Principled Technologies®

TEST REPORT
AUGUST 2009

Virtualized database hypervisor scaling report: Red Hat Kernel Virtual Machine running on the Dell PowerEdge R710 solution

Executive summary

Red Hat, Inc. (Red Hat) commissioned Principled Technologies (PT) to analyze the scalability of virtualized online transaction processing (OLTP) performance of the Red Hat® Kernel Virtual Machine® (KVM) hypervisor technology.

KEY FINDINGS

While running one VM, the KVM solution delivered 14,204 OPM; while running two VMs, the KVM solution delivered 28,219 OPM (1.99 times more than 1 VM); while running four VMs, the KVM solution delivered 49,826 OPM (3.51 times more than 1 VM); and while running eight VMs, the KVM solution delivered 77,197 OPM (5.43 times more than 1 VM). (See Figures1 and 2.)

We ran the KVM hypervisor solution on a Dell™ PowerEdge™ R710 server and two shelves of Dell EqualLogic™ PS5000XV storage. The Dell PowerEdge R710 server had two 2.66GHz Intel® Xeon® Processor X5550s and 72 GB of RAM. We connected the Dell PowerEdge R710 server to the Dell EqualLogic storage via a 10Gb iSCSI connection. We installed Red Hat Enterprise Linux 5.4 Beta (kernel-2.6.18-159.el5) as the host operating system, and installed Red Hat Enterprise Linux 5.4 Beta (kernel-2.6.18-155.el5) as the guest operating system.

For this scaling analysis, we selected the DVD Store Version 2 (DS2) test tool. DS2 is an open-source simulation of an online e-commerce DVD store, where customers log in, browse, and order products. Each virtual machine (VM, also known as "guest") ran a single MySQL workload. One copy of the DS2 database was on each VM, with a 4GB database per VM. By combining multiple virtual machines, all running a CPU-, memory-, and disk-intensive workload, the test showed the scaling ability of the Red Hat Enterprise Linux with KVM platform, especially as it relates to an active OLTP environment. The main DS2 metric is orders per minute (OPM). We report OPM results from a 300-second period of steady activity and heavy load during the DS2 test run. The Workload section further explains the MySQL database workload.

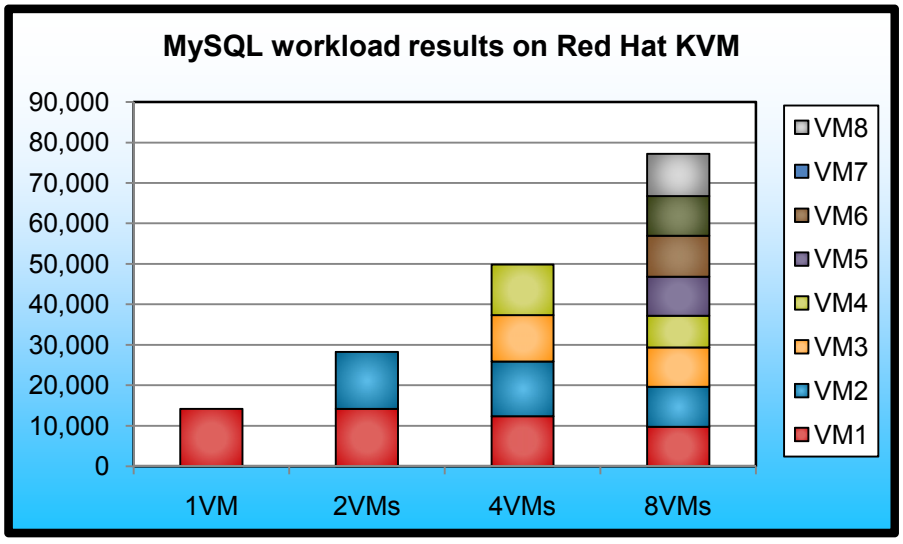

Figure 1: MySQL workload performance results for the KVM solution. Higher numbers are better.

Figure 1 compares the overall combined OPM of all VMs running on Red Hat KVM. We ran this solution with one, two, four, and eight VMs. The two-VM cumulative output was 1.99 times that of the one-VM output (28,219 OPM vs. 14,204 OPM, respectively), the four-VM cumulative output was 3.51 times that of the one-VM output (49,826 OPM vs. 14,204 OPM, respectively), and the eight-VM cumulative output was 5.43 times that of the one-VM output (77,197 vs. 14,204 OPM, respectively).

All test configurations featured two virtual CPUs (vCPUs) on each server VM and 4 GB of RAM on each server VM. Additionally, the boot partition was on the virtual IDE controller, while the remaining partitions used the virtio controller type.

We installed and ran the DS2 client from physical clients with Intel Xeon E5345 2.33Ghz processors and 8 GB of RAM. We ran a single instance of the DS2 client per physical client, and each client machine targeted only one server VM. We connected our client test bed to our physical server, and hence our virtual servers, via a Dell PowerConnect™ 6248 switch with a 10Gb uplink module.

Workload

We conducted our testing using DVD Store Version 2, an open-source application with a back-end database component, a front-end Web application layer, and a driver layer that actually executes the workload. We ran the driver application directly via its command-line interface.

DS2 models an online DVD store. Virtual customers log in; browse movies by actor, title, or category; and purchase movies. The workload also creates new customers. Browsing movies involves select operations, some of which use full-text search and some of which do not. The purchase, login, and new customer procedures involve updates and inserts, as well as selects.

For more details about the DS2 tool, see http://www.delltechcenter.com/page/DVD+Store.

Each client machine ran a single instance of DS2, which spawned 32 threads and used no think time. This simulated a heavily loaded environment; the load-generating client machines ran with no think time, generating requests as quickly as the servers could handle them. We used the default DS2 application parameters.

The DS2 driver application creates an orders-per-minute performance counter on the client machine. While the DVD Store client application outputs OPM at 10-second intervals visually, we chose to collect this OPM metric via a performance monitor counter on each client machine at 1-second intervals. We ran this workload on all virtual machines simultaneously for 30 minutes.

We show the number of OPM each solution achieved during our test. We calculate OPM as the number of overall orders divided by the elapsed seconds multiplied by 60. As such, we report the last OPM score the benchmark reported while all VMs were simultaneously running. We then sum those results at each VM count.

Test results

Figure 2 shows the cumulative OPM totals at each VM set during testing. The two-VM cumulative output was 1.99 times that of the one-VM output (28,219 OPM vs. 14,204 OPM, respectively), the four-VM cumulative output was 3.51 times that of the one-VM output (49,826 OPM vs. 14,204 OPM, respectively), and the eight-VM cumulative output was 5.43 times that of the one-VM output (77,197 vs. 14,204 OPM, respectively).

Server	Number of VMs			
	1	2	4	8
KVM solution	14,204	28,219	49,826	77,197
Normalized result	1.0	1.99	3.51	5.43

Figure 2: The cumulative OPM score for all VMs at each VM count. Higher numbers are better. We have normalized results to the one-VM output.

Test methodology

Setting up and configuring the storage

Internal storage configuration

We configured two volumes on the internal storage. We configured the first volume as a RAID 1 mirror of two 73GB drives, and we installed the host operating system on this volume. We configured the second volume on the remaining 146GB drive and used that volume for scripts and utility files.

Dell EqualLogic storage configuration

Each of the two Dell EqualLogic arrays had 16 drives, for a total of 32 drives. We configured the two Dell EqualLogic arrays in regular RAID 10 mode (14 active drives and 2 spares each), for a total of 28 active drives.

We connected the Dell PowerEdge R710 server to the Dell EqualLogic storage using a Dell PowerConnect 6248 switch. We connected the 10Gb network card on our server to the 10GbE uplink module on the switch via a Cat6 Ethernet cable. We created two storage pools and then alternated our VM storage between the two storage pools, with odd-numbered virtual machines on one array and even-numbered virtual machines on the other array.

We dedicated one 50GB logical unit number (LUN) to each VM. Then, in the Red Hat Enterprise Linux (RHEL) KVM host, we attached logical drives to those LUNs, where we then

stored our virtual disks. Each guest VM's data was stored on its respective LUN. This allowed us to maximize the throughput on the 10Gb NIC by opening several sessions on each iSCSI target. On the RHEL KVM host, we explicitly set paths to the storage and we used the iSCSI initiator at the host level on Red Hat to connect to the storage. Figure 3 shows the layout of the Dell EqualLogic storage and VM configuration.

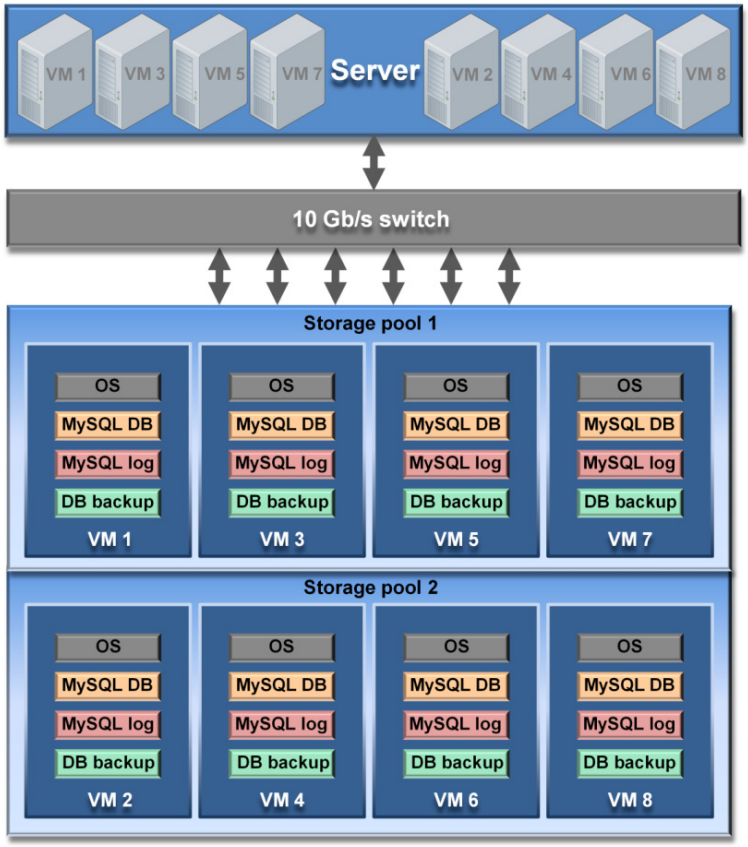

Figure 3: The layout of the Dell EqualLogic storage and VM configuration.

Setting up the storage

1. If the Dell EqualLogic PS5000 needs to be reset, connect through the serial port, and issue the following command upon login:

   ```
   # reset
   ```

2. Using the command line wizard, proceed with the reset process, and provide a group IP address, subnet mask, and login information.
3. Perform the same procedure on the second PS5000 array, but provide the group IP address from Step 2 as the group to join when the wizard prompts you.
4. Using the Group Manager web application, setup a storage group with the following RAID policies:
   a. PS5000XV-1: RAID-10
   b. PS5000XV-2: RAID-10
5. Adjust the members of the default storage pool to include only PS5000XV-1.
6. Create a storage pool name of `pool2` with the following member: PS5000XV-2.
7. Enable all network connections on each PS5000XV using the following IP address scheme:

a. IP Address: `192.168.1.##` (## being 11,13, and 15 for array 1, and 12, 14, 16 for array 2)
b. Subnet Mask: `255.255.255.0`

8. Create eight 50GB volumes with no snapshot reserve for KVM VMs, one for each of the eight VMs, alternating them between storage pools.
9. Enable shared access to the iSCSI target from multiple initiators on every volume.
10. Create an access control record for every volume without specifying any limitations.
11. Create a read-only SNMP community name to use for group monitoring.

Adjusting BIOS settings

We enabled the virtualization feature in the BIOS for our virtualization testing.

Setting up the host server with Red Hat 5.4 Beta with KVM

For the Red Hat KVM configuration, we installed kernel-2.6.18-159.el5 of Red Hat Enterprise Linux 5.4 Beta on the host server, and then installed the packages necessary for KVM.

Installing Red Hat 5.4 Beta

1. Insert and boot from the Red Hat Enterprise Linux 5.4 install DVD.
2. Press Enter to install using graphical mode.
3. At the media test screen, select Skip.
4. At the Red Hat Enterprise Linux 5 title screen, click Next.
5. At the Choose a Language screen, select English, and click Next.
6. At the Keyboard Type screen, select U.S. English, and click Next.
7. At the Installation Number pop-up screen, enter no Installation number, and click OK.
8. At the Installation requires partitioning screen, click Next.
9. If a warning appears, click Yes.
10. At the hard drive layout screen, click Next.
11. At the Boot loader screen, click Next.
12. At the Network Devices screen, click Edit, and deselect Enable IPv6. Click OK.
13. Under Hostname, select manually, and set the hostname to ##.domain.com, where ## is the name of your domain, and click Next. If you receive any warnings, click Continue.
14. At the Time zone selection screen, select the appropriate time zone, and click Next.
15. Enter the root password in the root password and confirm fields, and click Next.
16. At the default installation screen, leave the customized items unchecked, and click Next.
17. Click Next to begin the installation.
18. At the Congratulations screen, click Reboot.
19. After the system reboots and the Welcome screen appears, click Forward.
20. Agree to the EULA, and click Forward.
21. At the Firewall screen, select Disabled from the Firewall drop-down menu, and click Forward.
22. If a warning pop-up screen appears, click Yes.
23. At the SELinux screen, select Disabled from the SELinux Setting drop-down menu, and click Forward.
24. If a warning appears, click Yes.
25. At the Kdump screen, uncheck Enable Kdump, and click Forward.
26. At the Date and Time screen, set the time and date, and click Forward.
27. At the Set Up Software Updates screen, select No, I prefer to register at a later time, and click Forward.
28. If a pop-up screen appears asking if you are sure, click No thanks.
29. At the Finish Updates Setup, click Forward.
30. At the Create User Screen, click Forward.
31. If a warning appears, click Continue.
32. At the Sound Card screen, click Forward.
33. At the Additional CDs screen, click Finish.

34. At the pop-up screen, click OK.
35. At the logon screen, type root for the user, enter the root password, and press Enter.

Installing the KVM packages in RHEL 5.4 Beta

1. Copy the RHEL 5.4 ISO image, and mount it at /mnt/rhel54/.
2. Create a text file at /etc/yum.repos.d/rhel-54-dvd.repo with the following contents:

```
[rhel-54-beta-dvd]
name=Red Hat Enterprise Linux $releasever - $basearch - DVD
baseurl=file:///mnt/rhel54/Server/
enabled=1

[rhel-54-vt-beta-dvd]
name=Red Hat Enterprise Linux $releasever VT - $basearch - DVD
baseurl=file:///mnt/rhel54/VT/
enabled=1
```

3. Install the two GPG keys located on the Red Hat ISO by typing the following commands in a command prompt:

```
rpm --import /mnt/rhel54/RPM-GPG-KEY-redhat-beta
rpm --import /mnt/rhel54/RPM-GPG-KEY-redhat-release
```

4. Type the following command to install KVM packages:

```
# yum groupinstall kvm
```

Setting up the network adapters

1. Log into the host, right-click the desktop, and choose Open Terminal.
2. Type the following command to edit the network configuration settings, where X is the relevant host NIC to modify:

```
# gedit /etc/sysconfig/network-scripts/ifcfg-ethX &
```

3. Modify the following lines to set the static IP address and netmask, where X is the remaining parts of the relevant IP address:

```
BOOTPROTO=static
IPADDR=192.168.XXX.XXX
NETMASK=255.255.255.0
```

4. Save the file, and exit gedit.
5. Type the following command to modify the remaining network settings:

```
# gedit /etc/sysconfig/network &
```

6. Modify the following lines to disable IPv6 and set the hostname, where “hostname” is the hostname of this server:

```
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=hostname.local.domain
```

7. Save the file, and exit gedit.
8. Repeat steps 2 through 7 for the NIC dedicated to iSCSI traffic and the NIC dedicated to VM traffic.
9. Reboot the server.

Connecting the storage to the host on Red Hat KVM

On the Dell EqualLogic storage, we connected the eight LUNs dedicated to each virtual machine to the RHEL 5.4 Beta server via one physical 10Gb NIC. We created one volume group per LUN, then one logical volume per volume group. We then used these logical volumes to store virtual disks for the VMs. Each VM had four virtual drives: one for the Red Hat 5.4 Beta guest OS, one for MySQL database data, one for MySQL Innodb log files, and one utility partition for database backups.

We bridged a virtual NIC to an additional physical 10Gb NIC for DVD Store client to server communications.

Configuring the iSCSI software initiator and multipathing in Red Hat KVM

1. Log into the host, right-click the desktop, and choose Open Terminal.
2. Type the following command to install the iSCSI initiator utilities:

```
# yum install iscsi-initiator-utils
```

3. Using gedit or other text editor, edit /etc/iscsi/iscsid.conf, and modify the following settings:

```
# node.conn[0].timeo.noop_out_interval = 5
# node.conn[0].timeo.noop_out_timeout = 10
# node.session.timeo.replacement_timeout = 15
```

4. Type the following sequence of commands to configure the iSCSI interface, configure iSCSI to start on boot, and to start it immediately:

```
# iscsiadm -m iface -I iface0 --op=new
# . /etc/sysconfig/network-scripts/ifcfg-eth1
# iscsiadm -m iface -I iface0 --op=update -n iface.hwaddress -v $HWADDR
# chkconfig iscsid on ; service iscsid start
# chkconfig iscsi on ; service iscsi start
```

5. Type the following command to discover the LUNs on the Dell EqualLogic PS5000 arrays, where XX completes the IP address of the iSCSI group IP address:

```
# iscsiadm -m discovery -t st -p 192.168.1.XX
```

6. Note the target devices. If necessary, pipe the output to a file.
7. Log out of the iSCSI targets:

```
# iscsiadm -m node --logoutall=all
```

8. Log into each target in order (LUN1, LUN2, etc.), again where XX is the IP address of the iSCSI group:

```
# iscsiadm -m node -T iqn.2001-05.com.equallogic:0-8a0906-15d214102-
7a02a913e514a54f-01-vm-rhel -l -p 192.168.1.XX
```

9. Log out of the default portal:

```
# iscsiadm -m node --logoutall=all
```

10. Using the information from the above steps, and the IP addresses on the Dell EqualLogic storage, use iscsiadm utility to log into the storage once again, assigning 3 NICs to each LUN. The below example connects the host server to the LUNs assigned to VM1, using the 3 NICs assigned to that Dell EqualLogic array:

```
# iscsiadm -m node -T iqn.2001-05.com.equallogic:0-8a0906-15d214102-
7a02a913e514a54f-01-vm-rhel -o new -n node.startup -v automatic -I iface0 -p
192.168.1.10
# iscsiadm -m node -T iqn.2001-05.com.equallogic:0-8a0906-15d214102-
7a02a913e514a54f-01-vm-rhel -o new -n node.startup -v automatic -I iface0 -p
192.168.1.12
# iscsiadm -m node -T iqn.2001-05.com.equallogic:0-8a0906-15d214102-
7a02a913e514a54f-01-vm-rhel -o new -n node.startup -v automatic -I iface0 -p
192.168.1.14
```

11. To configure multipathing, edit /etc/multipath.conf, and change these sections to reflect the following:

```
blacklist {
        devnode "sd[a-c]$"
        devnode "^(ramrawloopfdmddm-srscdst)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss!c[0-9]d[0-9]*"0
}

        device {
                vendor "EQLOGIC"
                product "100E-00"
                path_grouping_policy group_by_prio
                getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
                features "1 queue_if_no_path"
                path_checker readsector0
                failback immediate
        }
```

12. Set multipathd to start automatically, and start it now:

```
# chkconfig multipathd on; service multipathd start
```

Adding the storage volumes to Red Hat

1. In a terminal window, use the pvcreate command to initialize the disk for use by LVM, where X is the device letter:

```
# pvcreate /dev/sdX
```

2. Repeat Step 1 for each iSCSI volume.
3. Issue the lvcreate command to create the logical volume, where “volname” is the logical volume name, and “volgroupname” is the volume group name:

```
# lvcreate -C y -n volname -l 100%VG volgroupname
```

4. Repeat Step 3 for each iSCSI volume.
5. Issue the mkfs.ext3 command to make the file system:

```
# mkfs.ext3 -L volname /dev/volgroupname/volname
```

6. Repeat Step 5 for each volume.
7. Edit the fstab to mount the volumes on boot. Add a line for each volume, such as below. This assumes the /mnt/volname folder exists.

```
/dev/volgroupname/volname /mnt/volname _netdev 0 0
```

Completing configurations in the Red Hat KVM host

Disabling unneeded services

1. Create a shell script to disable the following services:

```
service auditd stop
service avahi-daemon stop
service anacron stop
service qpidd stop
service smartd stop
service crond stop
service haldaemon stop
service opensmd stop
service openibd stop
service yum-updatesd stop
service collectd stop
service bluetooth stop
service cups stop
service isdn stop
service kudzu stop
service mcstrans stop
service mdmonitor stop
service messagebus stop
service restorecond stop
service rhnsd stop
service rpcgssd stop
service setroubleshoot stop
```

Installing the systat package for system monitoring

1. Insert the RHEL 5.4 DVD into the host machine.
2. At the command line, browse to the DVD drive.
3. Find the systat package under the Server directory by typing the following command (note the resulting file in the list):

```
# ls systat*
```

4. Install the package by typing the following command, where rpmpackage is the name we noted in Step 3 above:

```
# rpm –i rpmpackage
```

Enabling huge pages memory functionality in Red Hat KVM

1. Log into the host, right-click the desktop, and choose Open Terminal.
2. Type the following command to edit the system configuration settings:

```
# gedit /etc/sysctl.conf &
```

3. Add the following lines to the bottom of the file to reserved 48GB for VM huge pages, with each page at 2,048 KB:

```
# vm.nr_hugepages = 24576
```

4. Type the following command to edit the rc.local file:

```
# gedit /etc/rc.local &
```

5. Add the following lines to the bottom of the file:

```
mkdir -p /mnt/libhugetlbfs
mount -t hugetlbfs hugetlbfs /mnt/libhugetlbfs
chmod 777 /mnt/libhugetlbfs
```

Setting up the bridged network for Red Hat KVM

1. Log into the host, right-click the desktop, and choose Open Terminal.
2. Type the following command to create a configuration file for the bridged network device:

```
# gedit /etc/sysconfig/network-scripts/ifcfg-br0 &
```

3. Add the following lines to the file. Note that, in the below example, "X" signifies a placeholder:

```
# NIC details - copy from ifcfg-ethX file
DEVICE=br0
BOOTPROTO=none
IPADDR=192.168.XX.XXX
NETMASK=255.255.255.0
ONBOOT=yes
TYPE=Bridge
USERCTL=no
IPV6INIT=no
DELAY=0
```

4. Save the file, and exit.
5. Type the following command to edit the ifcfg-ethX file, where "X" is the number of the NIC you are using for your bridge:

```
# gedit /etc/sysconfig/network-scripts/ifcfg-ethX &
```

6. Modify the file to point to the bridge device:

```
# NIC details
DEVICE=eth0
```

```
HWADDR=00:1B:21:29:CE:74
BRIDGE=br0
```

7. Save the file, and exit.
8. Create the file /etc/qemu-ifup.
9. Modify the contents as follows:

```
#!/bin/sh
/sbin/ifconfig $1 0.0.0.0 up
/usr/sbin/brctl addif br0 $1
```

10. Save the file, and exit.
11. Restart the network by typing the following:

```
# /etc/init.d/network restart
```

Setting the elevator=deadline option in the grub.conf file

1. Log into the host, right-click the desktop, and choose Open Terminal.
2. Type the following command to edit the grub configuration settings:

```
# gedit /etc/grub.conf &
```

3. Add the following text to the end of the kernel line:

```
elevator = deadline
```

4. Save the file, and exit.

Creating and configuring the VMs

Creating the virtual machine using the Virtual Machine Manager

1. Log into the RHEL 5.4 host.
2. Click Applications, System Tools, Virtual Machine Manager.
3. Click New.
4. Click Forward.
5. Choose a name for the VM. Click Forward.
6. Keep the default of “Fully virtualized,” choose the CPU architecture, and choose kvm as the hypervisor.
7. Assuming you have copied the RHEL 5.4 beta media to the host machine, choose “Local install media,” Linux as the OS Type, and Red Hat Enterprise 5 as the OS Variant. Click Forward.
8. Click Browse to browse to the ISO Linux location.
9. Locate the ISO file, select it, and click Open. Click Forward.
10. Choose File (disk image), and specify the location where you wish to store the IMG file. In our case, we chose the relevant mounted iSCSI lun, which was at /mnt/0#-vm-rhel/OS.img.
11. Specify 9000MB for the size of the IMG file, and uncheck and recheck the “Allocate entire virtual disk now” checkbox. Click Forward.
12. Choose “Shared physical device,” and select the physical NIC configured for virtual networking. Choose virtio as the type, and click Forward.
13. Set Max memory size and Startup memory size to 4096MB, and set the number of Virtual CPUs to 2. Click Forward.
14. Review the summary information, and click Finish.

Installing the guest operating system

1. Double-click the new VM to connect to the console.
2. On the Hardware tab in Virtual Machine Manager, specify the ISO image on the host machine as the CD drive of the VM. Right-click the VM, and choose Run.
3. Power the VM on.
4. Click Enter to enter the graphical Red Hat installation.
5. At the media test screen, select Skip.
6. At the Red Hat Enterprise Linux 5 title screen, click Next.
7. At the Choose a Language screen, select English, and click Next.
8. At the Keyboard Type screen, select U.S. English, and click Next.
9. At the Installation Number pop-up screen, choose to skip Installation Numbers, and click OK. Click Yes to proceed if the installer prompts you again.
10. At the Installation requires partitioning screen, click Next.
11. If a warning appears, click Yes.
12. At the hard drive layout screen, click Next.
13. At the Time zone selection screen, select the appropriate time zone, and click Next.
14. Type `Password1` in the root password and confirm fields, and click Next.
15. At the default installation screen, choose Customize Now, and click Next.
16. Choose Servers, and select MySQL server.
17. Choose Development, select Development Libraries and Development tools, and click Next.
18. Click Next to begin the installation.
19. At the Congratulations screen, click Reboot.
20. After the system reboots and the Welcome screen appears, click Forward.
21. Agree to the EULA, and click Forward.
22. At the Firewall screen, select Disabled from the Firewall drop-down menu, and click Forward.
23. If a warning pop-up screen appears, click Yes.
24. At the SELinux screen, select Disabled from the SELinux Setting drop-down menu, and click Forward.
25. If a warning appears, click Yes.
26. At the Kdump screen, uncheck Enable Kdump, and click Forward.
27. At the Date and Time screen, set the time and date, and click Forward.
28. At the Set Up Software Updates screen, select No, I prefer to register at a later time, and click Forward.
29. If a pop-up screen appears asking if you are sure, click No thanks.
30. At the Finish Updates Setup, click Forward.
31. At the Create User Screen, click Forward.
32. If a warning appears, click Continue.
33. At the Sound Card screen, click Forward.
34. At the Additional CDs screen, click Finish.
35. At the pop-up screen, click OK.
36. At the logon screen, type `root` for the user and `Password1` for the password, and press Enter.
37. Right-click the RHEL desktop, and choose Open Terminal.
38. Type the following command to edit the /etc/grub.conf file:

```
# gedit /etc/grub.conf &
```

39. Add the following to the end of the kernel line:

```
divider=10
```

40. Save the file, and exit gedit.
41. Restart the VM.

Configuring networking in the VM

1. Power on the VM, and open the console.
2. Log into the VM, right-click the desktop, and choose Open Terminal.
3. Type the following command to edit the network configuration settings:

```
# gedit /etc/sysconfig/network-scripts/ifcfg-eth0 &
```

4. Modify the following lines to set the static IP address and netmask, where XXX is the remaining portion of your IP Address:

```
BOOTPROTO=static
IPADDR=192.168.XXX.XXX
NETMASK=255.255.255.0
```

5. Save the file, and exit gedit.
6. Type the following command to modify the remaining network settings:

```
# gedit /etc/sysconfig/network &
```

7. Modify the following lines to disable IPv6 and set the hostname:

```
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=VM1.local.domain
```

8. Save the file, and exit gedit.
9. Reboot the VM.

Mounting additional volumes in the Red Hat guest VM

1. Log into the VM.
2. Right-click the desktop, and choose Open Terminal.
3. Make a directory for MySQL data, MySQL Log, and utility mount points:

```
# mkdir /mysqldata
# mkdir /mysqllog
# mkdir /ds2backup
# chown mysql:mysql /mysqldata
# chown mysql:mysql /mysqllog
```

4. Type the following command to view devices attached to the SCSI channel:

```
# ls /dev/vd*
```

5. Find the additional virtual disks you need to partition and mount.
6. Type the following command to start the partition process:

```
# fdisk /dev/vdX
```

   Where `X` is the letter of the relevant disk. For example, /dev/vdb.

7. Type `n` for new partition.

8. Type `p` for primary partition.
9. Type `1` for partition number.
10. Press Enter to accept the default for the first cylinder.
11. Press Enter to accept the default for the last cylinder.
12. Press w to write the configuration.
13. Type the following command to make the file system on the partition on the disk device:

```
# mkfs.ext3 -L mysqldata /dev/vdb1
```

    We chose to mount the volumes manually as part of testing procedures instead of adding them to /etc/fstab.

14. Mount the drives immediately to build the DVD Store database.

Starting the MySQL service

1. Power on the VM, if not already running, and open the console.
2. Copy the new my.cnf file to /etc folder. We used a my.cnf sample file bundled with RHEL 5.4, at /usr/share/mysql/my-innodb-heavy-4G.cnf.
3. Modify this file to reflect the following changes:

```
datadir = /mysqldata
innodb_log_group_home_dir = /mysqllog
innodb_buffer_pool_size = 3G
innodb_data_file_path = ibdata1:10240M:autoextend
```

4. Save the file, and exit the text editor.
5. Right-click the desktop, and choose Open Terminal.
6. Type the following command to start MySQL:

```
# service mysqld start
```

7. Create a MySQL user with full permissions for the DVD Store workload using the following commands:

```
# mysql –u root -p -h localhost
mysql> CREATE USER 'web'@'localhost' IDENTIFIED BY 'web';
mysql> GRANT ALL PRIVILEGES ON *.* TO 'web'@'localhost' WITH GRANT OPTION;
mysql> CREATE USER 'web'@'%' IDENTIFIED BY 'web';
mysql> GRANT ALL PRIVILEGES ON *.* TO 'web'@'%' WITH GRANT OPTION;
```

Building the DVD Store database

1. Power on the VM, and open the console.
2. Log into the VM, right-click the desktop, and choose Open Terminal.
3. Mount all partitions.
4. Copy the DVD Store MySQL create scripts and relevant datasets to the VM.
5. Browse to the main DVD Store folder (/mysqlds2/ds2/mysqlds2).
6. Execute the following shell script to build the DVD Store database. For more information on building the data, see the following section, DVD Store Setup.

```
# ./mysqlds2_create_all.sh
```

Installing the systat package for system monitoring

Follow the procedures we describe above in the Installing the systat package for system monitoring section to install the systat monitoring package inside the VM.

Configuring additional tuning options in the VM

1. Log into the VM using Virtual Machine Manager, or using an ssh client, such as Putty.
2. Add the following lines to /etc/sysctl.conf:

```
vm.swappiness = 0
dirty_ratio = 90
dirty_background_ratio = 80
```

3. Save the file, and exit.
4. Create a shell script to disable the following services:

```
service auditd stop
service avahi-daemon stop
service anacron stop
service irqbalance stop
service qpidd stop
service smartd stop
service crond stop
service haldaemon stop
service opensmd stop
service openibd stop
service yum-updatesd stop
service collectd stop
service bluetooth stop
service cups stop
service hidd stop
service isdn stop
service kudzu stop
service mcstrans stop
service mdmonitor stop
service messagebus stop
service restorecond stop
service rhnsd stop
service rpcgssd stop
service setroubleshoot stop
```

Creating VM Startup Shell scripts to use huge pages

You must create shell scripts to start the VMs on the RHEL KVM platform in order to utilize the huge pages functionality.

1. Log into the host.
2. Start the Virtual Machine Manager.
3. Start the first VM by right-clicking and choosing Run.
4. Right-click the desktop, and choose Open Terminal.
5. Type the following command to show VM specific parameters:

```
# ps -ef | grep qemu
```

6. Using the parameters shown in the output from Step 5 above, modify the below script to start the RHEL KVM VM (Note that we have used the # symbol below to show placeholder characters where you should modify):

```
numactl --physcpubind=1,3 --localalloc usr/libexec/qemu-kvm --mem-path
/mnt/libhugetlbfs -m 4096 -smp 2 -name VM1 -uuid ########-####-####-####-
############ -usbdevice tablet -cpu qemu64,+sse2 -monitor pty -boot c -drive
file=/mnt/01-vm-rhel/OS.img,if=ide,index=0,boot=on -drive
file=,if=ide,media=cdrom,index=2 -drive file=/mnt/01-vm-
rhel/MySQLData.img,if=virtio,index=0,cache=writethrough -drive file=/mnt/01-
vm-rhel/MySQLLog.img,if=virtio,index=1,cache=writethrough -drive
file=/mnt/01-vm-rhel/Utility.img,if=virtio,index=2,cache=writethrough -net
nic,macaddr=##:##:##:##:##:##,vlan=0,model=virtio -net tap,script=/etc/qemu-
ifup,vlan=0,ifname=vnet0 -serial none -parallel none -usb -usbdevice tablet -
vnc 127.0.0.1:5900 -k en-us
```

7. Repeat steps 3 through 6 to create a shell script to start each VM on the RHEL KVM platform.
8. Using numactl in each start script, we configured our CPU affinity in each VM start script as follows:

When running 4 VMs or under:
VM1: `numactl --physcpubind=1,3`
VM2: `numactl --physcpubind=0,2`
VM3: `numactl --physcpubind=5,7`
VM4: `numactl --physcpubind=4,6`

When running 8 VMS:
VM1: `numactl --physcpubind=1,9`
VM2: `numactl --physcpubind=0,8`
VM3: `numactl --physcpubind=5,13`
VM4: `numactl --physcpubind=4,12`
VM5: `numactl --physcpubind=3,11`
VM6: `numactl --physcpubind=2,10`
VM7: `numactl --physcpubind=7,15`
VM8: `numactl --physcpubind=6,14`

Setting up DVD Store

Data generation overview

We built the database schema using the scripts in the DS2 distribution package, though we made a few minor modifications. The DS2 stress tool provides options to generate 10MB, 1GB, or 100GB datasets. To get the tool to generate the 4 GB of user data we used in this test, we had to make a few straightforward changes to the source code and to the DVD Store application's scripts. Note: We created our test data on a Linux system to take advantage of the larger RAND MAX.

Editing the ds2_create_orders.c module

The module ds2_create_orders.c defines constants that define the maximum values for the customer ID and the product ID. The constants for the 4GB database size did not exist. We added the constants for this size.

On the command line for the ds2_create_orders.c module, we specified the size. The available options were S (small), M (medium), and L (large). We added the case R for the 4GB database. In the switch statement that sets

the values for the variables max_cust_id and max_prod_id, we added cases that assigned them the proper values for the 4GB database size.

We recompiled the ds2_create_orders.c module on Linux, following the instructions in the header comments. We used the following command line: `gcc -o ds2_create_orders ds2_create_orders.c -lm`

Editing the ds2_create_cust.c module

We had to make the same changes to the ds2_create_cust.c module that we made to the ds2_create_orders.c module. On the command line for the ds2_create_cust.c module, we specified the size. The available options were S (small), M (medium), and L (large). We added the case R for the 4GB database. In the switch statement that sets the values for the variables max_cust_id and max_prod_id, we added cases that assigned them the proper values for the 4 GB database size.

We recompiled the ds2_create_cust.c module on Linux, following the instructions in the header comments. We used the following command line:

```
gcc -o ds2_create_cust ds2_create_cust.c -lm
```

Generating the data for the 4GB database

We used shell scripts to run all four of the executables that generate the data. The distribution did not include shell scripts for the 4GB size. We wrote shell scripts based on the ds2_create_cust_large.sh and ds2_create_orders_large.sh scripts. The ds2_create_prod and ds2_create_inv executables did not ship with associated shell scripts, so we created shell scripts using the instructions in the readme files. We ran the shell scripts in the following order to generate the data for the 4GB database:

1. `ds2_create_orders_4gb.sh`
2. `ds2_create_inv_4gb.sh`
3. `ds2_create_prod_4gb.sh`
4. `ds2_create_cust_4gb.sh`

We waited until the processes finished before we moved onto the next step.

Creating the database

After creating the MySQL user as we note above, we then placed the generated CSV data into the appropriate folders in the DVD Store distribution and ran the mysqlds2_create_all.sh shell script as distributed to create the database. In our case, we placed this data on our utility partition, which we had mounted at /ds2data on each guest VM.

After configuring and creating the MySQL DS2 database, we performed a cold backup. We stopped the mysqld service, copied all relevant files to our utility partition, and restarted the mysqld service. This backup allowed us to restore each server VM to a pristine state relatively quickly between tests.

Editing the workload script - ds2xdriver.cs module

To use the 4GB database we created earlier, we had to change the following constants in the DVD Store client application:

- In the routine Controller(), we changed the string sizes. We added the R option for the 4GB database size. DS2 uses the sizes string to interpret the db_size_str option.
- In the class Controller, we changed the arrays MAX_CUSTOMER and MAX_PRODUCT. To each, we added values specifying the bounds for the customer and product IDs. The Controller() routine uses these arrays.

Recompiling the ds2mysqldriver.exe executable

We recompiled the ds2xdriver.cs and ds2mysqlfns.cs module on Windows by following the instructions in the header comments. Because the DS2 instructions were for compiling from the command line, we used the following steps:

1. Install the MySQL connector on the development workstation.
2. Copy the MySql.Data.dll file to the same directory as the DVD Store source files.
3. Open a command prompt.
4. Use the cd command to change to the directory containing our sources.
5. Run the batch file C:\Program Files\Microsoft Visual Studio 9.0\Common7\Tools\vsvars32.bat. This sets up the environment variables for us.
6. Execute the following command:

```
csc /out:ds2mysqldriver.exe ds2xdriver.cs ds2mysqlfns.cs /r:MySql.Data.dll
/d:USE_WIN32_TIMER /d:GEN_PERF_CTRS
```

Creating the additional VMs on the host server

After installing and configuring the initial VM, we created the additional VMs through by using virt-clone on the RHEL platform.

Cloning the VMs on RHEL

Run the following command to clone VM1 to the area reserved for VM2. Repeat as necessary for multiple VMs:

```
# virt-clone --connect qemu:///system --original VM1 --name VM2 --file
/mnt/02-vm-rhel/OS.img --file /mnt/02-vm-rhel/MySQLData.img --file /mnt/02-
vm-rhel/MySQLLog.img --file /mnt/02-vm-rhel/Utility.img --nonsparse
```

Configuring the additional VMs after cloning

Modify the IP addresses in each VM as we discuss in the above section, Configuring networking inside the VM.

Creating the client machines

For the DVD Store scripts, we used a number of client machines to simulate a number of users putting a load on the server. For our client machines, we installed Windows Server 2003 R2 x86 w/ SP2, installed the .NET framework 3.5, and copied the DVD Store files to the appropriate folders. We created a folder we called `C:\ClientShare` to store workload scripts, and shared this folder for access from our controller machine. We installed the .NET 3.5 framework on each client, as the DVD Store test executable requires this. We created a performance counter log on each client machine to track the number of OPM each virtual database server returns. We followed this process for each installation.

Installing Windows Server 2003 R2 Enterprise Edition

1. Boot the system to the Windows Server 2003 installation disk 1.
2. At the Welcome to Set up screen, press Enter to begin the installation.
3. At the Windows Licensing Agreement screen, press F8 to agree to the terms.
4. Press Enter to install in the default location.
5. Select Format the partition using the NTFS file system (Quick), and press Enter to continue.
6. At the Regional and Language Options screen, click Next.
7. Enter Name and Organization, and click Next. Note: Organization is an optional field, while username is required.
8. At the Your Product Key screen, enter your product key, and click Next.
9. At the Licensing Modes screen, click Next.
5. At the Computer Name and Administrator Password screen, type `ClientX` for the computer name, `X` being the client number.
6. Type `Password1` as the new password in both fields, and click Next.

7. At the Date and Time Settings screen, change the Date & Time and Time Zone if appropriate, and click Next.
8. At the Network Settings screen, select Typical settings, and click Next.
9. Accept the domain name defaults, and click Next.
10. After the system restarts automatically, follow the prompt to log into the server.
11. At the Windows Server Post-Setup Security Updates screen, click Finish.
12. Click Yes.
13. At the Manage Your Server screen, click Don't display this page at logon, and close the screen.
14. At the Windows Setup screen, the application prompts you to insert the second installation CD. Insert the appropriate CD, and click OK.
15. At the Windows Server 2003 R2 Setup Wizard Welcome screen, click Next.
16. At the Windows Server 2003 R2 Setup Wizard Summary screen, click Next.
17. When the application prompts you to do so, click Finish to end the installation.

Installing Microsoft .Net Framework 3.5

1. Run dotnetfx35.exe.
2. At the Welcome to Setup screen, select I have read and Accept the terms of the License Agreement, and click Install.
3. When the installation completes, click Exit.

Configuring the IP address

1. Click Start→Control Panel→Network Connection→Local Area Connection.
2. Double-click Internet Protocol (TCP/IP).
3. Select Use the following IP address.
4. Type `192.168.10.X` for the IP address, `X` being the appropriate IP address.
5. Type `255.255.255.0` for the Subnet mask, and click OK.
6. Click OK to close the Local Area Connection Properties.

Creating and setting up a shared folder for DVD Store execution

1. Create a new folder at the root of C:, and name it `ClientShare`.
2. Right-click the folder, and click Properties.
3. Click the Sharing tab.
4. Select Share this folder, and click Permissions.
5. Select Everyone, and click the Allow box beside Full Control.
6. Click OK to close the Permissions for ClientShare window.
7. Click OK to close the ClientShare Properties window.
8. Copy the DVD Store executable client application and all necessary scripts to C:\ClientShare.

Testing procedure

To perform the test, we used a series of batch files and shell scripts, which we executed from a controller machine. We stored batch files respective to each system on that system; either under the/root/Desktop/DS2scripts folder for the server under test and VMs, or the C:\ClientShare folder (client machines), and we used the PsExec, Plink, and WinSCP utilities to coordinate the test procedures amongst the client machine, server machine, virtual machines, and controller. We used simple file sharing and WinSCP to copy files from machine to machine, as we needed.

The testing procedure consisted of the following steps:

1. Execute batch files and shell scripts to clean up prior outputs on client machines, the server physical machine, the server virtual machines, and the controller.
2. Stop the MySQL service.
3. Delete all prior database files.
4. Copy all original database files from the backup utility partition.
5. Start the MySQL service on each server VM.
6. Pause for 1 minute to wait for background tasks to complete before VM shutdown.

7. Shutdown the virtual machines cleanly.
8. Reboot the client machines.
9. Reboot the server under test.
10. Wait for a ping response from all client machines.
11. Wait for a ping response from the physical server machine.
12. Start the VMs under test.
13. Wait 10 additional minutes for any background tasks to complete.
14. Record idle power measurements for 2 minutes.
15. Stop unneeded services on the RHEL guest OS.
16. Mount all necessary partitions.
17. Start the MySQL service on each server VM.
18. Start performance statistics gathering processes on client machines and on the server machine.
19. Start the workload connections.
20. Start the workload ramp up period.
21. Start the workload.
22. Stop the workload.
23. Copy all output files to the controller.

Appendix A – Server configuration information

Figure 4 provides detailed configuration information about the Dell PowerEdge R710 server, and the Red Hat KVM configuration.

Servers	Dell PowerEdge R710 running Red Hat KVM solution
General dimension information	
Height (inches)	3.50
Width (inches)	17.50
Depth (inches)	27.00
U size in server rack (U)	2
Power supplies	
Total number	2
Wattage of each (W)	870
Cooling fans	
Total number	5
Dimensions (h x w) of each	2.5” x 2.5”
Voltage (V)	12
Amps (A)	1.60
General processor setup	
Number of processor packages	2
Number of cores per processor package	4
Number of hardware threads per core	2
System power management policy	OS control
CPU	
Vendor	Intel
Name	Xeon X5550
Stepping	D0
Socket type	LGA1366
Core frequency (GHz)	2.66
L1 cache	32 KB + 32 KB (per core)
L2 cache	1 MB (4 x 256 KB)
L3 cache	1 x 8 MB
Platform	
Vendor and model number	Dell PowerEdge R710
Motherboard model number	PWBYN967
Motherboard revision number	X08
BIOS name and version	Dell 1.1.4
BIOS settings	Virtualization enabled
Memory modules	
Total RAM in system (GB)	72
Number of types of memory modules	1
Vendor and model number	Crucial CT51272BB1339

Servers	Dell PowerEdge R710 running Red Hat KVM solution
Type	PC3-10600 DDR3
Speed (MHz)	1,333
Speed in the system currently running @ (MHz)	800
Timing/latency (tCL-tRCD-iRP-tRASmin)	7-7-7-20
Size (GB)	4
Number of RAM modules	18
Chip organization	Double-sided
Hard disks	
First hard disk type	
Vendor and model number	Fujitsu Limited MBC2073RC
Number of disks in system	2
Size (GB)	73
Buffer size (MB)	16
RPM	15,000
Type	SAS
Second hard disk type	
Vendor and model number	Fujitsu Limited MBB2147RC
Number of disks in system	1
Size (GB)	146
Buffer size (MB)	16
RPM	10,000
Type	SAS
Operating system	
Name	Red Hat Enterprise Linux 5.4
Build number	kernel-2.6.18-159.el5
File system	Ext3
Language	English
Network cards/subsystems	
First network card/subsystem	
Vendor and model number	Intel PRO/1000 Dual Port Adapter D33682
Type	Discrete
Second network card/subsystem	
Vendor and model number	Intel 10 Gigabit AT Server Adapter EXPX9501AT
Type	Discrete
Third network card/subsystem	
Vendor and model number	Intel 10 Gigabit AT Server Adapter EXPX9501AT
Type	Discrete
Fourth network card/subsystem	
Vendor and model number	Broadcom BCM5709c Dual Port Ethernet Controller (unused)
Type	Integrated

Servers	Dell PowerEdge R710 running Red Hat KVM solution
Fifth network card/subsystem	
Vendor and model number	Broadcom BCM5709c Dual Port Ethernet Controller (unused)
Type	Integrated
Optical drive	
Vendor and model number	TEAC DV-28S-VDB
USB ports	
Number	4
Type	2.0

Figure 4: Detailed system configuration information for the test server.

Appendix B – Storage array configuration information

Figure 5 presents detailed configuration information for the storage arrays.

	Dell EqualLogic iSCSI SAN
Arrays	Two Dell EqualLogic PS5000XV arrays
Disks	28 active 15,000RPM SAS disks total for the two shelves
Active storage cache	2 GB (1 GB per PS5000XV)
Number of active storage controllers	2 (1 per array)
Number of passive storage controllers	2 (1 per array)
Number of active storage ports	6 (3 per PS5000XV array)
Firmware revision	v4.1.4
Switch type/ model/firmware revision	One Dell PowerConnect 6248 v2.2.0.3
Disk vendor and model number	16x Seagate ST3146855SS 16x Maxtor ATLAS15K2_147SAS
Disk size (GB)	146
Disk buffer size (MB)	16
Disk RPM	15,000
Disk type	SAS
RAID type	RAID 10

Figure 5: Detailed system configuration information for the storage arrays.

About Principled Technologies

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 you assess how it will fare against its competition, its performance, whether it's ready to go to market, 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.

Principled Technologies, Inc.
1007 Slater Rd., Suite 250
Durham, NC 27703
www.principledtechnologies.com
info@principledtechnologies.com