com/kb/1034165 	<ul style="list-style-type: none"> • RAC skills are required: Generally a more complex and costly solution with respect to time and resources to implement and maintain. • In case of a node failure, the Oracle instance is still available on remaining nodes, but transactions on the failed node produce errors except for SELECT statements.

6. High Availability Options – Discussion

Each of the previous scenarios provides varying degrees of protection against downtime for an Oracle database. The following table summarizes and compares the high availability scenarios described in the previous sections.

Table 1. Summary of High Availability Scenarios

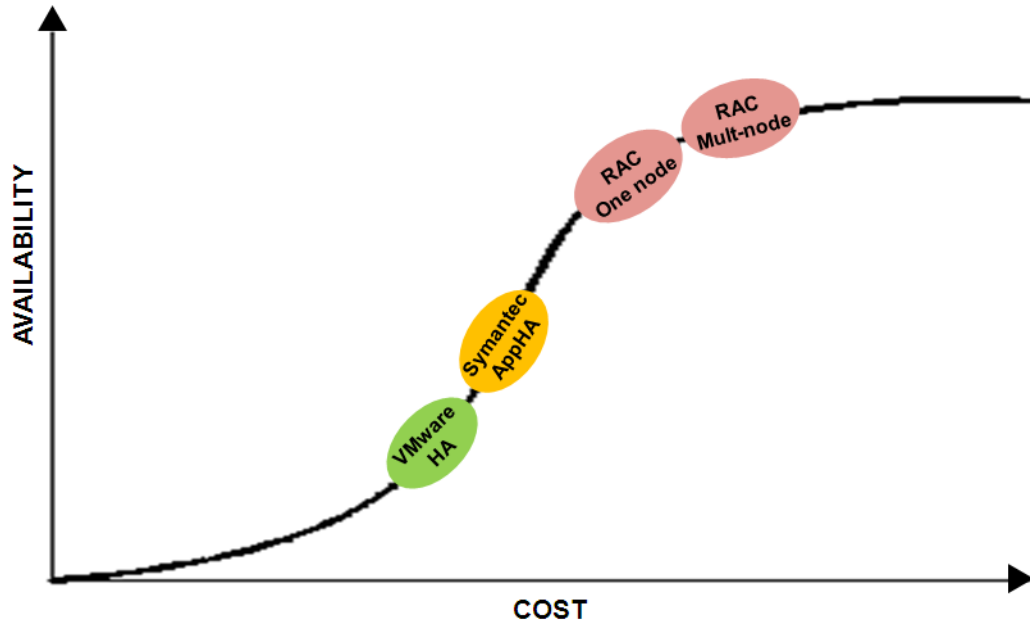
<i>Oracle DB in VM Scenario</i>	<i>ESX Host Protection</i>	<i>Oracle DB Protection</i>	<i>Minimizes Downtime for Guest OS & Oracle Patching</i>	<i>Oracle Available During Failover</i>	<i>Session/Select Persistence Through Failover Oracle TAF</i>	<i>Cost / Complexity</i>
VMware HA	YES	NO	NO	NO	MED	LOW
VMware HA + Symantec ApplicationHA	YES	YES	NO	NO	MED	MED
Oracle RAC One Node in Cluster	YES	YES	YES	NO	MED	HIGH
Oracle RAC Multiple Nodes	YES	YES	YES	YES	VERY HIGH	VERY HIGH

Consider the following points when deciding on a high availability solution for Oracle databases on VMware:

- High levels of uptime cannot realistically be achieved with any of the above scenarios alone, as overall system availability also depends on redundancy designed into the other parts of the infrastructure (network, power, storage, and the like).
- Symantec ApplicationHA provides an effective solution that bridges the gap between VMware HA and RAC by providing application awareness of the Oracle database inside a virtual machine.
- RAC solutions provide the higher degrees of protection and are preferred options for businesses that require near zero downtime in availability of the Oracle database. For organizations that can tolerate some downtime (for example, those that can accept some loss of availability for software maintenance), then VMware HA and Symantec ApplicationHA scenarios can offer a cost-effective solution to address the required service levels.
- What is the Service Level Agreement (SLA) for the business with respect to uptime/downtime, or how much downtime is the business willing to tolerate?
- What is the business cost/uptime trade-off? The RAC deployments are generally more expensive, and VMware is the most cost-efficient solution. Therefore, willingness to incur additional costs for increased availability is a key consideration.

The following figure charts the degree of availability in a deployment against the associated cost to deliver each solution. We see here a pattern of diminishing returns where at some point an organization receives fewer and fewer additional levels of availability for additional investment into the solution.

Figure 5. Availability versus Cost Trade-Off



The final design choice depends on how much downtime a business can tolerate and the cost they are willing to invest in the additional resources and skills to install and operate software for Oracle monitoring or RAC—it is a trade-off.

7. Summary

Architectural scenarios were described showing how virtualized Oracle databases can be protected from failures by using VMware HA, Symantec ApplicationHA, or Oracle RAC clustering technologies. While RAC requires a more complex setup, it provides the highest degree of protection. A major difference between the RAC and non-RAC implementations is the ability to address downtime required for guest operating system and Oracle database patching. If a business is able to tolerate some planned downtime for software maintenance then the VMware HA and ApplicationHA scenarios can offer a cost-effective high availability design that can address service level needs.

Note that all of these high availability scenarios require redundancy designed into other parts of the infrastructure (such as network, storage, and power). Ultimately, designing a highly available Oracle database on vSphere requires a trade-off between a tolerable level of downtime (which has a business cost), and the complexity of the setup, which also has a cost with respect to skills and IT resources. So, organizations must determine their realistic requirements for availability.

8. References

VMware High Availability: Concepts, Implementation, and Best Practices

http://www.vmware.com/files/pdf/VMwareHA_twp.pdf

Virtualizing Business-Critical Applications with Confidence

<http://www.vmware.com/files/pdf/techpaper/vsp4-virtualize-biz-critical-apps.pdf>

Symantec Datasheet: *Symantec ApplicationHA Virtualize business critical applications with confidence*

http://www.symantec.com/content/en/us/enterprise/fact_sheets/b-applicationHA_DS_21152712.en-us.pdf

Links to Oracle RAC documentation

<http://www.oracle.com/technetwork/database/clustering/overview/index.html>

Oracle Support statement for VMware

<https://support.oracle.com/CSP/ui/flash.html>

Document ID 249212.1 *Support Position for Oracle Products Running on VMware Virtualized Environments*

Disabling simultaneous write protection provided by VMFS using the multi-writer flag

<http://kb.vmware.com/kb/1034165>