Virtualizing Oracle: Oracle RAC on VMware vSphere

Executive Summary
While most database systems running in a VMware environment benefit from the increased reliability and recoverability inherent with virtual environments, achieving the highest possible availability still requires database clustering.

Yet, traditional clustering methods that use Raw Disk Mappings (RDMs) have generally achieved redundancy at the expense of the many benefits that result from running in virtual environments. Recent advances in the capabilities of VMware vSphere have opened the door to new clustering methods. These methods enable individual VMs in a database cluster to be migrated via VMware vMotion from one ESX host to another, creating an opportunity to synergistically combine the natural resiliency of database clusters with the high-availability and load-balancing properties of VMware virtual environments. The net result is a high-performance database system with greater reliability than what could otherwise be achieved through traditional clustering methods on either physical or virtual infrastructure.

We have delivered Oracle RAC on VMware vSphere to several large clients, including being first in the industry to do so in production environments on converged infrastructure and with vMotion enabled – something thought to be impossible at the time.

Database Virtualization: Getting Started
The fundamentals of running database systems such as Oracle in a virtual environment have become increasingly well established with newer releases of VMware vSphere. Advances in the capabilities and overall performance of VMware have put to rest the arguments about running high-performance applications as virtual machines (VMs).

However, the current release of VMware vSphere can provide continuous availability through VMware Fault Tolerance only for single vCPU systems, and then only in limited configurations. vSphere is not yet able to provide fault tolerance for multi-CPU systems, which are often needed to meet the demands of high-performance databases and other Tier 1 platforms. Thus, concerns remain around enabling high availability on virtual machines with more than one virtual CPU along with other properties that are not yet supported by VMware Fault Tolerance. Organizations with enterprise-class database platforms that require mission-critical availability or carrier-grade stability must find other ways to meet this need in a virtual environment.

As a result, traditional database clustering is still required for both mission-critical, high-availability and high-performance compute capacity. Yet, when using traditional methods, clustering virtual
machines in VMware leads to another limitation. The individual nodes in a typical cluster - whether or not these nodes are running on physical, virtual or even mixed architectures - require access to shared data. These shared drives are used for storing information common to all systems, as well as for keeping all of the nodes in a given cluster coordinated (voting and quorum drives).

In VMware, traditional VM clustering methods have required the use of Raw Disk Mappings (RDMs) on a shared Fiber Channel or iSCSI storage system. When used in this way, RDMs introduce several limitations in virtual infrastructure environments:

- RDMs are often difficult to backup and restore using traditional VMware backup methods, particularly if they are physical as opposed to virtual RDMs (vRDM).
- When using backup methods designed to take advantage of special disk access (e.g., the vStorage API), RDMs are not always backed up in the same way as the other VMware storage, leading to more complex restore procedures.
- RDMs, when used for voting and quorum drives, require VMs to turn on a feature called SCSI Bus Sharing. This feature is incompatible with certain key VMware technologies, the most important of which is VMware vMotion, which enables a VM to be migrated from one ESX host to another with no downtime (aka live migration).

As a result, a VM that is used in traditional database clustering is always tied to a dedicated ESX host. It cannot be moved to another ESX host without incurring some amount of downtime. This lack of mobility makes other key features that rely on VMware vMotion technology, such as VMware Distributed Resource Scheduler (DRS), unavailable.

The end result is that workloads within a traditional, RDM-based VMware cluster are more difficult to load balance across a DRS cluster. Further, the primary method used to ensure high availability for a database cluster is to use multiple VMs in the cluster itself - just as multiple physical servers would do in a physical cluster. VMware is unable to contribute to or enhance this capability in any meaningful way, at least for the foreseeable future. While VMware High Availability (HA) can automatically restart a failed VM in a database cluster, it is unable to follow the additional load balancing rules provided by DRS as a part of that process. Thus, the potential that system performance issues will arise in the event of a HA restart due to either the VM or ESX host failure is increased.

Oracle Support

On November 8, 2010, Oracle announced a change to its support statements for all Oracle products when running on VMware. Prior to this announcement, Oracle would provide support on VMware only when an issue could first be duplicated on physical infrastructure. This effectively kept some companies from virtualizing Oracle products and applications, as many in the user community already knew that specific Oracle configurations worked well without it. More recently (but still prior to this announcement), Oracle changed its stance on supporting virtualized applications when running on its hypervisor product. In all of these cases, Oracle RAC was expressly excluded from being supported.

The recent Oracle support statement changed things dramatically. The key portion of that change is as follows: "If a problem is a known Oracle issue, Oracle support will recommend the appropriate solution on the native OS. If that solution does not work in the VMware virtualized environment, the customer will be referred to VMware for support. When the customer can demonstrate that the Oracle solution does not work when running on the native OS, Oracle will resume support, including logging a bug with Oracle Development for investigation if required. If the problem is determined not to be a known Oracle issue, we (Oracle) will refer the customer to VMware for support. When the customer can demonstrate that the issue occurs when running on the native OS, Oracle will resume support, including logging a bug with Oracle Development for investigation if required."

NOTE: For Oracle RAC, Oracle will only accept Service Requests as described in this note for Oracle RAC version 11.2.0.2 and later. While they are known to work, Oracle still does not support versions of Oracle RAC prior to this. In making this statement, it is also clear that Oracle appropriately expects VMware will provide support for VMware when running on vSphere based virtual infrastructure. As an added measure, VMware has created an Oracle Support center with highly skilled Oracle resources, and will also now take support calls for Oracle issues. As a result, there is arguably now a greater level of support for Oracle systems when running on VMware than when running on bare metal hardware.
The issue of certification is worth a specific note as several organizations have expressed concern about it. Oracle has certified its products when running on its own hypervisor product, which is a cousin to the Xen hypervisor. This, however, appears to be more of a marketing effort than a technical issue. Certification is different from support, so confusing the two with each other should be avoided. Oracle also does not certify its products on any specific Intel hardware platforms (HP, IBM, Dell, etc.). As a practical matter, Oracle should not be expected to certify its products on vSphere because it operates at the same level in the overall stack with respect to Oracle products as physical hardware does. VMware vSphere is no different in this case than any other hardware vendor. Ironically, this is not necessarily the case for Oracle’s Hypervisor or for its close cousin, Xen. Thus, certifying on one hypervisor vs. another, in contrast to that none of the underlying hardware is certified even when running on bare metal, leads to an inconsistent and confusing message. Add to this that Oracle now enthusiastically advocates running their products on (their own) virtual infrastructure environments, and that vSphere is acknowledged in the industry as the leading and most advanced virtual infrastructure platform. Thus, the opportunity and incentive to migrate Oracle systems to virtualized infrastructure, including VMware vSphere, has never been greater.

**Breaking Free of RDM**

Overcoming this last barrier for most mission-critical applications is key. By eliminating the need for RDMs in VM clusters, the high availability of traditional database clusters can be combined synergistically with the built-in features of VMware vSphere to provide a high-performance database cluster environment with even greater resiliency than would be possible on either physical infrastructure or via traditional VMware high-availability methods. Thanks to the performance enhancements starting with VMware vSphere 4 and continuing with vSphere 5, this final barrier can now be broken with new options to take the place of RDMs. These include:

- Shared virtual disk files.
- iSCSI (or NFS) Gateway VM.
- iSCSI guest to SAN storage.

Each of these has individual advantages and drawbacks. Depending on your individual situation, one may be better than the others.

**Shared VMDK**

The Shared VMDK file is the simplest to set up. vSphere now supports using a VMDK file that can be accessed as a shared disk by multiple VMs. This is the same technology that supports VMware Fault Tolerance, and is documented in the VMware KB article KB1034165. The shared VM must have the following characteristics:

- Accessible by all VMs in the database cluster.
- Provisioned as Eager Zeroed Thick.
- Multi-Write Flag Enabled.

Once configured, each VM in the cluster can access the shared virtual disk. The primary limitations to this configuration include:

- Maximum of 8 ESX hosts per cluster.
- Shared VMDK is not uniquely associated with a VM, which can lead to more complex backup and restore issues.

**iSCSI Gateway VM**

The iSCSI Gateway VM is essentially an iSCSI SAN within the virtual infrastructure environment itself. As a virtualized SAN, it uses traditional shared disk and re-shares it via iSCSI protocol to the other VMs in the database cluster. To enhance redundancy, the iSCSI Gateway VM can be configured to take advantage of VMware Fault Tolerance. It solves the limitations of the shared VMDK file because it keeps the shared VMDK file associated with a unique VM and is not subject to the maximum cluster size of 8 hosts. It is more difficult to set up than a shared VMDK, which is also its primary drawback.

**iSCSI Guest to SAN Storage**

The final option is to configure an iSCSI target on the primary SAN storage. This option provides similar flexibility to the iSCSI Gateway VM and can potentially deliver superior performance, depending on the type of storage system. The primary drawbacks of this option are:

- Additional LUNs must be set up for each database cluster.
- Backup and restore for shared disk has the same issues as traditional RDMs.
- Not supported by all SAN types, which is problematic for traditional Fiber Channel SAN storage systems.

**Take Advantage of Improved vSphere Performance**

VMware vSphere 4 ushered in key performance enhancements across the board for virtual
machines. These enhancements continue with the introduction of vSphere 5 and allow vSphere to easily meet and exceed the compute capacity needed to run high-performance, Tier 1 applications. In particular, the enhancements to the iSCSI and networking stacks have increased I/O and efficiency gains by as much as 50% and a factor of 10, respectively. As a result, both in-guest iSCSI and NFS can be used to access shared drives, as needed. In virtual infrastructure environments leveraging converged 10 Gb Ethernet networks, the options and benefits are significant. However, traditional Fiber Channel environments can also take advantage of these benefits through the use of an iSCSI Gateway VM. When combining multiple systems with the sophistication of virtual infrastructure and Tier 1 database clusters, a significant amount of feature overlap can occur. Managing and eliminating performance bottlenecks requires a clear understanding of how these products interact with virtual infrastructure environments and with each other. While this can sometimes look complex, understanding how and why certain components provide performance boosts can be broken into logical components.

As an analogy, a sound engineer’s mixing board at a concert has dozens of control knobs, levers and switches, which can appear daunting to manage. But there is a logical flow to the system.

Sound from a microphone or instrument is first filtered into the top of the mixing board on one of several channels through a “trim” control. The sound is then “mixed” in a variety of ways (treble, bass, echo effects, etc.) as it travels down from the top to the bottom of the mixing board where another lever, called a “fader,” controls how much sound comes out on that instrument’s channel. The processed sound from each channel is then sent to a master volume control which is used to set overall volume for all of the instruments and voices. Understanding this flow lets a sound engineer use his highly skilled ears to make the concert sound great.

There is a similar logical layout and flow to how physical infrastructure, VMware and Oracle database components interact. Knowing how and where data flows through the network, how CPU and memory is assigned and how storage is accessed provides a skilled architect or admin-
istrator with a similar framework for optimizing performance. Balancing these for maximum performance still requires skill and knowledge, but the concepts of what each component does and how it works can be easily understood.

Virtual Infrastructure Architecture

Virtual infrastructure environments that are based on converged network topologies such as 10 Gb Ethernet are especially friendly to virtualized Tier 1 applications such as Oracle RAC. This is due to the available network bandwidth and the use of IP-based storage protocols (iSCSI and NFS). These architectures allow the shared drives needed for VM clusters to be hosted directly from the physical storage system. As a result, they are able to take better advantage of the hardware infrastructure which supports the virtual environment.

However, this doesn’t rule out the ability to also use traditional Fiber Channel storage systems. Here, an iSCSI Gateway VM (as described above) is used to share the FC SAN storage using iSCSI protocol. While this particular method of sharing has an additional step and requires additional tuning to achieve optimum performance, it has the advantage that all of the storage for the VM clusters is kept in one or more virtual disk files stored on a VMware VMFS data store. This provides for a consistent method of storage across all systems that can be backed up using the same virtual machine backup methods. The primary difference between Fiber Channel and IP-based storage solutions is solely that an iSCSI Gateway VM is required in FC SAN environments. While it provides clear benefits in all SAN storage solutions, the iSCSI Gateway VM is not absolutely required where iSCSI or NFS is already used as the primary storage system.

Since all of these configurations allow clustering without the need for SCSI bus sharing, all of the VMs – including iSCSI Gateway VMs – can be moved between the various ESX hosts in a DRS cluster via vMotion. This enables clusters to be freely configured such that the benefits of HA and DRS can be synergistically added to the failover capabilities inherent in Oracle RAC clusters.

Virtual Machine Architecture

The virtual machine configuration for the individual Oracle RAC nodes relies on in-guest iSCSI or in-guest NFS protocol for all shared drives. This means that each virtual machine connects directly to an iSCSI or NFS shared drive for all data that must be held in common. This connection uses the same protocols and security mechanisms that would be used if these VMs were instead servers in a purely physical environment.

On appropriate underlying infrastructure, iSCSI and NFS deliver similar performance, with unique benefits and drawbacks that are well known among storage administrators and architects. Which one to choose can be driven by available skills, layout of the underlying infrastructure, company security policies, and even personal tastes and preferences. As such, the examples used in this document are based on iSCSI, but can also be readily applied to NFS configurations.

Configuring a Virtualized Oracle System

Properly sizing the VMs which make up a RAC cluster and the Gateway VM (if implemented) is critical to maximizing performance. An example VM configuration for Oracle RAC nodes might have the following characteristics:

- Four vCPUs.
- 12 GB RAM.
- 50 GB Primary Disk (can be thin provisioned).
- Two vNICs (vmxnet3 driver): one public and one private.
- Current Linux distribution. (CentOS, Ubuntu, and Fedora have been successfully tested. Red Hat Enterprise Linux, SuSE Linux and Oracle Enterprise Linux have been used in other Oracle database solutions and should work as well.)

For those using an iSCSI Gateway VM, the configuration might look something like this:

- One vCPU.
- 4GB RAM.
- VMware Fault Tolerance (optional).
- 10 GB primary disk (can be thin provisioned).
- 100 GB secondary Disk, thick provisioned (to be shared via iSCSI).
- Two vNICs (vmxnet3 driver) - one for administration and one for iSCSI network.
- Current Linux distribution. (CentOS, Ubuntu, and Fedora have been successfully tested. Red Hat Enterprise Linux, SuSE Linux, and Oracle Enterprise Linux have been used in similar solutions and should work as well.)
Further, the example VMs as configured in this document make use of a 10 Gb Ethernet converged network for both network and storage access. When configuring for Gig-E networks, additional, dedicated network ports and interfaces at the physical layer will be required.

The above example configuration is intended to support up to a medium-sized Oracle database for development, small-scale production, and secondary support for enterprise-class, large-scale database solutions such as Oracle Exadata. This configuration should be modified as necessary to support alternate use cases.

**iSCSI Tuning**

There are a variety of options for an appropriate iSCSI Gateway VM, most all of which are some variant of Linux. These include Red Hat Enterprise Linux, Ubuntu, SUSE, Fedora, and FreeNAS, to name a few. All have an iSCSI target capability built into them. The most common iSCSI target applications found on current Linux distributions are:

- iSCSI Enterprise Target.
- TGT.
- Open iSCSI.

The iSCSI Enterprise Target is the oldest and most mature of these alternatives, but new versions of Open iSCSI have newer features and are rapidly replacing IET. Open iSCSI is included “in the box” with current versions of Red Hat and its derivatives, whereas TGT is usually found in older versions. Both are more than capable as a iSCSI platform. However, the default settings for all iSCSI systems are generally too conservative for the level of performance needed by Oracle RAC. Tuning is required to achieve a desirable level of performance. There are several online resources for tuning and configuring an iSCSI target on Linux for Oracle.

The following settings have been compiled from several community-based sources (online blogs, “man pages,” etc.). These represent some of the more common settings and should provide adequate performance for most situations. A full explanation of these parameters can be found in the Linux man pages of the iSCSI Enterprise Target configuration file (ietd.conf). Online explanations of each parameter can also be found online at (http://www.linuxcertif.com/man/5/ietd.conf), as well as other locations.

**Configuring iSCSI Enterprise Target**

On the target server, place the following in the /etc/ietd.conf file:

- `MaxConnections 1`
- `InitialR2T No`
- `ImmediateData Yes`
- `MaxRecvDataSegmentLength 262144`
- `MaxXmitDataSegmentLength 262144`
- `MaxBurstLength 262144`
- `FirstBurstLength 262144`
- `MaxOutstandingR2T 16`
- `Wthreads 16`
- `DataDigest None`
- `HeaderDigest None`

Next, adjust the amount of memory the iSCSI target system is configured to use. To do this, edit /etc/init.d/iscsitarget and change the MEM_SIZE variable to MEM_SIZE=1073741824 and then restart the iSCSI Target server by issuing the command: `/etc/init.d/iscsitarget restart`.

**Configuring iSCSI Targets with TGT**

If configuring the iSCSI target Gateway VM using TGT, use the following commands:

```
tgtadm -lld iscsi -mode target -op update -tid $tid -name MaxRecvDataSegmentLength -value 262144
tgtadm -lld iscsi -mode target -op update -tid $tid -name MaxXmitDataSegmentLength -value 262144
tgtadm -lld iscsi -mode target -op update -tid $tid -name HeaderDigest -value None
tgtadm -lld iscsi -mode target -op update -tid $tid -name DataDigest -value None
tgtadm -lld iscsi -mode target -op update -tid $tid -name InitialR2T -value No
tgtadm -lld iscsi -mode target -op update -tid $tid -name NameDataDigest -value None
```

The primary issue is that the default settings for iSCSI target servers in Linux do not allocate sufficient resources to handle the I/O needs of databases such as Oracle RAC. Tuning iSCSI to have larger memory caches and to handle larger chunks of data, as well as to spawn more threads to handle data requests more efficiently, can reap significant performance benefits. When combined with enabling Jumbo Frame support, iSCSI performance increases even more. Performance boosts of 30% to 40% have been reported by clients who enabled Jumbo Frames on 10 Gb Ethernet networks.
Configuring iSCSI Initiators (on each RAC VM)

On each of the Oracle RAC VM nodes, the iSCSI initiator needs to be tuned. To do so, add the following to /etc/sysctl.conf:

- `net.core.rmem_max = 1073741824`
- `net.core.wmem_max = 1073741824`
- `net.ipv4.tcp_rmem = 1048576 16777216`
- `net.ipv4.tcp_wmem = 1048576 16770216`
- `net.ipv4.tcp_mem = 1048576 16770216`

Reload the system parameters with the command:

```
sysctl -p
```

Then finally backup and overwrite /etc/iscsi/iscsid.conf on each VM server so it contains:

```
node.startup = automatic
node.session.timeo.replacement_timeout = 120
node.conn[0].timeo.login_timeout = 15
node.conn[0].timeo.logout_timeout = 15
node.conn[0].timeo.noop_out_interval = 10
node.conn[0].timeo.noop_out_timeout = 15
node.session.initial_login_retry_max = 4
node.session.cmds_max = 128
node.session.queue_depth = 128
node.session.iscsi.InitialR2T = No
node.session.iscsi.ImmediateData = Yes
node.session.iscsi.FirstBurstLength = 262144
node.session.iscsi.MaxBurstLength = 262144
# the default is 131072
node.conn[0].iscsi.MaxRecvDataSegmentLength = 262144
# the default is 32768
discovery.sendtargets.iscsi.MaxRecvDataSegmentLength = 262144
node.conn[0].iscsi.HeaderDigest = None
node.session.iscsi.FastAbort = No
```

Once this is done, restart the iscsi daemon with the command:

```
service iscsi restart
```

The max data length parameters are determined by the size of the kernel page (usually 4K) and then multiplied by 64 (4096 * 64 = 262144). You can experiment with the size for additional performance by doubling this size. Depending on the iSCSI target used, the maximum size for these variables differs, and if the maximum allowed size is exceeded, the default size is assumed. As such, be certain to verify these parameters based on the iSCSI target and initiator used.

Oracle Automatic Storage Management

Oracle Automatic Storage Management (ASM) should be used in this configuration to provide storage for shared disk management in exactly the same way that it would be used in a traditional physical server deployment. The primary difference is that ASM and its components all operate from within the virtual infrastructure environment, but access the shared iSCSI or NFS disk in exactly the same way. It makes no difference if the iSCSI target is directly on the storage system or accessed through a “Gateway” VM.

Key factors to keep in mind for any VM configuration include:

- All nodes in a given RAC cluster should have an identical virtual hardware configuration. Ideally, it’s best to clone a properly configured RAC VM to create the other RAC nodes in a cluster.
- VM performance, especially CPU, RAM and CPU Ready parameters, should be closely monitored to ensure maximum performance and resource utilization efficiency.
- Make use of VMXNET3 and PVSCSI drivers in VMs whenever possible to ensure maximum network and disk performance.
- Enable Jumbo Frames on all network interfaces (Suggested MTU = 9000).
- Disable unneeded services in Linux VMs.
- Tune the iSCSI initiators and targets, especially on Gateway VMs, for the performance needs...
of the VMs in the cluster using it. Multiple Gateway VMs should be considered when multiple database clusters are deployed.

The configuration of the underlying hosts systems, network, and storage in a virtual environment can have a significant impact on virtual machine performance. Oracle is particularly sensitive in this area. Be sure that the underlying hardware infrastructure is optimized to support Oracle just as if it were running directly on physical infrastructure.

It is also important to note that Oracle ASM is a sophisticated and robust database storage mechanism that is designed to make the most of physical storage systems with multiple disks. In a virtualized environment, the virtual storage system will normally have most of the performance and reliability that ASM would normally provide for itself. As a result, ASM configurations in VMware environments are usually much simpler to set up. Don't be misled! A redundant disk volume in VMware is normally presented to ASM as if it were a single disk drive. Just because ASM doesn't know that a disk volume it is using is redundant doesn't mean there is no redundancy. By the same token, ensure that you have built appropriate levels of data protection into your storage system.

**Time Synchronization**

Time synchronization in vSphere environments can be tricky, and applications which are sensitive to time require special attention. Oracle RAC is no exception. Each virtualized Oracle RAC node must be time synchronized to the other nodes. There are two methods for keeping the cluster nodes in sync. Each has its benefits and works just as well:

- **Cluster Time Synchronization Service**: This is the easier of the two options to set up. Prior to beginning the installation of Oracle RAC, make sure that all Network Time Protocol (NTP) programs are disabled (and ideally uninstalled). The Oracle RAC installer then automatically installs and configures Oracle Cluster Time Synchronization Service.

- **Enable NTP**: The default NTP configuration must be modified only to allow the Slew option (-x). This will force the NTP daemon to ensure the clock on the individual nodes does not move backwards. This option is set in different places depending on the Linux distribution used. Please refer to the documentation for the specific Linux distribution chosen for additional details.

Because time synchronization in vSphere can be sensitive, best practices suggest using VMware Tools to synchronize with the hardware clock of the ESX host system on which they are running. Testing to date has proved this to be unnecessary.

The above methods have been proven to satisfy the actual need. Follow Oracle best practices with respect to time synchronization regardless of the platform.

**High Availability and DRS Configuration**

One of the primary drivers for deploying Oracle RAC is the high availability provided by a RAC cluster. This cluster failover carries forward into a VMware vSphere environment and — with cluster nodes that can be migrated via vMotion — can now be configured to take advantage of these capabilities. Remember that VMware HA will — in the event a physical ESX host fails — automatically restart all of the failed host's VMs on surviving ESX hosts. VMs do experience down time when this happens. For this reason, allowing more than one virtualized RAC server node in a given RAC cluster to run on a single ESX host needlessly exposes the RAC cluster to failure scenarios from which it potentially may not recover gracefully.

As such, it is important to set a series of DRS anti-affinity policies between all nodes in a given RAC cluster. A typical virtualized Oracle RAC environment will consist of three server nodes. Since anti-affinity DRS policies can currently only be set between two specific VMs, multiple policies are required to keep three or more nodes in a RAC cluster properly separated. Be sure to name the DRS policies such that they can be easily identified and grouped together. Note that having multiple RAC nodes from different clusters running on the same host server is acceptable, subject to resource utilization and other resource management issues common to all virtual machines.

For optimal HA detection and monitoring, configure VM heartbeat monitoring for all nodes in the RAC cluster. This will ensure that, if VM is powered on, but not actually functioning, VMware HA will automatically restart the VM.

**Database Clustering Advances**

Thanks to the performance enhancements first introduced in VMware vSphere 4, it is now possible to cluster database systems reliably without the use of Raw Disk Mappings. This change enables individual nodes in a virtualized database cluster to migrate freely across ESX hosts in a HA/DRS cluster, and adds the benefits of database
clustering to those provided by vSphere. When configured this way, vSphere HA and DRS work to complement the inherent HA capabilities of Oracle RAC clusters.

vSphere DRS will ensure that all virtual Oracle RAC nodes receive the resources they require by dynamically load-balancing the nodes across the vSphere HA/DRS cluster. In the event any ESX host in the cluster fails (or RAC node when HA heartbeat monitoring is used), vSphere HA will automatically restart all failed RAC nodes on another available ESX host. The process of restarting these nodes will follow all HA and DRS rules in place to ensure that the failed nodes are placed on a host where no other nodes in the same RAC cluster are running. With this combination, Oracle RAC will automatically manage the loss of a failed node from an application perspective, and vSphere will then automatically recover the failed RAC node, restoring the Oracle RAC cluster’s state to normal. All of this occurs with no human intervention required.

The end result is that by using in-guest, iSCSI (and/or NFS) storage for shared data, virtualized Oracle RAC database clusters can achieve improved levels of redundancy and — on appropriate hardware infrastructure — enhanced levels of performance that cannot be achieved on physical infrastructure alone.

Footnote
1 Support Position for Oracle Products Running on VMware Virtualized Environments [ID 249212.1]), November 8, 2010.

About the Author
Christopher (Chris) Williams is a Senior Manager and Principal Architect in Consulting and Professional Services with Cognizant’s IT Infrastructure Services business unit, where he serves as the Lead Consultant in the Virtualization practice. Chris holds a Bachelor of Science degree from Metropolitan State College of Denver, and an MBA with an information systems emphasis from the University of Colorado. He can be reached at chris.williams@cognizant.com.

About Cognizant
Cognizant (Nasdaq: CTSH) is a leading provider of information technology, consulting, and business process outsourcing services, dedicated to helping the world’s leading companies build stronger businesses. Headquartered in Teaneck, N.J., Cognizant combines a passion for client satisfaction, technology innovation, deep industry and business process expertise and a global, collaborative workforce that embodies the future of work. With over 50 delivery centers worldwide and approximately 104,000 employees as of December 31, 2010, Cognizant is a member of the NASDAQ-100, the S&P 500, the Forbes Global 2000, and the Fortune 1000 and is ranked among the top performing and fastest growing companies in the world.

Visit us online at www.cognizant.com for more information.