

BEST PRACTICES

# Microsoft SQL Server and VMware Virtual Infrastructure



# Contents

|  |           |
|--|-----------|
| <b>Introduction.....</b>   | <b>1</b>  |
| Common Questions and Considerations.....                                     | 1         |
| Representative Customer Data.....  | 2         |
| <b>Characterizing SQL Server for Virtualization .....</b>                    | <b>3</b>  |
| Understanding Your Business and Technical Requirements.....                  | 3         |
| Understanding Availability Requirements .....                                | 4         |
| Understanding Other Deployment, Management, and Monitoring Requirements..... | 5         |
| Capturing Resource Utilization Baseline for Existing Databases .....         | 6         |
| Migrating to Virtual Environments.....                                       | 6         |
| <b>VMware Infrastructure for SQL Server — Best Practices .....</b>           | <b>6</b>  |
| General Guidelines .....   | 7         |
| CPU.....   | 8         |
| Memory .....   | 9         |
| Networking.....  | 10        |
| Storage.....   | 10        |
| <b>Availability Best Practices .....</b>                                     | <b>11</b> |
| <b>Deployment, Monitoring, and Management Best Practices.....</b>            | <b>14</b> |
| <b>Conclusion .....</b>  | <b>15</b> |
| <b>Resources .....</b>   | <b>16</b> |

## Introduction

Microsoft SQL Server is a very popular and widely deployed general purpose database server supported on Windows Server operating systems. Microsoft has consistently invested to ensure that SQL Server provides a comprehensive general purpose database platform that is competitive with Oracle, IBM DB2, and MySQL.

Because SQL Server is such a general purpose database server, applications that use SQL Server have very diverse resource requirements. It is very important that you clearly understand both the specific application requirements across important dimensions (including performance, capacity, headroom, and availability), and the virtualization context (including consolidation, scale-up, and scale-out models) when you deploy SQL Server on VMware® Infrastructure.

This best practices paper is for database administrators and IT administrators who are seriously considering virtualizing SQL Server and need to understand:

- How to characterize their SQL Server databases for virtualization,
- VMware Infrastructure best practices for SQL Server — best practice guidelines for ensuring your VMware Infrastructure platform is properly configured for running SQL Server, including guidance in the areas of CPU, memory, storage, and networking
- Deployment and operations — best practice guidelines for deploying SQL Server on VMware Infrastructure focusing on monitoring, management, backup, and related common operational best practices

The recommendations in this paper are not specific to any particular set of hardware or to the size and scope of the SQL Server implementation. The examples and considerations in this document provide guidance only and do not represent strict design requirements. VMware encourages you to use this guidance but also to play close attention to your specific workloads and deployment requirements.

To get the most out of this paper, you should have a basic knowledge of VMware Infrastructure 3 and recent versions of SQL Server (2005 or 2008).

## Common Questions and Considerations

As virtualization has become a standard practice within enterprises, many customers are now embarking on a “virtualize first” policy for all applications. The most common questions customers ask about virtualizing SQL Server are:

- How will SQL Server perform when virtualized?
- What are the best practices to virtualize SQL Server?

VMware and our industry partners have studied SQL Server virtualization in depth and have come to the following conclusions:

- SQL Server generally performs very well on VMware Infrastructure. This has been consistently demonstrated in a series of studies on scalability and performance from VMware, EMC, Dell and Brocade. For details, see “Virtualizing Microsoft SQL Server.” (For links to this and other documents mentioned in this paper, see the Resources section.)
- The vast majority of SQL Server deployments are good candidates for virtualization and can benefit from improvements offered by a virtualized infrastructure, such as improved management, availability, and scalability, thus reducing overall TCO.

- VMware Infrastructure makes it simpler and less expensive to provide higher levels of availability for SQL Server. VMware Distributed Resource Scheduler (DRS), VMware High Availability (HA), and the VMware Site Recovery Manager Plug-In (SRM plug-in) make it possible to eliminate planned downtime, reduce unplanned downtime, and recover rapidly from component, systematic, or sitewide outages.
- You can continue to leverage the most common SQL Server high availability implementations such as failover clustering, database mirroring, and log shipping in VMware Infrastructure 3.
- VMware Infrastructure features such as templates and cloning, together with VMware Lab Manager and VMware Stage Manager, make it easier to provision and stage SQL Server virtual machines into production deployments.
- VMware Consolidated Backup and vCenter Data Recovery can provide adequate backup functionality to protect SQL Server virtual machines in a VMware Infrastructure environment.

## Representative Customer Data

VMware has worked with many enterprise customers to understand how they use SQL Server in their IT environments. Virtualization assessments using VMware Capacity Planner showed the following key characteristics of SQL Server use in physical IT environments:

- SQL Server typically runs on two physical cores.
- Average CPU utilization was less than 6 percent, with 85 percent of the SQL Server hosts using less than 10 percent of CPU and 95 percent of hosts below 30 percent.
- On average, 3.1GB of memory was installed in a host, with 60 percent used.
- Average I/O was 20 IOPS, with 95 percent of hosts below 100 IOPS.
- Average network traffic was 400KB/sec or 3.2Mbps.

These relatively small, lightly loaded SQL Server databases represent a large consolidation opportunity. Such databases are regularly successfully migrated to virtual machines running on VMware ESX.

| CPU  | Total MHz | RAM MB Used | %CPU | Paging/Sec | Disk Busy | Disk Bytes | Disk IO/Sec |
|------|-----------|-------------|------|------------|-----------|------------|-------------|
| 2.00 | 4,812     | 123,112     | 6.21 | 21         | 5.76      | 634,039    | 33.00       |

**Table 1: Typical SQL Server utilization in physical environment (source: VMware Capacity Planner)**

Recent VMware customer surveys also indicate that:

- A very high percentage of VMware customers use VMware Infrastructure for virtualizing SQL Server in development and test environments.
- A significant and growing percentage of customers have successfully virtualized their production SQL Server instances.
- A significant percentage are strongly considering, evaluating, or going into production with virtualized SQL Server in 2009.
- A majority of customers prefer a homogeneous virtualization approach in which a given ESX host supports only multiple SQL Server virtual machines or instances — that is, they prefer not to mix SQL Server and other workloads on a given host.

## Characterizing SQL Server for Virtualization

When you consider virtualizing SQL Server deployments, two key concerns are:

- How to determine which SQL Server instances are good virtualization candidates
- What processes to use when migrating to a virtual environment.

To evaluate existing SQL Server deployments as candidates for virtualization, you need a clear understanding of your organization's business and technical requirements for each SQL Server instance. These requirements span several dimensions, such as availability, performance, scalability, growth and headroom, patching, and backups. Follow the steps outlined below to simplify the process of identifying SQL Server candidates for virtualization. The next several sections discuss approaches to:

- Understanding the business and technical requirements on various dimensions — such as database workload profiles, user base, response times, size of database, and growth factors —for your applications
- Understanding your availability requirements, including uptime guarantees (number of nines) and site recovery
- Understanding any other deployment and operational requirements
- Capturing the physical attributes of your running SQL Server instances over a period of time to establish baseline parameters
- Using the profiles and baselines and starting to map them onto a set of requirements for a VMware Infrastructure architecture

### Understanding Your Business and Technical Requirements

As a database, SQL Server supports a very wide variety of applications. One of the first tasks before deploying SQL Server in a virtual machine is to understand the business and technical requirements for SQL Server databases that support the applications in your environment. These databases differ significantly based on various factors such as mission-criticality, user count, database size, resource utilization, response time requirements, and workload profile. The databases often have unique deployment needs. You might use these requirements to define service level agreements (SLAs) and classify your databases into multiple tiers.

Mission critical databases (also referred to as tier-1 databases) are the ones you deem absolutely essential to your company's core operations. These databases and the applications they support often have SLAs that require very high availability and minimal yearly downtime, and they are often deployed on dedicated hardware.

Other databases and applications are busy only during specific periods for such tasks as reporting, batch jobs, and application integration or ETL workloads. These databases and applications might be essential to your company's operations, but they have much less stringent requirements for availability, scalability, number of users, and other similar factors. They may, nonetheless, have other very stringent business requirements, such as data validation and audit trails.

Still other smaller, lightly used databases, typically support departmental applications that may not adversely affect your company's real-time operations. You can tolerate such databases and applications being down for extended periods.

Database classification such as these often dictate system resource requirements. You should understand resource needs for SQL Server deployments in terms of the usual system utilization

metrics such as CPU, memory, and disk and network I/O, as well as user connections, transaction throughput, query execution efficiency and latencies, and database size. You might have established targets for system utilization on your SQL Server hosts, such as 70 percent CPU utilization, leaving enough headroom for any usage spikes. You must take this sizing and capacity planning into account along with plans for future growth when you design for deployments on VMware Infrastructure.

Understanding all these requirements helps you define appropriate virtual machine configurations for individual SQL Server databases. Because you consolidate multiple workloads on a single ESX host, this characterization also helps you design an ESX and storage hardware configuration that provides all the resources you need to deploy the multiple workloads successfully on VMware Infrastructure. Furthermore, this analysis also helps you identify how you can leverage advanced VMware Infrastructure features such as DRS to optimize overall resource utilization and virtual machine performance.

## Understanding Availability Requirements

Enterprises have various high availability requirements for database deployments, specifying such factors as uptime guarantees and site resilience. Uptime guarantees are measured in the number of nines — ranging from single nine (90 percent availability, or 21.6 hours daily) to five or more nines (99.999 percent availability, or approximately 5 minutes per year downtime). These high availability requirements determine the amount of downtime allowed in a particular deployment. Your deployment solution needs to address the applicable uptime guarantees, accounting for both planned and unplanned downtimes for various system and site components.

SQL Server also supports a variety of high availability features of its own to enable very high availability configurations. These features include log shipping, database mirroring with various levels of protection, and failover clustering using Microsoft Cluster Services (MSCS). Table 2 summarizes the level of high availability these features provide.

| Technology   | Granularity        | Data Loss                | Downtime          |
|--|--------------------|--------------------------|-------------------|
| Failover clustering  | Server             | No                       | ~30sec + recovery |
| Database mirroring<br>High safety mode with automatic failover | Server<br>Database | No                       | < 3secs           |
| Log shipping   | Database           | Possible transaction log | Variable          |
| Database mirroring<br>High protection configuration            | Database           | None                     | < 3secs           |
| Database mirroring<br>High performance configuration           | Database           | Some                     | < 3secs           |
| Replication  | Publication        | Some                     | Variable          |

**Table 2: Levels of high availability provided by key SQL Server features**

Traditional operating system clustering solutions that ensure higher levels of availability have been costly, hard to implement, and difficult to manage. Typically, only mission-critical databases that require three to five nines of availability implement such solutions. Where clustering is not

viable because of cost or complexity, databases that have high uptime requirements either rely on SQL Server high availability features or use other hardware-based availability measures.

Companies implementing disaster recovery solutions usually have strict hardware requirements at the disaster recovery site because of hardware and device driver compatibility issues. These companies also have comprehensive procedures and SLAs for bringing up business applications at the disaster recovery site.

Understanding these availability requirements helps you decide how you should leverage such advanced features of VMware Infrastructure as MSCS support, DRS, HA, and SRM for your SQL Server virtual machines, together with or instead of SQL Server high availability features.

## Understanding Other Deployment, Management, and Monitoring Requirements

SQL Server databases in an organization have unique deployment needs because of the variety of applications they serve. These applications can require 32-bit or 64-bit versions of the operating system and SQL Server, particular service packs or hot fixes, security and other access control settings, and support for specific legacy application components. When you consolidate such SQL Server databases in physical environments, these specific requirements often force you to take a least common denominator configuration approach, thus compromising optimal performance. Because a VMware Infrastructure environment enables you to run both 32-bit and 64-bit virtual machines side-by-side, and yet isolated from each other, virtualized consolidation is much more flexible and much less constrained. However, understanding these specific deployment requirements can help you refine virtualization approaches that work best. This understanding can help you decide whether to adopt a scale-up approach with multiple databases in a single large virtual machine or a scale-out approach with one or only a few databases per virtual machine. You can also identify the optimal workload, security, and other application-specific optimal configurations that need to be locked at the virtual machine level for your databases.

In planning a migration to a virtual environment, you should also understand other deployment and operational requirements, such as those for patching the operating system and SQL Server and those for rolling out and maintaining applications in test and development environments as well as production environments. Understanding these requirements can help you define how you should create, stage, and provision virtual machines in your datacenter, leveraging VMware Converter or the templates and cloning features in VMware Infrastructure.

Your options for backup and restore tools do not change for virtualized instances of SQL Server. However, because you may consolidate many SQL Server instances on a single ESX host, you need to plan carefully when you establish time windows for backing up and restoring data for different databases. Backup and restore for virtualized instances of SQL Server include:

- Use backup tools inside virtual machines.
- Use backup agents inside the virtual machines.
- Use backup agents in the ESX service console.

You can also use VMware Consolidated Backup or the snapshot capabilities provided by your storage vendor. When you understand the recovery point objectives and recovery time objectives for your SQL Server databases, you can define which of the above options is appropriate for your environment. ESX supports the Volume Shadow Copy Service (VSS) framework, so you can use a VSS-compliant solution to create application-consistent copies of data with any of these backup options.

## Capturing Resource Utilization Baseline for Existing Databases

After you clearly understand your organization's needs — business and technical requirements, availability and other operational requirements for implementing SQL Server— the next important step is to establish a baseline, using data from the current running physical deployment. The best way to accomplish this is to use the virtualization assessment exercise delivered by VMware or its partners. Using VMware Capacity Planner, this exercise collects all essential performance metrics — processor, disk, memory, and network statistics, including specific SQL Server metrics — on existing database servers in the environment. Capacity Planner analyzes this data to recommend server consolidation opportunities. See the appendix for a list of performance counters specific to SQL Server that VMware Capacity Planner collects. You can monitor these metrics, in addition to the essential system metrics, if you want to capture the baseline information yourself. It is important to collect the data over a period of time long enough to reflect all variations in the usage patterns of SQL Server in your organization. The duration can range from one week to one month, depending on seasonal or cyclical usage peaks — when they occur, their intensity, how long they last, and other factors. This exercise not only helps you understand what resources your current physical SQL Servers are using, it also makes it easier for you to create a short list of SQL Server instances to virtualize and to determine the order in which you should virtualize the SQL Server instances.

## Migrating to Virtual Environments

After you establish baseline profiles for your existing SQL Server databases, the next step is to design a VMware Infrastructure 3 architecture that meets these profiles. The best practices described in the next section help you ensure that your VMware Infrastructure environment is optimized for SQL Server deployment.

Microsoft offers extensive information on best practices for deploying SQL Server on Microsoft TechNet in the SQL Server TechNet Center. These best practices papers provide real-world guidelines and expert tips, and you should follow them for SQL Server deployments in a virtual environment. Similarly, VMware recommends following vendor-specific best practice guidelines for configuring your server hardware, storage subsystems, and network. In general, best practices in physical environments also apply to deployments on VMware Infrastructure without any changes.

Virtualizing SQL Server should provide benefits that go beyond server consolidation and lower total cost of ownership. A successful SQL Server deployment using VMware Infrastructure should provide better management and administration flexibility as well as will higher levels of availability at lower price.

## VMware Infrastructure for SQL Server — Best Practices

A variety of factors can affect SQL Server implementation on VMware Infrastructure 3, including processors and memory allocated to the virtual machine, storage layout and design, virtual machine placement, and application clustering methods, to name a few. This section defines best practices for designing VMware Infrastructure when virtualizing SQL Server environments. Starting with general guidelines, the section provides more specific details about CPU, memory, storage, and networking for ESX hosts and SQL Server virtual machines. The paper "Performance Best Practices and Benchmarking Guidelines" (see the References section for a link) outlines other performance-specific details.

## General Guidelines

Consider the following general guidelines when you plan your virtualized SQL Server deployment.

- **Use VMware Infrastructure** — VMware hosted products — VMware Server, VMware Workstation, and even VMware Fusion — are all capable of running operating systems that support SQL Server. However, VMware recommends using the VMware enterprise-class hypervisor — ESX or the free ESXi — to deploy any virtualized SQL Server instances, even for development and test environments. When you use the hosted products, depending on configurations and guest operating system support, the disk I/O caching performed by the host operating system can provide unpredictable performance and application availability in corner cases.
- **Use recent hardware** — x86-based servers and storage systems continue to become more capable than ever, all at a decreasing real cost. With support for dual, quad, six, and soon eight-core CPUs, larger processor caches, and up to 512GB of RAM, x86-based servers now rival mainframe systems. Similarly, support for the latest technologies, such as 8Gbps Fibre Channel connectivity, 10 GigE, jumbo frames, and Infiniband, have significantly increased storage performance. Furthermore, newer processors have faster implementations of hardware-assisted functions. Both Intel and AMD have made significant improvements in the performance of hardware assisted functions by reducing VMEXIT costs and implementing such new features as Intel's VMDq and AMD's RVI. All these hardware advances significantly help alleviate CPU, memory, and disk I/O bottlenecks when consolidating several SQL Server virtual machines on an ESX host. VMware ESX is the only hypervisor that can leverage these technologies and deliver returns on your investments.
- **Storage configuration is critical** to any successful database deployment, especially in virtual environments in which you may consolidate many different SQL Server workloads on a single ESX host. Your storage subsystem should provide sufficient I/O throughput as well as storage capacity to accommodate the cumulative needs of all virtual machines running on your ESX hosts. Most traditional physical SQL Server environments created many islands of information. When you move to virtualized SQL Server deployments, a shared storage model strategy provides many benefits, such as more effective storage resource utilization, reduced storage white space, better provisioning, in addition to mobility using VMware VMotion and VMware Storage VMotion. Most SQL Server performance issues in virtual environments can be traced to improper storage configuration. Microsoft also provides a Web page about best practices for SQL Server storage configuration, "Storage Top 10 Best Practices" (see Resources for a link). In addition, follow the storage best practices for VMware Infrastructure 3 outlined in this paper.
- **Follow SQL Server best practices** — Microsoft offers extensive best practices for deploying SQL Server on the "SQL Server Best Practices" page on Microsoft TechNet (see Resources for a link). These best practices papers provide real-world guidelines and expert tips, and you should follow them for SQL Server deployments in a virtual environment.
- **Size your ESX hosts with adequate capacity** to provide resources for all running virtual machines and allow enough headroom to account for normal workload variability and growth. This is especially important when multiple virtual machines on a host exhibit similar workload profiles and are thus likely to encounter bottlenecks on the same resources.

## CPU

The specifics of your SQL Server virtual machine configuration usually depend on the specific profile of your database. A thorough virtualization exercise, as recommended earlier in this paper, greatly simplifies sizing your virtual machines. In general, the following CPU guidelines apply.

- The number of virtual CPUs you need for a SQL Server virtual machine depends on the database workload you plan to host in that virtual machine. A thorough Capacity Planner exercise significantly eases this decision.
- A scale-out approach with multiple small SQL Server virtual machines is generally more appropriate for VMware Infrastructure. The scale-out approach also allows more granularity (for example, one SQL Server database per virtual machine) and makes it easier for you to customize the configuration as needed. Not only does this approach provide better workload and security isolation, it also allows easier maintenance and more granular change management. VMware DRS functions more effectively with smaller virtual machines, with the added benefit of faster VMotion migrations.

A scale-up approach, which involves a large virtual machine with multiple databases per SQL Server instance, usually exhibits higher performance overheads associated with the scaling of SMP virtual processors. The scale-up approach can also result in bottlenecks in the operating system itself, especially if you run into the operating system's 32-bit memory limits. Additionally, using larger virtual machines also makes it more challenging for VMware DRS to move the virtual machines around because moving them places greater demands placed on available resources.

- If you do not know the exact workload, begin tests in your test environment with a single virtual CPU. You can increase the number if necessary. Allocate multiple virtual CPUs to a virtual machine only if the anticipated SQL Server workload can truly take advantage of all the virtual CPUs. Overprovisioning virtual CPUs can result in higher virtualization overhead. VMware performance testing for SQL Server shows that even a virtual machine with a single virtual CPU can support high transaction throughput, at rates that are high enough for most SQL Server databases.
- Use the appropriate Windows server operating system to meet your requirements. Make sure the guest operating system is supported by ESX. When using Windows Server 2003, you can use either the uniprocessor (UP) hardware abstraction layer (HAL) or the symmetric multiprocessor (SMP) HAL. To avoid a need for updating the HAL when you add more processors, build your virtual machines with two vCPUs so you can install the SMP HAL, and then reduce the vCPU count to one vCPU for your initial configuration. You can add more vCPUs as needed. Windows Server 2008 dynamically adjusts the HAL to accommodate additional CPUs, so you do not need to update the HAL. The approach used in Windows Server 2008 provides more flexibility. However, when it is clear that only one vCPU is necessary and you do not need to make further changes, it is better to use the UP HAL, which can provide a small performance benefit.
- Install the latest version of VMware Tools in the guest operating system. Make sure to update VMware Tools after each ESX upgrade.
- It is important to evaluate the current SQL Server workloads in the physical environment before you take on any SQL Server virtualization deployments to ensure a successful SQL Server virtualization effort. Proper planning and sizing for capacity and growth are required to ensure that you meet or exceed any production level requirements or agreements currently in force in your organization.
- When you consolidate multiple virtual machines on single ESX host, proper hardware sizing is critical for optimal performance. Ensure that cumulative physical CPU resources on a server are adequate to meet the needs of the virtual machines by testing the workloads in the planned virtualized environment. ESX allows overcommitment of CPU resources, so the total number of virtual CPUs you configure across all virtual machines can be more than the actual number of physical CPU cores available on the host. CPU overcommitment should not adversely affect virtual machine performance.

as long as the virtual machines are underloaded compared to the compute resources and as long as the actual resource requirements of the virtual machines are less than the total CPU resources available on the system. For performance-critical SQL Server virtual machines (that is, production systems), ensure that sufficient CPU resources are available and that CPU overcommitment is not causing any performance degradation.

- ESX 3.x supports virtual machines with up to four virtual processors. If the Capacity Planner exercise indicates that a particular database needs more than four CPU cores, even on the newer hardware targeted to deploy ESX, that database may not be a suitable candidate for virtualization using ESX 3.x. The next release of ESX will alleviate this limitation by supporting up to eight virtual processors.
- To use VMotion for the deployment on VMware Infrastructure, ensure that you satisfy the CPU compatibility constraints that make VMotion possible only between certain revisions of CPUs. See “VMware VMotion and CPU Compatibility” (see the Resources section for a link).
- The ESX service console provides an execution environment to monitor and administer the entire ESX host. The service console is scheduled on CPU0 for ESX version 3.0.x. Under high loads, the service console competes with virtual machines that are scheduled on CPU0. VMware does not recommend running programs in the service console. The elimination of the service console in ESXi removes this contention.

## Memory

In general, the following guidelines apply to sizing memory for a SQL Server virtualization project.

- The amount of memory you need for a SQL Server virtual machine depends on the database workload you plan to host in that virtual machine. A thorough Capacity Planner exercise can easily help determine memory needed by a SQL Server virtual machine.
- ESX provides optimizations such as memory sharing and memory ballooning to reduce the amount of physical memory used on the underlying host. In some cases these optimizations can save more memory than is taken up by the virtualization overhead. VMware recommends that you enable both memory page sharing and memory ballooning.
- Database performance is heavily dependent on the amount of memory available. A common tuning technique is increasing the database buffer cache to reduce or avoid disk I/O and thus improve SQL Server performance. VMware Infrastructure 3 supports up to 64GB of memory per virtual machine. This enables you to configure SQL Server databases with large in-memory caches for better performance.
- As with CPU resources, ensure that cumulative physical memory available on a server is adequate to meet the needs of the virtual machines by testing target workloads in the virtualized environment. You can overcommit memory on ESX. Memory overcommitment should not adversely affect virtual machine performance as long as the actual virtual machine memory requirements are less than the total memory available on the system. For performance-critical SQL Server virtual machines (that is, production systems), ensure that sufficient memory resources are available and that memory overcommitment is not causing any performance degradation. If memory is overcommitted on the host, set reservations for performance-critical SQL Server virtual machines to guarantee that the memory for those virtual machines is not ballooned or swapped out. If you set the SQL Server `lock pages in memory` parameter, make sure to set the virtual machine's reservations to match the amount of memory you set in the virtual machine configuration. This setting can interfere with the ESX balloon driver. Setting reservations stops the balloon driver from inflating into the virtual machine's memory space.
- Both ESX and SQL Server support NUMA servers. As with SQL Server, ESX 3.5 has intelligent, adaptive NUMA scheduling and memory placement policies that can manage all virtual machines transparently, so you do not need to deal with the complexity of balancing virtual machines among nodes by hand.

To avoid any performance latency resulting from remote memory accesses, you should size a SQL Server virtual machine's memory so it is less than the amount available per NUMA node.

- ESX 3.5 supports 2MB memory pages (commonly referred to as large pages), giving the guest operating system the full advantage of using large pages. If your SQL Server deployment on a native system benefits from large memory pages, you can potentially achieve a similar performance improvement in a SQL Server virtual machine backed with 2MB machine memory pages. Using large pages reduces memory management overhead and can therefore increase hypervisor performance. See the paper "Large Page Performance" (see the Resources section for a link).

## Networking

In general, the following guidelines apply to provisioning the network for a SQL Server virtualization project.

- Use the NIC teaming and VLAN trunking support in ESX for better redundancy and isolation for SQL Server virtual machines.
- Use separate virtual switches, each connected to its own physical network adapter, to avoid contention between the ESX service console, the VMkernel, and virtual machines, especially virtual machines running heavy networking workloads.
- In addition to implementing separate virtual switches, use VLANs to separate production users, management, virtual machine network, and iSCSI storage traffic within the virtualized infrastructure. Separating these types of traffic ensures that management or iSCSI storage network traffic does not affect the production environment that the virtual machine is serving.
- Use the VMXNET network adapter for optimal performance. The Enhanced VMXNET driver also supports jumbo frames and TSO for better network performance. To use the Enhanced VMXNET device you must explicitly select Enhanced VMXNET on the VI Client hardware configuration page.
- Network communications between co-located virtual machines usually outperforms physical 1Gbps network speed, so if possible, place the various virtual machines that make up an application stack on the same ESX host. For details, see ""Networking Performance: VMware ESX Server 3.5"" (see the Resources section for a link).

## Storage

Storage configuration is critical to any successful database deployment. As with any physical SQL Server deployment, you should size the storage in virtualized environments properly to meet the database I/O requirements. Microsoft also offers predeployment storage best practices information to help you configure storage for SQL Server environments. Follow these guidelines for deployment on VMware Infrastructure as well. In addition, when you design the storage for a SQL Server virtual environment, consider the following factors:

- VMware Virtual Machine File System (VMFS) or raw device mapping (RDM) disks
- Fibre Channel or iSCSI storage protocol

This section discusses best practices for these storage-related factors.

- When you configure your VMware Infrastructure environment, you must decide whether to use RDM or VMFS partitions to store your virtual machines. The option that provides best performance depends on the workload.

VMware recommends VMFS for most virtual disk storage, because VMFS provides much more flexibility and in almost all cases performs just as well as RDMs. RDM and VMFS can provide similar transaction throughput. In most of the tests reported in "Performance Characterization of VMFS and RDM Using a

SAN," the I/O performance difference between RDM and VMFS configurations was insignificant. You should decide which storage configuration to use based on other aspects of your deployment. For example, you can move virtual machines using VMFS more easily than those using RDM, and VMFS provides greater flexibility. Using templates and self-provisioning is more complicated when you use RDM. An RDM configuration requires more interaction with the SAN management software to provision new virtual machines. In addition, you must deal with memory concerns when you have a large number of LUNs mapped to a single host. RDM may be desirable in a small number of cases, such as when using MSCS-based failover clustering for SQL Server. Using RDM may help you during the transition to a VMware Infrastructure environment. However, after you complete migration to VMware Infrastructure, VMware recommends moving to VMFS for easier storage administration.

- VMware has found that wire speed is the limiting factor for I/O throughput when comparing the Fibre Channel and iSCSI storage protocols. In the tests reported in "Comparison of Storage Protocol Performance" (see the Resources section for a link), ESX reached link speeds in a single virtual machine environment and also maintained the throughput for as many as 32 concurrent virtual machines for both Fibre Channel and iSCSI protocols. You can continue to use your existing storage connectivity infrastructure when deploying SQL Server in a VMware Infrastructure deployment.
- Aligning file system partitions is a well-known storage best practice for database workloads. Partition alignment on both physical machines and VMware VMFS partitions prevents performance I/O degradation caused by I/O crossing track boundaries. VMware test results detailed in "Recommendations for Aligning VMFS Partitions" (see the Resources section for a link) show that aligning VMFS partitions to 64KB track boundaries results in reduced latency and increased throughput. VMFS partitions created using the VI Client, which is part of VMware Infrastructure 3, are aligned on 64KB boundaries as recommended by storage and operating system vendors.
- Multiple heavily used virtual machines concurrently accessing the same VMFS volume, or multiple VMFS volumes backed by the same LUNs, can cause decreased storage performance. Appropriately configured storage architectures can avoid this issue. For details, see "Scalable Storage Performance" (see the Resources section for a link).
- Specifying a larger queue depth for some VMkernel drivers may yield higher performance. You can also adjust the maximum number of outstanding disk requests per virtual machine in the VMkernel through the VI Client. You can set this parameter to help equalize disk bandwidth across virtual machines.
- Monitor I/O latency statistics and any disk I/O queuing. Use `esxtop` (or `resxtop`) to see reports of device latency, time spent in the kernel, and latency seen by the virtual machine. You can also monitor I/O queuing in the VMkernel by checking the number of queued commands.
- ESX supports the latest technologies such as 8Gbps Fibre Channel connectivity and 10 GigE connectivity for IP-based storage.

## Availability Best Practices

VMware Infrastructure includes features to address both planned and unplanned downtimes caused by failures ranging from component problems to datacenter issues. This section provides best practices for using VMware HA, DRS, support for MSCS clustering, and the SRM plug-in, to implement SQL Server high availability in a VMware Infrastructure 3 environment.

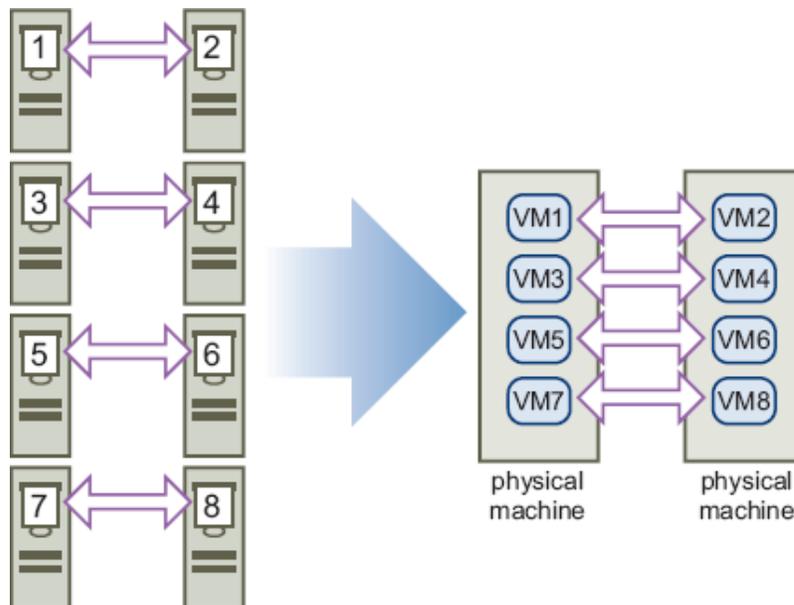
- An ESX host typically hosts several virtual machines and therefore needs to be reliable and highly available. To meet these requirements, you must configure your ESX hosts with adequate redundancy for various components such as power supplies, HBAs, network cards and system drives. You should also deploy redundant network and SAN switches.

VMware recommends setting up a minimum of four paths from an ESX host to a storage array, which means the host requires at least two HBA ports.

You can connect a single virtual switch to multiple, physical Ethernet adapters using the VMware Infrastructure feature called NIC teaming. This provides redundancy, aggregation, or both.

- In the physical server world, the use of traditional operating system based clustering solutions (like MSCS or VERITAS) is typically limited to the most mission critical systems (well under 10 percent) because of the costs and complexity involved. As a result, other essential systems in the environment have no a high availability solution. VMware HA can increase the baseline level of availability provided for SQL Server databases easily and cost effectively. It has typically proven to be more than sufficient for databases that require up to 99.99 percent high availability and is recommended for SQL Server databases that are not configured with other availability options.
- A VMware Infrastructure environment offers several possible failover clustering configurations: cluster in a box, cluster across boxes, and clustering physical machines with virtual machines. A cluster across boxes configuration provides both hardware- and software-level protection by placing the SQL Server cluster nodes on separate ESX hosts. VMware recommends this approach for production use.

You can expand the cluster across boxes model to place multiple virtual machines on multiple physical machines. For example, you can consolidate four clusters of two physical machines each to two physical machines with four virtual machines each, as shown in Figure 1.



**Figure 1: Consolidating clusters of physical machines to virtual machines**

- For SQL Server deployments using failover clustering, it is very important to use the `eagerzeroedthick` format when you create disks for virtual machines. By default, the VI Client and `vmkfstools` create disks in `zeroedthick` format. You can convert a disk to `eagerzeroedthick` format by importing, cloning, or inflating the disk. Disks you deploy from a template are also in `eagerzeroedthick` format.

Add virtual SCSI adapters and disks to the virtual machine configuration before you add networking. See the VMware knowledge base article "Adding a Virtual Disk on an Auxiliary Virtual SCSI Controller to a Virtual Machine Moves Other Devices to New PCI Slots" (see the Resources section for a link). When you implement Microsoft clustering technologies on VMware Infrastructure 3, VMware recommends that you use RDM in physical compatibility mode for shared storage. You must set SCSI bus sharing mode to **Physical**. It is also important to use the same target ID on the virtual SCSI adapter for the

same clustered disk for all cluster nodes. You must use a separate virtual SCSI adapter for clustered disks.

VMware Infrastructure 3 does not allow clustered virtual machines to be part of VMware clusters (DRS or HA). Also, you cannot use VMotion to migrate virtual machines that deploy MSCS software.

Given the complexity involved in deploying failover clustering, VMware recommends testing clustered configurations in various failover scenarios before staging them into production environments. For details, see “Setup for Microsoft Cluster Service” (see the Resources section for a link).

- Using VMware Infrastructure 3 with DRS can provide more efficient distribution of hardware resources for SQL Server virtual machines. DRS helps allocate computing resources more efficiently across the entire platform. DRS provides both proportional and absolute mechanisms to control priorities for various SQL Server workloads by using shares, reservations, and limits for CPU and memory, shares for virtual disks, and traffic shaping for the network. For example, in SQL Server environments virtualized using VMware Infrastructure 3, you can use these features to favor OLTP or DSS workloads as needed. During the batch or backup windows, you can set high shares for a specific database and low shares for OLTP workloads likely to be least used at that time.
- Virtual machines with less memory or fewer vCPUs provide more opportunities for DRS to migrate them in order to improve balance across the cluster. Virtual machines with more memory or more vCPUs add more constraints in migrating the virtual machines. These DRS considerations add one more reason to configure virtual machines with only as many vCPUs and only as much virtual memory as they need.
- DRS affinity rules can keep virtual machines on the same ESX host or make sure they are always on different hosts. For example, co-locating multitier components of an application stack can benefit from in-memory network transfers faster than 1Gbps. Conversely, you can use an anti-affinity rule to make sure that the SQL Server nodes in a failover cluster do not run on single ESX host.  
For details, see “DRS Performance and Best Practices” (see the Resources section for a link).
- SQL Server deployments in VMware Infrastructure 3 can use all the SQL Server high availability features — log shipping, database mirroring, and failover clustering — that you can deploy in physical environments. To achieve the high availability benefits, it is important to place the virtual machine with the secondary or mirror database and the witness virtual machine are on different ESX hosts. You can use the DRS anti-affinity rule to address this.
- Most disaster recovery solutions require you to maintain identical server hardware at the disaster recovery site to prevent issues with hardware and device driver compatibility when servers are brought online at the recovery site. VMware Infrastructure using the SRM plug-in eliminates this requirement, and you can quickly bring systems online on any supported ESX host machines running anywhere in the world. Because each virtual machine is encapsulated in a distinct set of files, replication to a disaster recovery site is simplified and systems can be back up and running quickly. Moving a running server from one place to another can be as easy as copying a set of files. VMware strongly recommends using the SRM plug-in for such mission critical applications as SQL Server databases.

## Deployment, Monitoring, and Management Best Practices

This section provides best practice guidelines focusing on creating SQL Server virtual machines, migrating to VMware Infrastructure 3, ongoing performance monitoring and management, backup, and related common operational best practices for deploying SQL Server on VMware Infrastructure.

- You should plan your deployment on VMware Infrastructure by creating a team that represents the users, the applications team, the development team, and the IT support team. Your team should walk through the processes of testing, verifying results, and eventually migrating to the VMware Infrastructure environment.
- Start the migration to a virtual environment by rigorous piloting and testing, by virtualizing a small set of SQL Server instances at a time. Perform regular data backups, and run normal IT processes on these virtualized SQL Server instances to validate that the deployment can handle not only the user load but also the additional day-to-day monitoring and operating load. As your comfort level increases, increase the pilot to include additional SQL Server instances and more challenging database workloads.
- When you create the SQL Server virtual machine, you can simplify the physical to virtual migration using VMware Converter, which can both hot and cold clone an existing physical SQL Server instance. Physical servers often have services and processes, such as SNMP monitoring and backup agents, that may be running in the background. When creating virtual machines using VMware Converter, be careful about such processes. These processes can add significant virtualization overhead and seriously degrade SQL Server performance in a virtual environment. Monitor any extra processes running in a virtual machine and stop them.

When creating new virtual machines, take advantage of VMware Infrastructure 3 templates and cloning features to build a set of common server images that you can easily replicate and update, reducing the time you need to provision and deploy new servers. The traditional method of deploying a new physical server generally takes weeks in many organizations. With VMware Infrastructure, you can deploy new servers in minutes from a template.

- To further ease the transition to a virtual environment, configure SQL Server virtual machines with RDM in physical mode that point to existing data, transaction logs, and `tempdb` LUNs. Although VMware recommends a full transition to VMFS partitions for maximum flexibility, starting with RDM means one less variable to worry about during the migration project.
- Use realistic workloads for the test environment. Stress testing the deployments on VMware Infrastructure with no “think times,” for example, or comparing physical to virtual performance from different physical servers can be misleading. Evaluate the proposed workloads to gain an understanding of how current physical resource utilization peaks and heterogeneous workloads will affect resource utilization when these workloads are virtualized.
- Use VMware and partner guides for sizing your storage and network. In most cases, you can implement the best practices for physical environments in your deployments on VMware Infrastructure without any changes.
- Traditional SQL Server performance monitoring uses the Microsoft Windows performance monitor tool PerfMon to collect statistics. SQL Server integrates with PerfMon to provide familiar counters that indicate system performance. However, as with all measurement tools run in the guest operating system, time-based performance measurements are subject to error. The degree to which the measurements are inaccurate depends on the total load of the ESX host. Always use `VirtualCenter`, `esxtop`, or `resxtop` to measure resource utilization. Depending on the load placed on the ESX host, CPU and memory usage reported within the guest operating system can be different from what ESX reports.

Pay close attention to the following VMware Infrastructure 3 counters, referring to online documentation on performance monitoring and analysis for more information on these counters and how to interpret them.

| Subsystem | esxtop Counter               | VirtualCenter Counter  |
|-----------|------------------------------|--|
| CPU       | %RDY<br>%USED                | Ready<br>Usage   |
| Memory    | %ACTV<br>SWW/s<br>SWR/s      | Active<br>Swapin<br>Swapout  |
| Storage   | ACTV<br>DAVG/cmd<br>KAVG/cmd | Commands<br>deviceWriteLatency and deviceReadLatency<br>kernelWriteLatency and kernelReadLatency |
| Network   | MbRX/s<br>MbTX/s             | packetsRx<br>packetsTx   |

**Table 3: VMware Infrastructure 3 Performance Counters of Interest to SQL Administrators**

Table 3 lists a few key counters you should add to your list of inspection points. Of the CPU counters, the total used time indicates system load. Ready time indicates overloaded CPU resources. A significant swap rate in the memory counters is a clear indication of a shortage of memory, and high device latencies in the storage section point to an overloaded or misconfigured array. Network traffic is not frequently the cause of SQL Server performance problems except when large amounts of iSCSI storage traffic are using a single network line. Check total throughput on the NICs to see if the network is saturated on the ESX host.

## Conclusion

You can virtualize SQL Server successfully using VMware Infrastructure. The vast majority of SQL Server deployments are good candidates for virtualization and can benefit from improvements offered by a virtualized infrastructure — such as improved management, availability, and scalability — thus reducing overall TCO.

VMware Infrastructure makes it simpler and less expensive to provide higher levels of availability for SQL Server. Taking advantage of such key VMware Infrastructure 3 features as DRS, HA, and the SRM plug-in, you can eliminate planned downtime, reduce unplanned downtime, and recover rapidly from component, systematic, or sitewide outages. You can continue to leverage the most common SQL Server high availability implementations, such as failover clustering, database mirroring, and log shipping in VMware Infrastructure.

To deploy SQL Server successfully in VMware Infrastructure, you must clearly understand your organization's needs — business and technical requirements, availability, and other operational requirements for implementing SQL Server. You must establish a baseline, using data from the current running physical deployment. Using this baseline, you can create a short list of SQL Server instances to virtualize and determine the order in which you should virtualize the SQL Server instances.

It is critical that you follow best practice guidelines specific to Microsoft SQL Server and other vendor products in your environment as well as VMware Infrastructure best practice guidelines. In

general, best practices in physical environments also apply to deployments on VMware Infrastructure without any changes.

Storage configuration is critical to any successful database deployment, especially in virtual environments in which you may consolidate many different SQL Server workloads on a single ESX host. Most SQL Server performance issues in virtual environments can be traced to improper storage configuration.

## Resources

- “Adding a Virtual Disk on an Auxiliary Virtual SCSI Controller to a Virtual Machine Moves Other Devices to New PCI Slots”  
<http://kb.vmware.com/kb/1513>
- “Benchmarking Microsoft SQL Server Using VMware ESX Server 3.5”  
[http://www.vmware.com/files/pdf/benchmarking\\_microsoft\\_sql\\_vmware\\_esx\\_server\\_wp.pdf](http://www.vmware.com/files/pdf/benchmarking_microsoft_sql_vmware_esx_server_wp.pdf)
- “Comparison of Storage Protocol Performance”  
[http://www.vmware.com/files/pdf/storage\\_protocol\\_perf.pdf](http://www.vmware.com/files/pdf/storage_protocol_perf.pdf)
- “DRS Performance and Best Practices”  
[http://www.vmware.com/files/pdf/drs\\_performance\\_best\\_practices\\_wp.pdf](http://www.vmware.com/files/pdf/drs_performance_best_practices_wp.pdf)
- “Large Page Performance”  
[http://www.vmware.com/files/pdf/large\\_pg\\_performance.pdf](http://www.vmware.com/files/pdf/large_pg_performance.pdf)
- “Microsoft SQL Server 2005 Virtualization in the Dell Scalable Enterprise”  
<http://www.dell.com/downloads/global/power/ps4q06-20060405-Muirhead.pdf>
- “Networking Performance: **VMware ESX Server 3.5**”  
[http://www.vmware.com/files/pdf/ESX\\_networking\\_performance.pdf](http://www.vmware.com/files/pdf/ESX_networking_performance.pdf)
- “Performance Best Practices and Benchmarking Guidelines”  
[http://www.vmware.com/pdf/VI3.5\\_Performance.pdf](http://www.vmware.com/pdf/VI3.5_Performance.pdf)
- “Performance Characterization of VMFS and RDM Using a SAN”  
[http://www.vmware.com/files/pdf/performance\\_char\\_vmfs\\_rdm.pdf](http://www.vmware.com/files/pdf/performance_char_vmfs_rdm.pdf)
- “Recommendations for Aligning VMFS Partitions”  
<http://www.vmware.com/vmtn/resources/608>
- “Scalable Storage Performance”  
<http://www.vmware.com/resources/techresources/1059>
- “Setup for Microsoft Cluster Service”  
[http://www.vmware.com/pdf/vi3\\_35/esx\\_3/r35u2/vi3\\_35\\_25\\_u2\\_mscs.pdf](http://www.vmware.com/pdf/vi3_35/esx_3/r35u2/vi3_35_25_u2_mscs.pdf)
- “SQL Server Best Practices”  
<http://technet.microsoft.com/en-us/sqlserver/bb671430.aspx>
- “SQL Server Workload Consolidation”  
[http://www.vmware.com/pdf/SQL\\_Server\\_consolidation.pdf](http://www.vmware.com/pdf/SQL_Server_consolidation.pdf)
- “SQL Server Performance in a VMware Infrastructure 3 Environment”  
<http://www.vmware.com/files/pdf/SQLServerWorkloads.pdf>
- “Storage Top 10 Best Practices”  
<http://www.microsoft.com/technet/prodtechnol/sql/bestpractice/storage-top-10.msp>
- “Virtualizing Microsoft SQL Server”  
<http://www.vmware.com/solutions/business-critical-apps/sql/>

- "VMware VMotion and CPU Compatibility"  
<http://www.vmware.com/resources/techresources/1022>
- "VMware VMotion Performance on the Dell PowerEdge 1955 Blade Server"  
[http://www.dell.com/downloads/global/solutions/vmware\\_1955.pdf](http://www.dell.com/downloads/global/solutions/vmware_1955.pdf)

## Appendix: SQL Server Specific Performance Counters Collected by VMware Capacity Planner

A virtualization assessment with VMware Capacity Planner collects information on a wide range of performance counters, including the following that are of special interest in planning a SQL Server virtualization project.

### SQL Server:Access Methods

| Counter                    | Description   |
|----------------------------|---|
| Full scans/sec             | Number of unrestricted full scans. These can be either base table or full index scans.  |
| Index searches/sec         | Number of index searches. Index searches are used to start range scans and single index record fetches and to reposition within an index. |
| Page splits/sec            | Number of page splits occurring as the result of index pages overflowing.   |
| Table lock escalations/sec | Number of locks escalated per second.   |

### SQL Server:Buffer Manager

| Counter                | Description  |
|------------------------|--|
| Buffer cache hit ratio | Percentage of pages found in the buffer pool, thus saving a read from disk.  |
| Free pages             | Number of pages on all free lists.   |
| Lazy writes/sec        | Number of buffers written by buffer manager's lazy writer.   |
| Page life expectancy   | Number of seconds a page stays in the buffer pool without references.  |
| Procedure cache pages  | Number of pages used to store compiled queries.  |
| Read-ahead pages/sec   | When SQL Server detects sequential access of data or index pages, it reads additional pages in anticipation of a read request. Instead of reading one 8KB page at a time, SQL Server reads 64KB of pages with a single request. This improves the execution time of the process performing the sequential scan. These pages become resident in the buffer cache. |
| Stolen pages           | Number of pages used for miscellaneous server purposes (including procedure cache).  |
| Total pages            | Number of pages in the buffer pool (includes database free and stolen).  |

## SQL Server:Databases

| Counter                | Description  |
|------------------------|--|
| Data file(s) size (KB) | Cumulative size of all the data files in the database. |
| Log file(s) size (KB)  | Cumulative size of all the log files in the database.  |
| Transactions/sec       | Number of transactions started for the database.       |

## SQL Server:General Statistics

| Counter          | Description                              |
|------------------|--|
| User connections | Number of users connected to the system. |

## SQL Server:Locks

| Counter                 | Description   |
|-------------------------|---|
| Average wait time (ms)  | The average amount of wait time (milliseconds) for each lock request that resulted in a wait. |
| Number of deadlocks/sec | Number of lock requests that resulted in a deadlock.  |

## SQL Server:Memory Manager

| Counter                       | Description   |
|-------------------------------|---|
| Connection memory (KB)        | Total amount of dynamic memory the server is using for maintaining connections  |
| Granted workspace memory (KB) | Total amount of memory granted to executing processes. This memory is used for hash sort and create index operations. |
| Lock memory (KB)              | Total amount of dynamic memory the server is using for locks.   |
| SQL cache memory (KB)         | Total amount of dynamic memory the server is using for the dynamic SQL cache.   |
| Target server memory (KB)     | Total amount of dynamic memory the server is willing to consume.  |
| Total server memory (KB)      | Total amount of dynamic memory the server is currently consuming.   |

## SQL Server:SQL Statistics

| Counter                | Description                                      |
|------------------------|--|
| Batch requests/sec     | Number of SQL batch requests received by server. |
| SQL compilations/sec   | Number of SQL compilations.                      |
| SQL recompilations/sec | Number of SQL recompilations.                    |

Revision: 20090313 Item: WP-087-ISV-01-01



**VMware, Inc. 3401 Hillview Ave. Palo Alto CA 94304 USA Tel 650-475-5000 Fax 650-475-5001 [www.vmware.com](http://www.vmware.com)**  
© 2009 VMware, Inc. All rights reserved. Protected by one or more of U.S. Patent Nos. 6,397,242, 6,496,847, 6,704,925, 6,711,672, 6,725,289, 6,735,601, 6,785,886, 6,789,156, 6,795,966, 6,880,022, 6,944,699, 6,961,806, 6,961,941, 7,069,413, 7,082,598, 7,089,377, 7,111,086, 7,111,145, 7,117,481, 7,149,843, 7,155,558, 7,222,221, 7,260,815, 7,260,820, 7,269,683, 7,275,136, 7,277,998, 7,277,999, 7,278,030, 7,281,102, 7,290,253, 7,356,679, 7,409,487, 7,412,492, 7,412,702, 7,424,710, 7,428,636, 7,433,951, 7,434,002, and 7,447,854; patents pending. VMware, the VMware "boxes" logo and design, Virtual SMP and VMotion are registered trademarks or trademarks 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.

