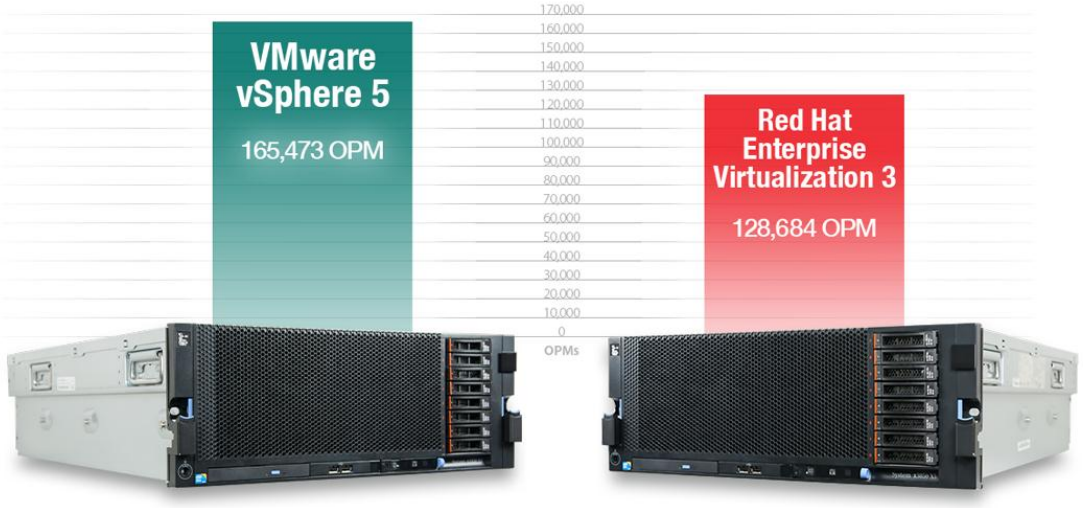
VIRTUALIZATION PERFORMANCE: VMWARE VSPHERE 5 VS. RED HAT ENTERPRISE VIRTUALIZATION 3

VMware® vSphere® 5 outperformed Red Hat® Enterprise Virtualization 3 by up to 28.6%



on a highly utilized host with 42 virtual machines running a database workload

When you invest in a virtualization platform, you can maximize the performance of your applications and the overall infrastructure by fully utilizing physical resources as much as a hypervisor allows. Planning for scenarios with greater virtual machine (VM) densities, such as during maintenance periods or during high availability (HA) events where a server must host additional VMs from a failed server, is critical to an organization's overall IT strategy. A hypervisor that excels at resource management allows for greater virtual machine (VM) density and better application performance.

We tested two hypervisors—VMware vSphere 5, and Red Hat Enterprise Virtualization 3 (RHEV)—to compare their performance and ability to manage resources at high levels of RAM utilization. When running a heavily loaded host with 39 virtual machines, VMware vSphere 5 outperformed RHEV 3 by 16.2 percent; after adding a few more virtual machines to the host, VMware's advantage increased to 28.6 percent. Additionally, VMware vSphere 5 continued to scale from 39 VMs up to 42 VMs with our workload: Overall server performance increased by 2.8 percent with VMware vSphere 5, whereas performance *decreased* by 7.2 percent with RHEV 3.

These results show that VMware vSphere 5 can deliver superior performance in a densely virtualized environment over RHEV 3. When you fully utilize your servers, you need fewer systems to perform the same amount of work during normal operations, and you do not need excessive server capacity to handle workload peaks. This results in overall savings for your organization.

ADVANCED RESOURCE MANAGEMENT MAXIMIZES PERFORMANCE

A hypervisor with exceptional resource management lets you optimize virtual machine performance across your entire infrastructure. The result? Greater density, scalability, and performance. VMware offers several unique features that enable vSphere 5 to utilize system resources better than competing hypervisors.

Direct driver model. The VMware approach is to install device drivers directly onto the hypervisor, effectively making the hypervisor an intermediary between the physical hardware and VMs that run on the server. The direct driver model improves performance and scalability as the number of VMs on a host increases.

High-performance "gang" scheduler. This feature allows VMware vSphere 5 to handle the challenging CPU and I/O needs of VMs. vSphere 5 is thus able to allocate resources and processor time slices to the VMs that most need it.

How VMware manages memory

Additional VMware technologies allow vSphere 5 to optimize physical memory allocation, dynamically shifting this critical resource from less active VMs to VMs that are more active. This is accommodated in vSphere by the following features, working in concert:

Transparent page sharing. Transparent page sharing (TPS) identifies common pages across VMs and stores each in physical memory only once. This is somewhat analogous to deduplication technologies used in storage implementations. All VMs then share only that single copy of the memory page. VMware vSphere 5 determines sharable pages by scanning the content of the virtual machines' physical memory for sharing opportunities. By eliminating these redundant copies, VMware vSphere 5 frees up memory for use by workloads.

Memory ballooning. When the hypervisor needs to give more memory to VMs that are powering on or experiencing a heavy workload, it "asks" the guest operating systems in other VMs to provide memory to a balloon process that runs in the guest. The hypervisor can then temporarily lend that "ballooned" memory to the busy VMs. When the busy VMs no longer need the extra memory, the hypervisor "deflates" the balloon, returning that memory to the original guest OS.

Memory compression. The innovative memory compression capability in VMware vSphere 5 sets aside a small portion of physical RAM as a cache. Compressing unused memory pages avoids hypervisor swapping and is orders of magnitude faster than disk.

Hypervisor swap. If a system's memory resources are experiencing intense pressure, hypervisor swap acts as a safety valve, ensuring reliable operation of the host and all workloads. While this may result in a short-term performance hit, it offers the

hypervisor another option to resolve memory issues. Furthermore, a new feature in vSphere 5, called swap to host cache, can use solid-state disks for swap purposes, reducing the impact on performance. However, we did not use this feature for this testing.

DRS with resource pools. This feature is a safety net of sorts, largely because it ensures that applications receive the resources they need when they need them. It accomplishes this by dynamically load balancing resources throughout a cluster of VMs. This does not apply to standalone hosts such as the one tested for this report, but to vSphere clusters. Using a vSphere-clustered environment with DRS ensures optimization of resources and the ability to accommodate shifting workloads.

PUTTING THE HYPERVISORS TO THE TEST

To compare these two hypervisors, we ran two scenarios on each hypervisor to demonstrate different levels of memory usage. In the first scenario, we ran database workloads against a heavily loaded host, and in the second scenario, we ran the same database workloads and increased the VM count. We first ran these scenarios with our server running VMware vSphere 5 and then ran the same scenarios with our server running Red Hat Enterprise Virtualization 3. Our four-socket server had 80 logical CPUs and 256 GB of RAM. We configured each VM with two virtual CPUs and 8 GB of RAM. For the first scenario, we ran 39 VMs to mimic a highly utilized server scenario. In the second scenario, we ran 42 VMs, taking the utilization higher. While RAM was fully allocated and utilized, CPU utilization was 46 and 51 percent for the two density scenarios on RHEV, and 39 and 41 percent for the two density scenarios on VMware vSphere.

In both scenarios, two-thirds of our guests ran Microsoft® Windows® Server 2008 R2 SP1 with Microsoft SQL Server® 2008 R2 SP1, and one-third of the guests ran Red Hat Enterprise Linux® 6.2 with PostgreSQL 9.1. Independent of platform, each guest VM hosted an 8GB database created with the DVD Store Version 2 (DS2) benchmark tool. We used clients to invoke the DS2 workload on each guest VM using identical workload parameters. DS2 simulates an online store, and reports orders per minute (OPM). We chose this database workload for our testing as it heavily utilizes processor, memory, and I/O to create a broad and demanding load on the system.

For more details about the DS2 tool, see http://www.delltechcenter.com/page/DVD+Store. For step-by-step instructions on how we tested, see appendices to this report.

VMWARE VSPHERE 5 DELIVERS

As Figure 1 shows, aggregate performance across all VMs in our 39-VM scenario with VMware vSphere 5 was 161,040 OPM, 16 percent higher than RHEV 3, where the total performance of all 39 VMs was 138,603 OPM.

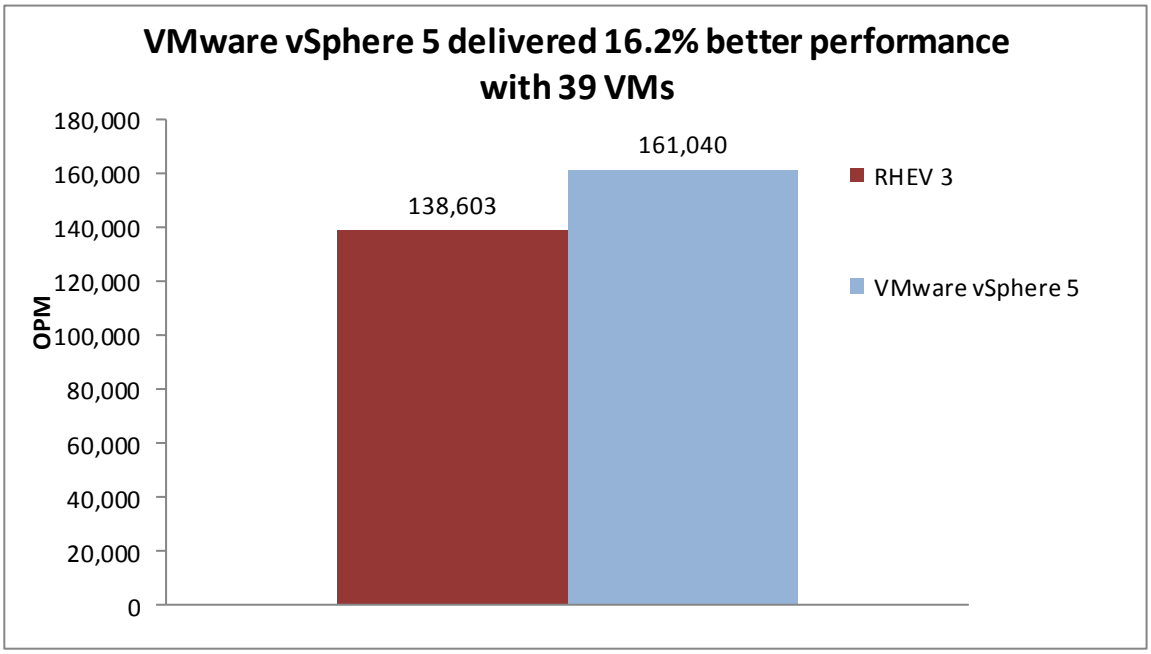


Figure 1. VMware vSphere 5 delivered 16.2 percent better overall performance with 39 VMs.

After increasing the VM density to 42 VMs on our single host, we ran the same test again. As Figure 2 shows, the performance advantage of VMware vSphere 5 increases at the higher density. Aggregate performance across all VMs in our 42-VM scenario with VMware vSphere 5 was 165,473 OPM, 28 percent higher than on the server running RHEV 3, where the total performance of all 42 VMs was 128,684 OPM.

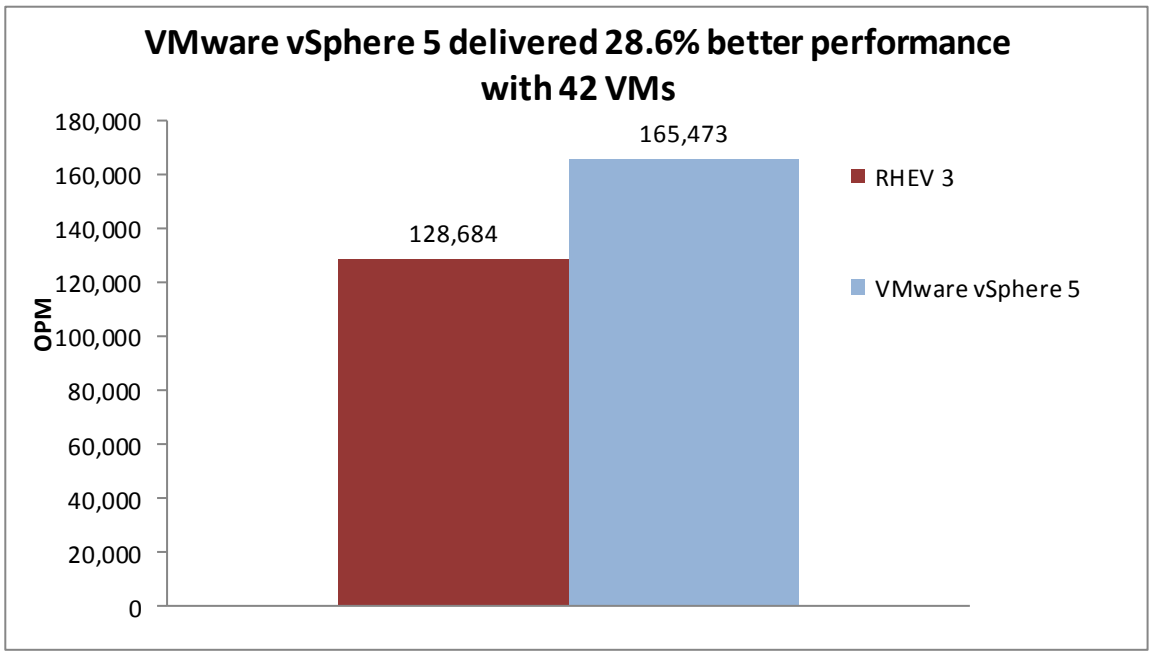


Figure 2. VMware vSphere 5 delivered 28.6 percent better overall performance with 42 VMs.

As the above results show, and as Figure 3 shows, VMware vSphere 5 better handled the increase in density from 39 to 42 VMs, increasing in total output by almost 3 percent. RHEV 3, however, decreased in performance by over 7 percent at the higher density.

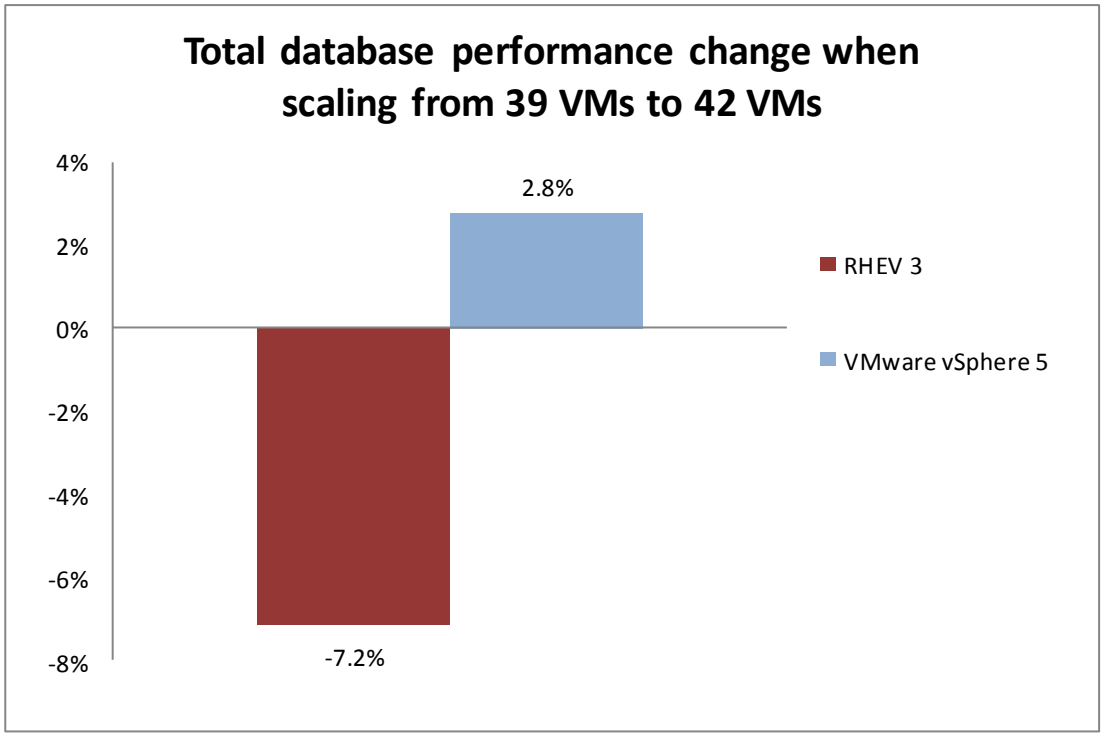


Figure 3. VMware vSphere 5 increases in output with an increase in density, while RHEV 3 performance degraded.

CONCLUSION

Using a hypervisor that offers better resource management and scalability can deliver excellent virtual machine performance on your servers. In our testing, VMware vSphere 5 allowed our host's virtual machines to outperform those running on RHEV 3 by over 28 percent in total OPM performance. Furthermore, VMware vSphere 5 performance continued to improve when going from 39 VMs to 42 VMs: Total performance for VMware vSphere 5 increased by 2.8 percent, whereas it decreased by 7.2 percent with RHEV 3.

With the capabilities and scalability that VMware vSphere 5 offers, you are able to utilize the full capacity of your servers with confidence and purchase fewer servers to handle workload spikes; this can translate to fewer racks in the data center, lower costs for your business, and more consistent overall application performance.

APPENDIX A – TEST CONFIGURATION OVERVIEW

We used 22 client machines for both test scenarios, with each client targeting one or two VMs on the server under test depending on the specific scenario. We first tested the VMs on VMware vSphere 5, and then on Red Hat Enterprise Virtualization Hypervisor 6.2. We ran both environments, one by one, on the same hardware.

We took advantage of the ability of DVD Store 2.1 to target multiple targets with each instance targeting one or two VMs. Each client machine was a VM running Windows Server 2003 Enterprise R2 SP2 with the following:

- A client share folder with the latest ds2postgresqlserverdriver.exe, ds2sqlserverdriver.exe, Mono.Security.dll, Npgsql.dll, and the necessary run scripts
- .NET framework 3.5 SP1, required for the DS2 SQL Server driver
- .NET framework 4, required for the DS2 PostgreSQL driver and Npqsql .Net Data Provider.

We ran one physical workstation class client machine to ensure no performance differences between running virtual and physical clients.

As Figure 4 shows, we cabled all four of the client host's 1Gb NICs to a Dell™ PowerConnect™ 6248 switch. We then created four virtual machine networks and dedicated each physical NIC to its own virtual machine network. We used a round robin approach to distribute the virtual NICs amongst the VMs.

On the server under test, we attached all six NICs in the IBM® System x3850 X5 to the same PowerConnect 6248 switch, using the two onboard NICs for management traffic and a quad-port Intel® I340-T4 Ethernet server adapter for VM traffic. We initiated and controlled all testing sessions from a separate controller machine.

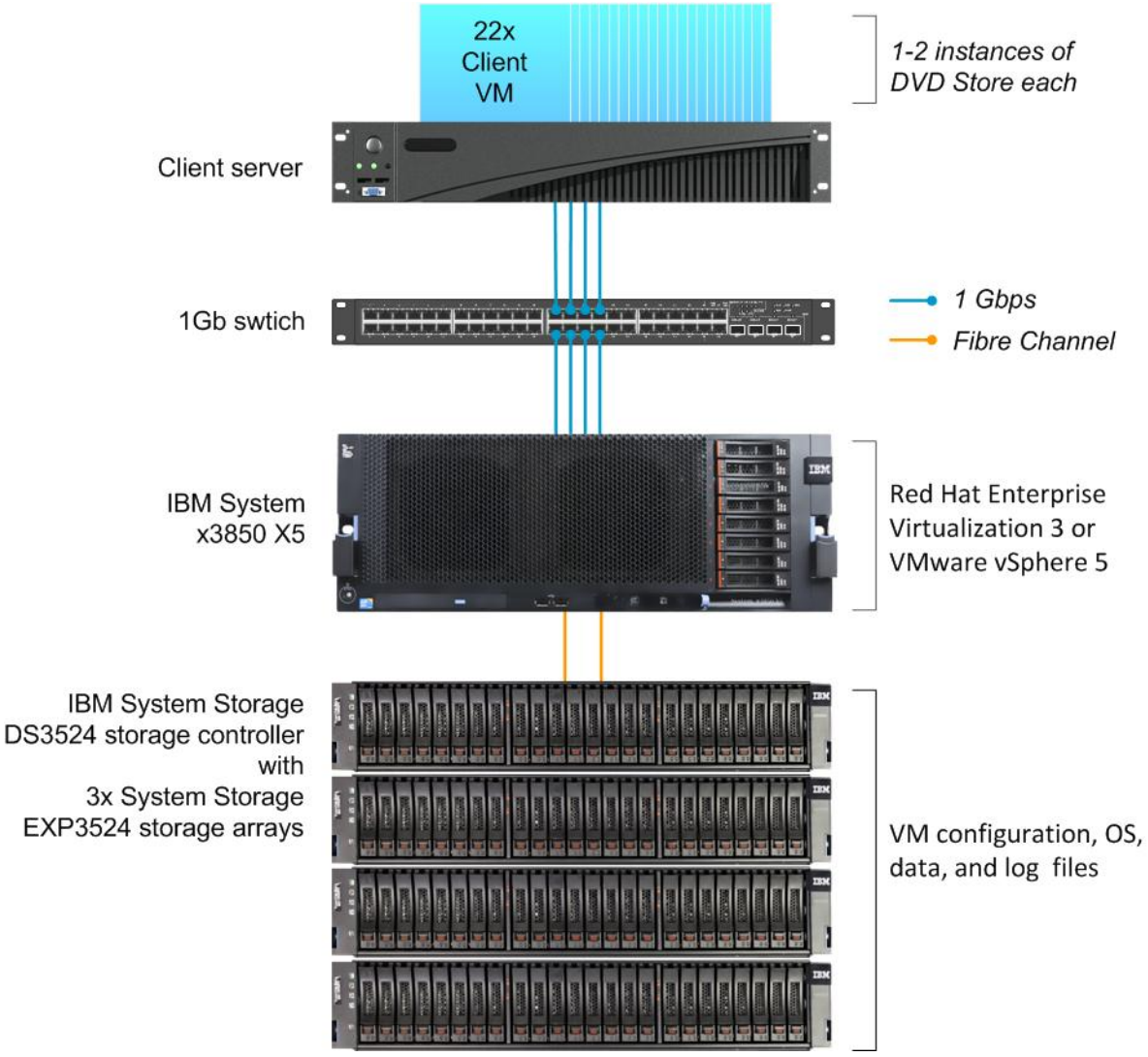


Figure 4. The test bed layout we used for both the VMware vSphere 5 and Red Hat Enterprise Virtualization 3 testing.

APPENDIX B – SETTING UP THE STORAGE

IBM System x3850 X5 and IBM System Storage DS3524 Express storage configuration overview

Our complete storage infrastructure consisted of four internal drives in the server, an IBM ServeRAID internal RAID controller, two Emulex® LightPath LPe12000 single port 8Gb/s fibre cards attached directly to the IBM System Storage DS3524, and one IBM System Storage DS3524 array with three IBM System Storage EXP3524 expansion trays. Figure 4 on the previous page shows the complete layout of our test bed hardware.

We configured the internal drives on the server in two RAID 1 volumes, each volume containing two physical disks. We dedicated one volume to the vSphere partition, and the other to the Red Hat Enterprise Virtualization Hypervisor partition. To switch between environments during testing, we toggled the assigned boot volume in the RAID controller LSI WebBIOS and enabled the Legacy Only boot option in the system BIOS (applicable to RHEV-H only).

For external storage, we used one IBM System Storage DS3524 Express array with three IBM System Storage EXP3524 expansion trays, each containing 24 drives, for a total of 96 usable disk drives. We cabled each IBM tray according to IBM best practices using SAS cables to interconnect the environmental service modules. We then cabled the two controllers from IBM System Storage DS3524 directly to the two Emulex LightPath LPe12000 single port 8Gb/s Fibre Channel cards installed in the IBM System x3850 X5 via two fibre cables. In both VMware and RHEV environments, to ensure an even distribution of traffic across both controllers and fibre cards, we assigned half of our configured LUNs to controller A and the other half to controller B using the IBM DS Storage Manager.

The entire IBM System Storage DS3524 Express array contained 96 drives. Due to the conceptual storage design differences between VMware and RHEV, we used a different storage configuration for each, namely RHEV's requirement that VMs are owned by a single storage domain. When switching between each hypervisor, we reconfigured the IBM System Storage DS3524 Express array using the parameters listed below using the IBM DS Storage Manager.

We created the following for VMware:

Note: VMware allows a virtual machine's virtual disks to reside on separate datastores, allowing the storage of OS, log, and data I/O across different sets of physical disks.

- A 24-disk array in a RAID 5 configuration. We then used this array to create two Logical Drives (LUNs) for our VMs' virtual disks containing the operating system files. We assigned each LUN to one of two controllers.
- An 18-disk array in a RAID 10 configuration. We then used this array to create two LUNs for our VMs' virtual disks containing SQL Server and PostgreSQL transaction log files. We assigned each LUN to one of two controllers.
- A 54-disk array in a RAID 10 configuration. We then used this array to create two LUNs for our VMs' virtual disks containing SQL Server and PostgreSQL database files. We also stored database backups on these LUNs. We assigned each LUN to one of two controllers.

We created the following for RHEV:

Note: RHEV uses the concept of storage domains, a logical unit that can contain multiple LUNs. However, VMs' virtual disks must all reside in a single storage domain. As such, we could not configure storage with detailed

granularity. Therefore, on each disk array we created, we used physical disks spread across all four IBM System Storage trays.

- A 96-disk array in a RAID 10 configuration. We then created four logical drives (LUNs), each with exactly one-quarter of the overall capacity. We assigned two logical drives to controller A and the other two to controller B.

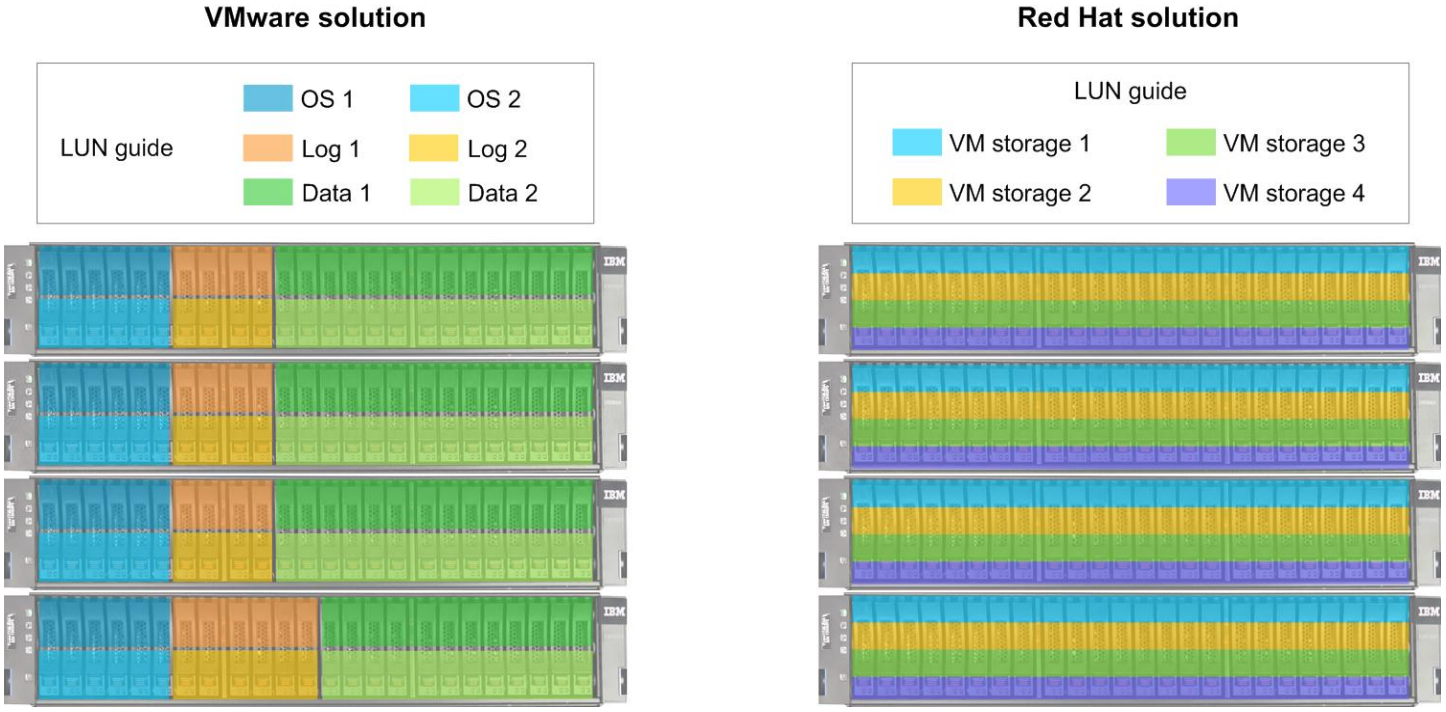


Figure 5. The storage layout we used for both the VMware vSphere 5 and Red Hat Enterprise Virtualization 3 testing.

Setting up the external storage for VMware and RHEV

1. Log in to the IBM System Storage DS3524 storage array via the IBM DS Storage Manager on a client system.
2. Select the Logical tab.
3. Right-click the Total Unconfigured Capactiy, and click Create Array…
4. Click Next.
5. Enter a name for the drive array based on the current environment.
6. Select Manual, and click Next.
7. Select the proper amount of disks and RAID level for the current environment, and click Add.
8. Click Calculate Capacity, and click Finish.
9. At the Array Created prompt, click Yes to begin the creation of a logical drive.
10. Click Next.
11. Enter the proper capacity amount for the current logical drive and environment.
12. Enter a proper name for the drive, and click Next.
13. Select Map later using the Mappings View, and click Finish.
14. Repeat steps 3 through 13 until the storage is completely setup for the current environment.
15. Prior to testing, allow each logical drive to complete the initialization task.

16. Ensure that the logical drives are divided between the two controllers by select each logical drive, and checking that the preferred and current owners are divided evenly between controller A and controller B.
17. Click the Mappings tab.
18. Right-click Storage Subsystem, and click Define→Host.
19. Enter a name for the host, and click Next.
20. Select FC for the host interface.
21. Select Add by selecting a known unassociated host port identifier.
22. Click Refresh to automatically identify attached Fibre Channel adapters.
23. Select the first host port identifier, enter an alias for the port, and click Add.
24. Repeat the last step for the second host port identifier.
25. Click Next.
26. Select the proper Host type (operating system) for the current hypervisor setup, and click Next.
27. Select No, and click Next.
28. Click Finish.
29. Once the host is created, click No.
30. In the left-hand pane, right-click the new host, and click Define→Additional Mapping.
31. Select each logical drive listed, and click Add.
32. Each LUN will now be accessible from the host.

APPENDIX C – SETTING UP THE VIRTUAL ENVIRONMENT – RED HAT ENTERPRISE VIRTUALIZATION 3

Adjusting BIOS settings

We used the latest released BIOS update on the IBM System x3850 X5, version IBM 1.73, and adjusted the default BIOS settings. We disabled CPU c-states and set the Operating Mode to Performance. For running RHEV-H, we added the Legacy Only option to the BIOS boot options and promoted it to the first boot option. RHEV-H would not boot properly if this option was not configured.

Setting up the Red Hat Enterprise Virtualization 3 environment

We used the following steps to install and configure the RHEV 3 environment. To manage our IBM System x3850 X5 running RHEV-H 6.2, we setup a separate management server running RHEV-M 3.0. We also configured a basic Windows Active Directory domain controller running DNS to handle domain and name resolution. We created a user on the domain to be used as the login for the RHEV-M web console. We also added the RHEV-M IP address to both the forward and reverse lookup zones in DNS.

Installing and configuring Red Hat Enterprise Virtualization Manager 3.0

We performed the following steps to install RHEV Manager, starting from a basic RHEL 6.2 installation. We assigned an IP address and hostname for the RHEV-M server during the installation of RHEL 6.2. The following steps require an active Internet connection and a subscription to Red Hat Network's service. We used a trial subscription for this testing.

1. Log in to the RHEV Manager server as root.
2. Disable SELinux by modifying the `/etc/selinux/config` file, and changing the SELINUX line to `SELINUX=disabled`
3. Synchronize the time source.
   a. Edit /etc/ntp.conf, adding the IP address of the domain controller as a time source.
   b. Run the command `chkconfig ntpdate on`.
   c. Run the command `chkconfig ntpd on`.
   d. Run the command `date`, and ensure that the time is synchronized with the domain controller.
4. Add the IP address and FQDN of the domain controller to /etc/hosts.
5. Type `rhn_register` to begin the registration on the RHN, and accept defaults, making sure to select the appropriate region.
6. Follow the prompts to register the system with RHN, ensuring that the yum repository for RHEL 6.2 is added during the process.
7. Add channels for RHEV by typing the following commands:

```
# rhn-channel --add --channel=rhel-x86_64-server-6-rhevm-3
# rhn-channel --add --channel=jbappplatform-5-x86_64-server-6-rpm
# rhn-channel --add --channel=rhel-x86_64-server-supplementary-6
```

8. Turn off RHEL firewalls by typing the following commands:

```
# service iptables save
```

```
# service iptables stop
# chkconfig iptables off
# service ip6tables save
# service ip6tables stop
# chkconfig ip6tables off
```

9. Update RHEL by typing `yum upgrade`
10. Start the RHEV-M install process by typing `rhevm-setup`
11. Accept the default HTTP port by pressing Enter.
12. Accept the default HTTPS port by pressing Enter.
13. Enter a password for the admin@internal username.
14. Enter a password for the internal database.
15. Choose your default storage type.
16. Allow the installer to configure an NFS share on the local server for an ISO Domain.
17. Choose a mount path for the ISO Domain share.
18. Choose a display name for the ISO Domain.
19. Allow the installer to automatically configure the firewalls.
20. After the install is complete run the following command:

```
# rhevm-manage-domains -action=add -domain=<Domian name> -user=<User
  created for RHEV-M> -interactive
```

21. Enter the password for the domain user, and press Enter to allow the command to complete the addition of the domain to RHEV-M.
22. Upload all necessary ISO images and drivers for the VM installs, including the RHEV tools ISO for Windows and the latest VirtIO drivers.

Installing RHEV-H 6.2

We performed the following steps to install RHEV Hypervisor on our system under test, and after installation configured approximately 120GB for swap. For further information, refer to the RHEV install documentation located on the RHEV-Hypervisor disk.

1. Boot to the Red Hat Enterprise Virtualization Hypervisor standalone installation disk.
2. Once the installer loads, select Install Hypervisor.
3. Select the proper internal logical disk to install RHEV on, and click Continue.
4. Select the disk again, and click Continue.
5. Enter and confirm a new password, and click Install.
6. Once the installation is completed, click Reboot.

Adding the RHEV-H 6.2 host to RHEV-M

1. Once the server has rebooted from the RHEV-H install, log in using the username “admin”. This will log in to the RHEV-H graphical setup interface.
2. Tab to Network, and press Enter.
3. Type in a proper host name and DNS server IP address for the RHEV-H server.
4. Tab down to Apply, and press Enter.
5. On the same screen, select the proper Ethernet device for management traffic, and press Enter.
6. Select Static, and assign the proper IP address, subnet mask, and gateway.

7. Tab to Apply, and press Enter.
8. Allow the network service to reboot, and press Enter when it completes.
9. Tab to Security on the left-hand side of the screen, and press Enter.
10. Select Enable ssh password authentication, and enter the desired password.
11. Tab to Apply, and press Enter.
12. Tab to RHEVM on the left-hand side of the screen, and press Enter.
13. Enter the IP address for the RHEV-M server.
14. Select Connect to RHEV Manager and validate certificate.
15. Enter the password for the RHEV-M admin, and click Apply.
16. Once the RHEV-H server finishes connecting to the RHEV-M server, press Enter.

Connecting to the RHEV-M server from the client

1. Connect to RHEV-M by typing the fully qualified domain name of the RHEV-M server in a Web browser from a client machine.
2. Click Administrator Portal (no SSL).
3. At the login screen, enter the name of the domain user created for RHEV-M, the user password, and select the proper domain.
4. Press Enter to log into the administrator console.
5. Change to the Clusters tab, select the default cluster, then move to the Logical Networks sub-tab, and select Add Network.
6. Add five additional virtual networks, one for Internet access, and four for the test network.
7. Navigate to the Default host cluster, and click Edit.
8. Click Memory Optimization, and select Optimized for Server Load to enable memory page sharing at a threshold of 150%.
9. Click OK.
10. Navigate to the Hosts tab.
11. The RHEV-H sever should be listed. Click the Install button to finalize adding the RHEV-H server to RHEV-M.
12. Once the RHEV-H server's status changes to Up, select the host and click Edit.
13. Select the host, and move to the Network Interfaces sub-tab.
14. In the Network Interfaces sub-tab, assign the logical networks to the appropriate ports by selecting a network interface, selecting Add/Edit, and assigning the appropriate logical network you created earlier. Ensure that the four test networks are assigned to the Intel quad-port NIC.
15. Change to the Storage tab and select New Domain. Add the ISO domain that was created automatically during the RHEV-M install.
16. Still in the Storage tab, add the remaining four data domains, selecting Data→Fibre Channel as the storage type. Use a naming scheme that will allow for easy identification when deploying the VMs via templates.
17. Ensure that each domain is attached to the Default cluster.

Creating the first Windows or RHEL VM

1. Log into the RHEV Manager.
2. In the Hosts tab, select New Server.
3. In the New Server Virtual Machine window, name your guest, select the Default cluster, make it based on the Blank template, select 8 GB of RAM, and select the appropriate operating system.
4. Click Resource Allocation, and enter 1 MB for Physical Memory Guaranteed.
5. Click OK.
6. A New Virtual Machine – Guide Me window appears. Click Configure Network Interfaces.
7. In the New Network Interface window, name the NIC, select the proper guest network, select the Red Hat VirtIO type, and click OK.

8. Back at the New Virtual Machine – Guide Me window, select Configure Virtual Disks.
9. In the New Virtual Disk window, make your disk 20 GB and select Thin Provision, and click OK. This virtual disk will be used to install the guest operating system.
10. Back at the New Virtual Machine – Guide Me window, click Configure Later.
11. Select the newly created VM, and click the Virtual Disks tab.
12. Click New.
13. Enter 22 GB for the size, leave the default settings, and click OK. This virtual disk will be used for SQL Server and PostgreSQL data files.
14. Click New.
15. Enter 13 GB for the size, leave the default settings, and click OK. This virtual disk will be used for SQL Server and PostgreSQL log files.
16. Click New.
17. Enter 13 GB for the size for RHEL guests or 3GB for the size for Windows guests, leave the default settings, and click OK. This virtual disk will be used for backup data.
    See Appendix E for VM related setup.

APPENDIX D – SETTING UP THE VIRTUAL ENVIRONMENT – VMWARE VSPHERE 5

Adjusting BIOS settings

We used the latest released BIOS update on the IBM System x3850 X5, version IBM 1.73, and adjusted the default BIOS settings. We disabled CPU c-states and set the Operating Mode to Performance.

Setting up the VMware vSphere 5 environment

We used the following steps to install and configure the VMware vSphere 5 environment. To manage our IBM System x3850 X5 running VMware vSphere 5 (ESXi), we setup a separate management server running Windows Server 2008 R2 with the latest build of VMware vCenter™ Server and the vSphere client software.

Installing VMware vSphere 5 (ESXi) on the IBM System x3850 X5

1. Insert the disk, and select Boot from disk.
2. On the Welcome screen, press Enter.
3. On the End User License Agreement (EULA) screen, press F11.
4. On the Select a Disk to Install or Upgrade Screen, select the relevant volume to install ESXi on and press Enter.
5. On the Please Select a Keyboard Layout screen, press Enter.
6. On the Enter a Root Password Screen, assign a root password and confirm it by entering it again, and press Enter to continue.
7. On the Confirm Install Screen, press F11 to install.
8. On the Installation complete screen, press Enter to reboot.

Configuring ESXi after Installation

1. On the 5.0.0 ESXi screen, press F2, enter the root password, and press Enter.
2. On the System Customization screen, select troubleshooting options, and press Enter.
3. On the Troubleshooting Mode Options screen, select enable ESXi Shell, and press Enter.
4. Select Enable SSH, press Enter, and press ESC.
5. On the System Customization screen, select Configure Management Network.
6. On the Configure Management Network screen, select IP Configuration.
7. On the IP Configuration screen, select set static IP, enter an IP address, subnet mask, and default gateway, and press Enter.
8. On the Configure Management Network screen, press Esc. When asked if you want to apply the changes, press Y.

Configuring VM networking on ESXi

1. Using the vSphere client from another machine, connect to the vCenter Server which manages the ESXi server.
2. Add the necessary vSwitches for the network that DVD Store traffic will use:
   a. Click the host, click the Configuration tab, and click Networking.
   b. Click Add Networking.
   c. Choose Virtual Machine, and click Next.
   d. Choose create a vSphere standard switch.
   e. Choose the NIC associated with VM traffic.
   f. Assign the network label and assign IP settings.
   g. Click Finish.
   h. Repeat steps a through g for the remaining three NICs from the Intel quad-port adapter.

Configuring the external volumes in VMware vSphere 5

1. In the vSphere client, select the host.
1. Click the Configuration tab.
2. Click Storage, and click Add Storage…
3. Choose Disk/Lun.
4. Select the disk, and click Next.
5. Accept the default of VMFS-5 for the file system.
6. Review the disk layout, and click Next.
7. Enter the datastore name, and click Next.
8. Accept the default of using maximum capacity, and click Next.
9. Click Finish.
10. Repeat steps 3 through 10 for the remaining LUNs.

Creating the first Windows or RHEL VM

1. In the vSphere client, connect to the vCenter Server, and browse to the ESXi host.
2. Click the Virtual Machines tab.
3. Right-click, and choose New Virtual Machine.
4. Choose Custom, and click Next.
5. Assign a name to the virtual machine, and click Next.
6. Select the first assigned OS Datastore on the external storage, and click Next.
7. Choose Virtual Machine Version 8, and click Next.
8. Choose the appropriate operating system, and click Next.
9. Choose two virtual sockets, and click Next.
10. Choose 8GB RAM, and click Next.
11. Click 1 for the number of NICs, select vmxnet3, and click Next.
12. Leave the default virtual storage controller, and click Next.
13. Choose to create a new virtual disk, and click Next.
14. Make the OS virtual disk size 20 GB, choose thin-provisioned, specify the first OS datastore on the external storage, and click Next.
15. Keep the default virtual device node (0:0), and click Next.
16. Click Finish.
17. Right-click the VM, and choose Edit Settings.
18. On the Hardware tab, click Add…
19. Click Hard Disk, and click Next.
20. Click Create a new virtual disk, and click Next.
21. Specify 22GB for the virtual disk size, choose thick-provisioned lazy zeroed, and specify the first datastore for backup and SQL Server/PostgreSQL data usage.
22. Choose SCSI(1:0) for the device node, and click Next.
23. On the Hardware tab, click Add…
24. Click Hard Disk, and click Next.
25. Click Create a new virtual disk, and click Next.
26. Specify 13GB for the virtual disk size, choose thick-provisioned lazy zeroed, and specify the first datastore for SQL Server/PostgreSQL log usage.
27. Choose SCSI(1:1) for the device node, and click Next.
28. On the Hardware tab, click Add…
29. Click Hard Disk, and click Next.
30. Click Create a new virtual disk, and click Next.

31. Specify 13GB for the virtual disk size for RHEL guests or 3GB for the virtual disk size for Windows guests, choose thick-provisioned lazy zeroed, and specify the datastore for usage.
32. Choose SCSI(1:2) for the device node, and click Next.
33. Click SCSI Controller 1, and choose Change Type.
34. Choose VMware Paravirtual, and click OK.
35. Click Finish, and click OK.
36. Start the VM.
37. Attach the Windows Server 2008 R2 SP1 ISO or RHEL 6.2 ISO image to the VM and install the operating system on your VM.
    See Appendix E for VM-related setup.

APPENDIX E – CONFIGURING THE VMS ON EACH HYPERVISOR

See the above sections regarding the initial creation of the virtual machines on each hypervisor. We provide steps below for installing the operating system, Microsoft SQL Server or PostgreSQL, and configuration of the VMs.

Installing the operating system on the Windows VM

1. Insert the installation DVD for Windows Server 2008 R2 SP1 Enterprise into the DVD drive, and attach the physical DVD drive to the VM. Alternatively, use an ISO image and connect to the ISO image from the VM console. For RHEV, right-click the VM, and click Run Once to open up a menu to attach both the VirtIO driver virtual floppy and the Windows installation ISO.
2. Open the VM console on vCenter Server or RHEV-M.
3. At the Language Selection Screen, click Next.
4. Click Install Now.
5. Select Windows Server 2008 R2 Enterprise (Full Installation), and click Next.
6. Click the I accept the license terms check box, and click Next.
7. Click Custom.
8. On the installation hard drive selection, select the virtual hard drive. For RHEL, the VirtIO drivers need to be installed before the disk will appear in the list.
9. Click Next.
10. At the User's password must be changed before logging on warning screen, click OK.
11. Enter the desired password for the administrator in both fields, and click the arrow to continue.
12. At the Your password has been changed screen, click OK.
13. Install the latest VMware tools package on the VM if using vSphere; install the latest RHEV tools on the VM if using RHEV. Restart as necessary.
14. Connect the machine to the Internet, and install all available Windows updates. Restart as necessary.
15. Enable remote desktop access.
16. Change the hostname and reboot when prompted.
17. Create a shared folder to store test script files. Set permissions as needed.
18. Adjust Internet Time synchronization.
    a. Click the clock in the lower right-hand corner.
    b. Select Change date and time settings.
    c. Click the Internet Time tab.
    d. Click Change settings…
    e. Enter the IP address for the domain controller in the Server field.
    f. Click Update now, and click OK.
    g. Click OK to close the properties window.
19. Set up networking:
    a. Click Start→Control Panel, right-click Network Connections, and choose Open.
    b. Right-click the VM traffic NIC, and choose Properties.
    c. Select TCP/IP (v4), and choose Properties.
    d. Set the IP address and subnet for the virtual NIC, which will handle outgoing server traffic. Click OK, and click Close.
20. In the VM, configure the VM storage:
    a. Click the Server Manager icon in the taskbar.
    b. In the left pane, expand Storage and click Disk Management.
    c. Right-click the first volume, and choose Initialize Disk.
    d. In the right pane, right-click the volume and choose New Simple Volume…
    e. At the welcome window, click Next.

f. At the Specify Volume Size window, leave the default selection, and click Next.
g. At the Assign Drive Letter or Path window, choose a drive letter, and click Next.
h. At the Format Partition window, choose NTFS and 64K allocation unit size, and click Next.
i. At the Completing the New Simple Volume Wizard window, click Finish.
j. Repeat steps c through i for the remaining VM volumes.

21. Copy the pre-created DVD Store backup file to the backup virtual disk inside the first VM.

Installing SQL Server 2008 R2 SP1 on the Windows VM

1. Open the console for the VM.
2. Log into the virtual machine.
3. Insert the installation DVD for SQL Server 2008 R2 into the host server's DVD drive.
4. Attach the physical DVD drive to the VM.
5. Click Run SETUP.EXE. If Autoplay does not begin the installation, navigate to the SQL Server 2008 R2 DVD, and double-click.
6. If the installer prompts you with a .NET installation prompt, click Yes to enable the .NET Framework Core role.
7. In the left pane, click Installation.
8. Click New installation or add features to an existing installation.
9. At the Setup Support Rules screen, wait for the check to complete. If there are no failures or relevant warnings, click OK.
10. Select the Enter the product key radio button, and enter the product key. Click Next.
11. Click the checkbox to accept the license terms, and click Next.
12. Click Install to install the setup support files.
13. If there are no failures displayed, click Next. You may see a Computer domain controller warning and a Windows Firewall warning. For now, ignore these.
14. At the Setup Role screen, choose SQL Server Feature Installation.
15. At the Feature Selection screen, select Database Engine Services, Full-Text Search, Client Tools Connectivity, Client Tools Backwards Compatibility, Management Tools –Basic, and Management Tools – Complete. Click Next.
16. At the Installation Rules screen, click Next once the check completes.
17. At the Instance configuration screen, leave the default selection of default instance, and click Next.
18. At the Disk space requirements screen, click Next.
19. At the Server configuration screen, choose NT AUTHORITY\SYSTEM for SQL Server Agent, and choose NT AUTHORITY\SYSTEM for SQL Server Database Engine. Click Next.
20. At the Database Engine Configuration screen, select Mixed Mode.
21. Enter and confirm a password for the system administrator account.
22. Click Add Current user. This may take several seconds.
23. Click Next.
24. At the Error and usage reporting screen, click Next.
25. At the Installation Configuration rules screen, check that there are no failures or relevant warnings, and click Next.
26. At the Ready to Install screen, click Install.
27. After installation completes, click Next.
28. Click Close.
29. Create a SQL Server login for the ds2user (see the Configuring the database server (DVD Store) section for the specific script to use).
30. Copy the pre-created DVD Store backup to the specified backup VHD volume.
31. Click Start→All Programs→Microsoft SQL Server 2008 R2→Configuration Tools, and click SQL Server Configuration Manager.
32. Expand SQL Server Network Configuration, and click Protocols for MSSQLSERVER.

33. Right-click TCP/IP, and select Enable.
34. Download and install Microsoft SQL Server 2008 R2 SP1.

Installing the operating system on the RHEL VM

1. Mount the RHEL 6.2 DVD or ISO to the VM.
2. Boot the VM to the Red Hat Enterprise Linux 6.2 installation DVD.
3. Press Enter to install using graphical mode.
4. At the media test screen, select Skip, and click Next.
5. At the Red Hat Enterprise Linux 6.2 title screen, click Next.
6. At the Choose a Language screen, select English, and click Next.
7. At the Keyboard Type screen, select U.S. English, and click Next.
8. At the Installation Devices screen, select Basic Storage Devices, and click Next.
9. Enter the VM's hostname, and click Next.
10. At the Time zone selection screen, select the appropriate time zone, and click Next.
11. Enter the root password in the Root Password and Confirm fields, and click Next.
12. At the type of installation screen, select Use All Space, check Review and modify partitioning layout, and click Next.
13. At the partitioning layout screen, select lv_home, and click Delete. At the Confirm Delete warning, click Delete.
14. Select lv_root, and click Edit. In the Size field, enter the Max size displayed. Click OK.
15. Click Next. If a warning appears, click Write changes to disk.
16. At the Boot loader screen, click Next.
17. At the default installation screen, select Basic Server, Customize now, and click Next.
18. Installation will begin automatically.
19. At the Congratulations screen, click Reboot.
20. Log into the server as root to complete the OS configuration.
21. Type `rhn_register` to begin the registration on the RHN, and accept defaults, making sure to select your appropriate region.
22. Follow the prompts to register the system with RHN, ensuring that the yum repository for RHEL 6.2 is added during the process.
23. Run the command `yum update` and follow the prompts to install all available system updates.
24. Edit the file /etc/selinux/config and change the line `SELINUX=enforcing` to `SELINUX=disabled`
25. Install additional software:
```
# yum install tuned
```
26. Disable these unused daemons with the following script:
```
# for s in auditd autofs avahi-daemon bluetooth cpuspeed crond cups dnsmasq \
   fcoe firstboot ip6tables iptables irqbalance kdump libvirt-guests lldpad \
   mdmonitor netconsole netfs nfs nfslock ntpdate portserve postfix qpidd \
   restorecond rhnsd rhsmcertd rpcbind rpcgssd rpcidmapd rpcsvcgssd; do
      chkconfig $s off
      service $s stop
   done
```
27. Ensure these services are enabled using the following script:
```
# for s in cpuspeed sshd sysstat multipathd tuned; do
      chkconfig $s on
      service $s start
   done
```
28. Synchronize the time source.
    a. Edit /etc/ntp.conf, adding the IP address of the domain controller as a time source.
    b. Run the command `chkconfig ntpdate on`

c. Run the command `chkconfig ntpd on`
d. Run the command `date`, and ensure that the time is synchronized with the domain controller.

29. Configure the VM's IP address on interface eth0.
30. Type the following command to restart networking to effect these changes:
```
# service network restart
```
31. Install guest tools or agents.
    a. (VMware only) Install VMware Tools on the guest:
        - Open a console window. Select VM on the window title-bar→Guest→Install/Upgrade VMware Tools.
        - Log onto the guest as `root`
        - Mount the CDROM device:
          `# mount –o ro /dev/cdrom /mnt`
        - Untar VMware Tools into a temporary directory:
          `# tar –C /tmp –zxf /mnt/VMwareTools-8.6.0-425873.tar.gz`
        - Run the install script and accept the defaults:
          `# /tmp/vmware-tools-distrib/vmware-install.pl`
        - The installer will automatically load the NIC drivers, create a new initrd, and unmount the CD.
    b. (RHEV only) Install the RHEV Agent via the latest RHEV Agent RPM, which in our case was rhev-agent-2.3.16-3.el6.x86_64.rpm).
32. Create one pair of file systems for the DS2 databases, and another for backup purposes:

```
# fdisk /dev/<device ID> ; fdisk /dev/<device ID> ; fdisk /dev/<device ID>
# mkfs.ext4 -L log01  /dev/<device ID>1
# mkfs.ext4 -L data01 /dev/<device ID>1
# mkfs.ext4 -L backup /dev/<device ID>1
# mkdir –p /vol/pglog1
# mkdir -p /vol/ds2data1
# mkdir –p /backup
```

33. Edit /etc/ fstab to mount these file systems on boot:

```
/dev/<device ID>1 /vol/pglog1   ext4 defaults 0 0
/dev/<device ID>1 /vol/ds2data1 ext4 defaults 0 0
/dev/<device ID>1 /backup ext4 defaults 0 0
```

34. Mount these file systems, and reset the tuned profile:

```
# mount -a
# tuned-adm profile default
```

35. Reboot the system:
```
# shutdown –r now
```

Installing PostgreSQL on the RHEL VM

We used the PostgreSQL database server, version 9.1, as the database software on the RHEL VMs.

1. Download the rpm for PostgreSQL 9.1 from http://yum.postgresql.org/repopackages.php and upload the rpm file to the VM.
2. Log onto the system as root.
3. Install the PostgreSQL database server and client:
```
# rpm –ivh <rpm name>
# yum install <name of installed postgresql package>
```

4. Edit these two lines /etc/init.d/postgresql-9.1 as follows:
   ```
   PGDATA=/vol/ds2data1
   PGLOG=/vol/ds2data1/pgstartup.log
   ```
5. Run the command `initdb` to initialize the PostgreSQL database cluster for the first time.
6. Ensure database files now appear in /vol/ds2data1.
7. To move the database log files to the /vol/pglog1 mounted to the separate virtual hard disk use the following steps.
   a. Run `service postgresql-9.1 stop`
   b. Remove the directory pg_xlog from /vol/ds2data1.
   c. Run `ln –s /vol/pglog1 /vol/ds2data1/pg_xlog`
   d. Run `service postgresql-9.1 start`
   e. Ensure that the PostgreSQL log files appear in /vol/pglog1.
8. Tune the database engine. Modify the following lines of the postgresql.conf file:
   ```
   checkpoint_segments = 10
   wal_buffers = 16MB
   shared_buffers = 2GB
   effective_cache_size = 6GB
   checkpoint_timeout = 300s
   checkpoint_completion_target=0.9
   ```
9. Add one line to the PostgreSQL configuration file pg_hba.conf to permit SQL queries from the client network:
   ```
   host all all <test bed IP subnet>.0/24    trust
   ```
10. Modify the listen_addresses line in the PostgreSQL configuration file postgresql.conf:
    ```
    listen_addresses = '*'
    ```
11. Run `service postgresql-9.1 restart` to restart PostgreSQL with the new settings.

Configuring additional VMs on Red Hat Enterprise Virtualization Manager 3

1. In the RHEV-M console, right-click the Windows VM, and click Make Template.
2. Enter a name, description, select the Default cluster, and select the proper storage domain.
3. Uncheck Make Private, and click OK.
4. Repeat steps 1 through 3 for the RHEL VM.
5. Select VMs, and click New Server.
6. Select the proper template, with every third VMs being based on the RHEL template. For example, VM-001 and VM-002 are based on the Windows template, while VM-003 is based on the RHEL template.
7. Type a name for the new VM.
8. Click Resource Allocation, and ensure that the Physical Memory Guaranteed parameter is set to 1 MB.
9. Select the proper storage domain. For RHEV, we spread the VMs virtual disk storage using a round robin approach across the four storage domains.
10. Click OK.
11. Continue deploying each VM up to 42 total, modifying the setting outlined above. For every third VM, use the RHEL VM for the clone image.
12. Ensure in each VM that the necessary virtual disks are all online.
13. Assign the proper IP addresses and hostnames to each VM.
14. For Windows/SQL Server VMs, modify the SQL Server hostname of each VM using the instructions provided by Microsoft (http://msdn.microsoft.com/en-us/library/ms143799.aspx).

Configuring additional VMs on VMware vSphere 5

1. Log into the vCenter Server, which manages the host.
2. Right-click the first Windows VM, and choose Clone.
3. Name the new VM.

4. Choose the cluster, and select the host.
5. For the storage screen, choose advanced and direct the new virtual disks to the applicable datastores. Again, we used a round-robin approach to spread the OS, database, log VMDKs and configuration files across our separate datastores. For instance, VM1's configuration files and OS VMDK were placed on datastore OS1, the log VMDK on Log1, and the database and backup VMDKs on DB1. VM2 files and VMDKs were placed on OS2, Log2, and DB2.
6. Choose to customize using the customization wizard. Save the clone details as a new customization specification.
7. Continue cloning each VM up to 42 total, modifying the customization specification as necessary for IP addressing and so on. For every third VM, use the RHEL VM for the clone image.
8. Ensure in each VM that the necessary virtual disks are all online, the hostname is renamed, and the IP addressing was properly assigned by the customization wizard.
9. For Windows/SQL Server VMs, modify the SQL Server hostname of each VM using the instructions provided by Microsoft (http://msdn.microsoft.com/en-us/library/ms143799.aspx).

APPENDIX F - CONFIGURING THE DATABASE (DVD STORE)

Data generation overview

We generated the data using the Install.pl script included with DVD Store version 2.1 (DS2), providing the parameters for our 8GB database size and the database platforms on which we ran: Microsoft SQL Server and PostgreSQL. We ran the Install.pl script on a utility system running Linux.

For our Windows VMs, we transferred the data files and schema creation files to a Windows-based system running SQL Server 2008 R2 SP1. We built the 8GB database in SQL Server 2008 R2 SP1, and then performed a full backup. We transferred that backup file to each Windows VM and used that backup file to restore on all Windows VMs between test runs.

For our RHEL VMs, we used the available scripts from the DVD Store distribution to load the generated data into PostgreSQL. We then stopped the PostgreSQL server, created a backup copy of the data files, and used those backup files on each RHEL VM to restore the PostgreSQL environment between test runs. Specifically, we followed the steps below:

1. We generated the data and created the database and file structure using database creation scripts in the DS2 download.
2. We transferred the files from our Linux data generation system to a Windows system running SQL Server and a Linux system running PostgreSQL .
3. We created database tables, stored procedures, and objects using the provided DVD Store scripts.
4. We loaded the data we generated into the database. For data loading in SQL Server, we used the import wizard in SQL Server Management Studio. Where necessary, we retained options from the original scripts, such as Enable Identity Insert. For data loading in PostgreSQL, we used provided scripts from DVD Store.
5. We created indices, full-text catalogs, primary keys, and foreign keys using the database-creation scripts.
6. We updated statistics on each table according to database-creation scripts, which sample 18 percent of the table data.
7. On each database platform, we created the necessary users.
8. For SQL Server, we created the necessary full text index using SQL Server Management Studio. For PostgreSQL, we used the provided scripts.

   For RHEL guests specifically, we made these additional adjustments.

1. Create the DS2 user on the OS on the Red Hat system
   ```
   # useradd ds2user
   ```
2. Download the latest DVD Store distribution for PostgreSQL from http://linux.dell.com/dvdstore/ and upload the file to the VM.
3. Unpack the DS2 distribution into /ds2.
4. Change directory to /ds2/postgresqlds2.
5. Run the shell script postgresqlds2_create_all.sh
6. Copy the database directories to /vol/ds2data/.

Running the DVD Store tests

We created a series of batch files, SQL scripts, and shell scripts to automate the complete test cycle. DVD Store outputs an orders-per-minute metric, which is a running average calculated through the test. In this report, we report the last OPM reported by each client/target pair.

Each complete test cycle consisted of the general steps listed below. For each scenario, we ran three test cycles, and chose the median outcome.

1. Clean up prior outputs from the host system and all client driver systems.
2. Drop all databases from all target VMs.
3. Restore all databases on all target VMs.
4. Shut down all VMs.
5. Reboot the host system and all client systems.
6. Wait for a ping response from the server under test (the hypervisor system), all client systems, and all VMs.
7. Let the test server idle for one hour.
8. Start the DVD Store driver on all respective clients.
   We used the following DVD Store parameters for testing the virtual machines in this study:

```
<driver exe name> --target=<target_IP> --ramp_rate=10 --run_time=45 --
n_threads=10 --db_size=8GB --think_time=0.1 --detailed_view=Y
```

APPENDIX G – SERVER, AND STORAGE CONFIGURATION INFORMATION

Figure 6 provides detailed configuration information for the IBM System x3850 X5 server, and Figure 7 provides configuration information for the IBM System Storage DS3524 Express storage array.

System	IBM System x3850 X5 server
Power supplies	
Total number	2
Vendor and model number	Emerson™ Network Power 70071524-J000
Wattage of each (W)	875
Cooling fans	
Total number	4
Vendor and model number	Nidec® V60E12BS1C3-07 \ Panasonic® PBT-GF30-FR
Dimensions (h x w) of each	1-1/2” x 2-1/5” / 1-1/2” x 4-1/2”
Volts	12 / 12
Amps	1.04 / 1.65
General	
Number of processor packages	4
Number of cores per processor	10
Number of hardware threads per core	2
CPU	
Vendor	Intel
Name	Xeon®
Model number	E7-4870
Stepping	A2
Socket type	Socket LGA1567
Core frequency (GHz)	2.4
Bus frequency	6.4 GT/s
L1 cache	32 KB + 32 KB (per core)
L2 cache	256 KB (per core)
L3 cache	30 MB (shared)
Platform	
Vendor and model number	IBM System x3850 X5
Motherboard model number	IBM 88Y5351 Microprocessor Board (Type 7145)
BIOS name and version	IBM 1.73
BIOS settings	Operating Mode set to Performance Mode; CPU C-states: Disabled
Memory module(s)	
Total RAM in system (GB)	256
Vendor and model number	Samsung® M393B1K73DH0-YF8
Type	PC3L-8500R
Speed (MHz)	1,066
Speed running in the system (MHz)	1,066
Size (GB)	8
Number of RAM module(s)	32
Chip organization	Double-sided

System	IBM System x3850 X5 server
Rank	Quad
RHEV-H hypervisor	
Name	Red Hat Enterprise Virtualization Hypervisor 6.2
File system	ext4
Kernel	2.6.32-220.4.2.el6.x86_64
Language	English
VMware hypervisor	
Name	VMware vSphere 5 Enterprise Update 1
Build number	623860
File system	VMFS
Kernel	5.0.0
Language	English
Graphics	
Vendor and model number	ATI ES1000
Graphics memory (MB)	16
RAID controller	
Vendor and model number	IBM ServeRAID M5015 6.0 Gb/s SAS/SATA Controller
Firmware version	2.120.223-1467
Cache size (MB)	512
Hard Drives	
Vendor and model number	IBM 42D0677
Number of drives	4
Size (GB)	146 GB
RPM	15,000
Type	6.0 Gb/s SAS
Onboard Ethernet adapter	
Vendor and model number	Broadcom® 5709S
Type	Integrated
Discrete quad-port Ethernet adapters	
Vendor and model number	Intel Ethernet Server Adapter I340-T4
Quantity	1
Type	Discrete
Optical drive(s)	
Vendor and model number	IBM UltraSlim Enhanced SATA DVD ROM
Type	DVD ROM
USB ports	
Number	6
Type	2.0

Figure 6. Detailed configuration information for the server under test.

Storage array	IBM System Storage DS3524 Express
Arrays	1 x DS3524 tray with 3 x EXP3524 expansion trays
Number of active storage controllers	2
Number of active storage ports	2
Firmware revision	07.77.20.00
Disk vendor and model number	IBM ST973452SS
Disk quantity	96
Disk size (GB)	73
Disk buffer size (MB)	16
Disk RPM	15,000
Disk type	SAS 6.0 Gb/s

Figure 7. Detailed configuration information for the storage array.

ABOUT PRINCIPLED TECHNOLOGIES

Principled Technologies, Inc.
1007 Slater Road, Suite 300
Durham, NC, 27703
www.principledtechnologies.com

We provide industry-leading technology assessment and fact-based marketing services. We bring to every assignment extensive experience with and expertise in all aspects of technology testing and analysis, from researching new technologies, to developing new methodologies, to testing with existing and new tools.

When the assessment is complete, we know how to present the results to a broad range of target audiences. We provide our clients with the materials they need, from market-focused data to use in their own collateral to custom sales aids, such as test reports, performance assessments, and white papers. Every document reflects the results of our trusted independent analysis.

We provide customized services that focus on our clients' individual requirements. Whether the technology involves hardware, software, Web sites, or services, we offer the experience, expertise, and tools to help our clients assess how it will fare against its competition, its performance, its market readiness, and its quality and reliability.

Our founders, Mark L. Van Name and Bill Catchings, have worked together in technology assessment for over 20 years. As journalists, they published over a thousand articles on a wide array of technology subjects. They created and led the Ziff-Davis Benchmark Operation, which developed such industry-standard benchmarks as Ziff Davis Media's Winstone and WebBench. They founded and led eTesting Labs, and after the acquisition of that company by Lionbridge Technologies were the head and CTO of VeriTest.