RED HAT®
ENTERPRISE
VIRTUALIZATION

redhat.

WHITEPAPER

ADVANCED OPERATIONS WITH RED HAT ENTERPRISE VIRTUALIZATION 3.6

EXECUTIVE SUMMARY

The objective of this testing was to establish best practices around Red Hat® Enterprise Virtualization in regards to resource-intensive workloads such as a virtualized database. To that end, we built and tested a Red Hat Enterprise Virtualization cluster backed with solid-state storage area network (SAN) and attached via fibre channel (FC), complete with a TPC-C style workload generator. Using this, we enabled, tested, and graphed several of Red Hat Enterprise Virtualization's new features while a live database was running. Not only did this help establish best practices to use under similar workloads, but it proved settings to avoid in certain circumstances.

TARGET AUDIENCE

This document is intended for virtualization engineers and architects who need a better understanding of Red Hat Enterprise Virtualization's advanced features and the impact of certain configuration settings. To properly evaluate the impact of these settings, you should carefully test them in a similar or identical environment prior to putting changes into production.

CONFIGURATION

The configuration was built with servers based on the Intel "Sandybridge" chip set.

HOST CONFIGURATION

COMPONENTS	DESCRIPTION
Hostnames	Perf{90,91}.perf.lab.bos.redhat.com
CPUs	Intel Xeon E5-2690@2.90GHz (2 socket - 32 CPUs w/ Hyperthreading)
Memory	256GB each
Networking	1xGbE and 1x10GbE each

Each host has two active networks—1GbE and 10GbE. The 1GbE network was used strictly for management, and the 10G network was used for live migration and traffic; a single 1GbE connection will not typically have the bandwidth to handle live migration. The configuration was designed to test virtual machines (VMs) with large memory footprints—the live migration of which typically takes a lot of network bandwidth.

As a best practice, use separate networks for different types of traffic to avoid network contention. This also lets you apply special settings to the networks based on application and traffic needs. In practice, the 10GbE network should be configured for high availability (HA) through network teaming, and different traffic types segregated with virtual LAN (VLAN) tagging.

Red Hat Enterprise Virtualization host and guest configuration

The following software versions were used for testing:

COMPONENTS	VERSION/CONFIGURATION
Red Hat Enterprise Virtualization Manager	3.6.3.2-0.1.el6
Red Hat Enterprise Linux® Hypervisor	Red Hat Enterprise Linux 7.2 9.el7
Kernel version	3.10.0 - 327.el7.x86_64
KVM version	2.3.0 - 31.el7_2.8
Libvirt version	libvirt-1.2.17-13.el7_2.2
Vodafone Secure Device Manager (VDSM) version	vdsm-4.17.23-0.el7ev
SPICE version	0.12.4 - 15.el7
VM Red Hat Enterprise Linux kernel version	3.10.0-327.el7.x86_64
Database	Oracle 12.1.0.2 (nonpluggable database mode)

STORAGE CONFIGURATION

This testing was run with Violin 6616 solid-state fibre channel (FC) storage. The storage unit has a capacity of 16TB, divided into a number of 200GB logical unit numbers (LUNs). The storage has 4 x 8Gb connectors, which connect to the host with Emulex LPe16002-M6 PCIe 2-port 16Gb FC adapters. This gives the host eight active paths to each LUN.

An example of the multipath output from the host is shown below.

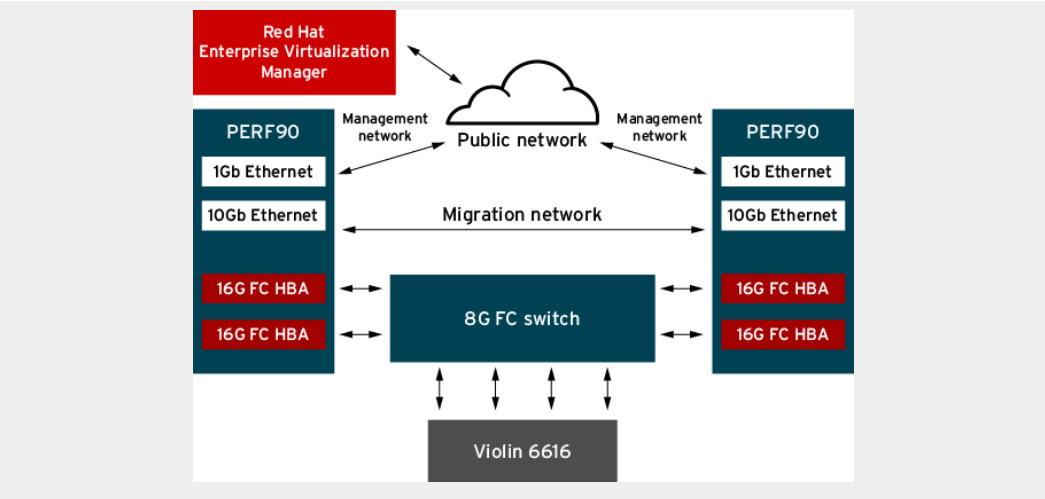
Each LUN – eight active paths (example of one LUN):

```
36001b97075ebc57f75ebc57f97aa8372 dm-24 VIOLIN  ,SAN ARRAY
size=200G features='0' hwhandler='0' wp=rw
|-+- policy='service-time 0' prio=1 status=active
| `- 8:0:4:3  sddb 70:144  active ready running
|-+- policy='service-time 0' prio=1 status=enabled
| `- 8:0:5:3  sdea 128:32  active ready running
|-+- policy='service-time 0' prio=1 status=enabled
| `- 8:0:8:3  sdez 129:176 active ready running
|-+- policy='service-time 0' prio=1 status=enabled
```

```
 | `- 8:0:9:3  sdfy 131:64  active ready running
 |-+- policy='service-time 0' prio=1 status=enabled
 | `- 7:0:4:3  sdf  8:80    active ready running
 |-+- policy='service-time 0' prio=1 status=enabled
 | `- 7:0:5:3  sdae 65:224  active ready running
 |-+- policy='service-time 0' prio=1 status=enabled
 | `- 7:0:8:3  sdbd 67:112  active ready running
 `-+- policy='service-time 0' prio=1 status=enabled
   `- 7:0:9:3  sdcc 69:0    active ready running
```

Figure 1 represents the lab configuration:

Figure 1 (lab diagram)



DATABASE WORKLOAD

An Oracle OLTP application was used for this testing. The workload was based on the industry-standard TPC-C workload; due to resource limitations, testing was not done according to the TPC specifications. However, the workload was suitably modified to run on file systems as it would in a production environment. The shared memory size for each test was based on VM size. The system and database parameters were configured for optimal performance. The test harness user count variable ran a number of simulated users.

Each test began with a fresh database build, and after a short warm-up run, the workload was run with 10, 20, 40, 80, and 100 users. This captured performance from low to high CPU utilization, and as the user count increased, it also changed the I/O access pattern. In each test, approximately 70% of each VM's memory was used for an Oracle shared memory segment. The number of transactions per minute (TPM) for each run was recorded and used as a metric for comparison.

TEST PLAN

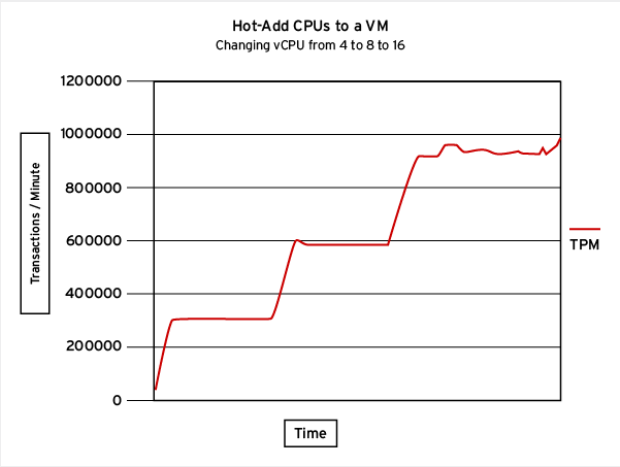
The test plan was to run the database online transaction processing (OLTP) workload in the VM, use the different options available in Red Hat Enterprise Virtualization to configure the CPU and storage, and measure the impact on the performance of the workload. Some of the tests were also used to showcase features of Red Hat Enterprise Virtualization, including live migration, hot CPU add, and non-uniform memory access (NUMA) support.

CPU HOT ADD FEATURE

The ability to dynamically add CPUs to a running system is a new feature in Red Hat Enterprise Virtualization 3.6. This allows users to increase the number of virtual CPUs (vCPUs) of a VM without downtime, which is useful for virtual machines that require more CPUs and high uptime—not to mention the ability to scale up on demand.

In this test, the workload was started in a four vCPU VM. In the middle of the run, the number of vCPUs was increased from 4 to 8 to 16 vCPUs. The Transactions per Minute (TPM) increased dynamically from 300K to 600K, then to 900K as the vCPU count increased. Figure 2 shows the change tracked during the run.

Figure 2 (hot add CPU)



LIVE MIGRATION TUNABLES

Live migration is a cornerstone feature of virtualization that allows users to move VMs across hosts without downtime and without users losing their host connection. Red Hat Enterprise Virtualization has features that allow users to control migration behavior.

MIGRATION WITH MIGRATION_MAX_BANDWIDTH PARAMETER

First, the default bandwidth of the migration is capped at 32MB/s. If the VMs are large, this value results in very long migration times. Migration value can be configured to desired bandwidth by adjusting the parameter **`migration_max_bandwidth`** in the file **`/etc/vdsm/vdsm.conf`** on the host. If the value is set to zero, the entire bandwidth of the migration network will be used.

To understand the impact of changing this parameter, we migrated a large VM (208GB memory, 32 vCPUs) with no workload running.

MIGRATION_MAX_BANDWIDTH VALUE	MIGRATION TIME
Default	35 minutes, 20 seconds
0 (Full bandwidth available for migration)	1 minute, 21 seconds

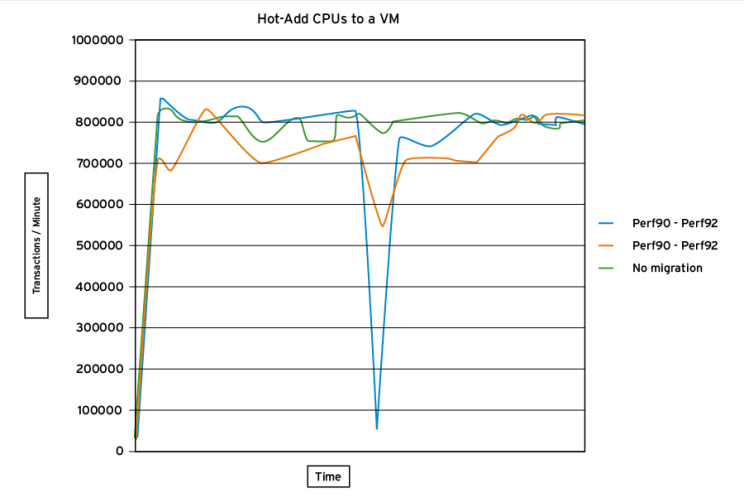
Once full migration bandwidth was set, we ran the workload inside the virtual machine, and after populating the Oracle shared segment, we migrated the VM with the auto-converge option. The test was run once without migration for a baseline performance result; then the test was run moving the VM from Perf90 to Perf92; and again moving the VM back from Perf92 to Perf90.

The migration timings and TPM were as follows:

AVERAGE TPM	MIGRATION TIME
779700	
732038	01:29.00
750671	00:56.00

The average TPM shows a slight drop in the migration tests. The actual TPM captured during all three tests is shown in Figure 3:

Figure 3 (VM migration | Standard maximum transmission unit MTU)



LIVE MIGRATION WITH DIFFERENT NETWORK FRAME SIZE

Another tunable network parameter is the frame size. While the default network frame size is 1,500 bytes, we set the frame size to 9,000 (jumbo frames) for the next test, and used a VM with 128GB memory, 16 vCPUs, and 96GB huge pages with an idle Oracle instance. The results were as follows for idle guest migration:

MIGRATION	1,500 MTU	9,000 MTU	% DIFFERENCE
Perf90 > Perf92	54 seconds	45 seconds	20
Perf92 > Perf90	52 seconds	44 seconds	18.18

The results show that a larger frame size aids migration of an idle VM. When we repeated the test with a workload running in the VM, there was some improvement in the migration times—but considering the workload and activity in the VM, performance difference was not consistent for migration in both directions.

MIGRATION	1,500 MTU	9,000 MTU	% DIFFERENCE
Perf90 > Perf 92	01:29.00	01:15.00	18.67
Perf92 > Perf90	00:56.00	00:53.00	5.66

NUMA OPTIMIZATION

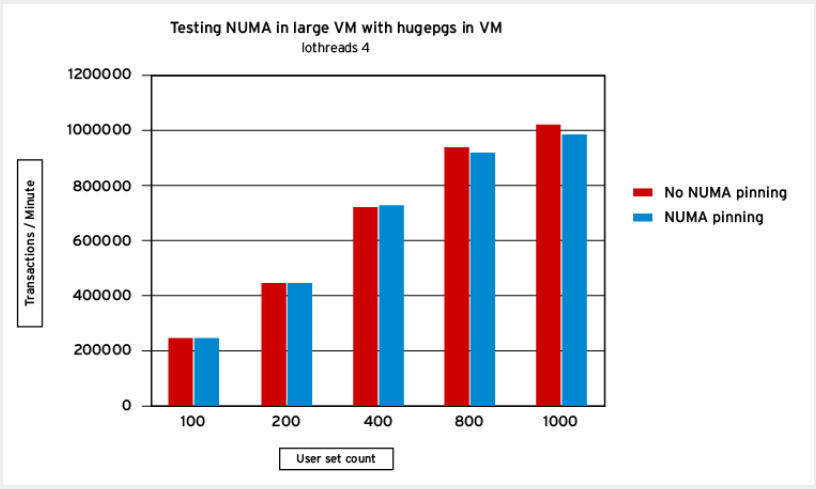
In large hosts, memory is physically laid out across the sockets to prevent single memory bank bottlenecks. This allows for large-system scaling, but can increase memory access latency in systems with four or more sockets. With proper NUMA optimization techniques, a user can place the processes and their memory on the same socket or adjoining sockets, reducing the memory access latency significantly and adding performance gains on large systems.

Red Hat Enterprise Virtualization offers various options for users to place the VMs in specific NUMA nodes for greater performance. However, users need to understand that the gains are only possible when the VM is properly aligned. For example, a very large VM that spreads across multiple sockets will always incur greater overhead as it accesses remote memory banks. VMs that can fit into a single NUMA node are perfect candidates for this tuning. Depending on the size of the host, larger VMs can also be configured to use two or more adjacent NUMA nodes to reduce remote memory access latency.

NOTE: NUMA optimization is currently available for a VM only if migration is turned off.

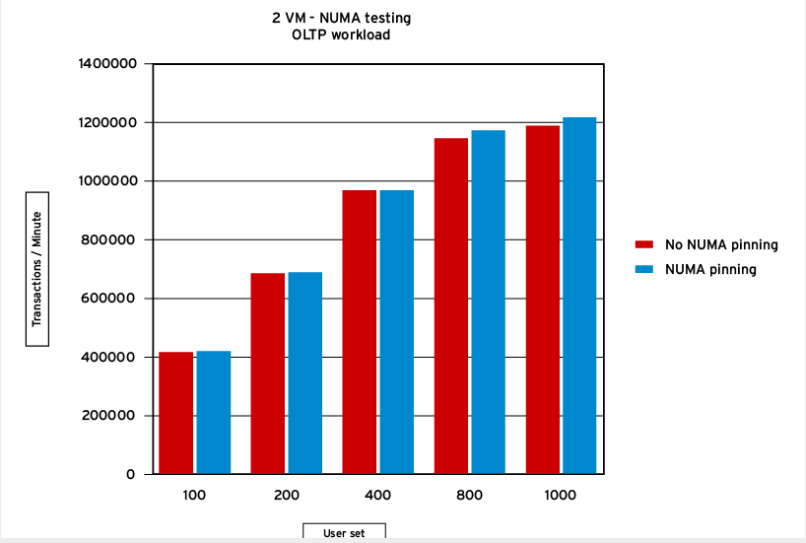
Figure 4 shows that using NUMA with a single large VM running an OLTP workload with huge pages does not show any improvement—because the VM is using both NUMA nodes, and defining NUMA nodes in the VM did not help the workload performance. If the VM workload is NUMA-aware, define NUMA in the VM, and align it with the host NUMA nodes.

Figure 4 (NUMA and huge pages)



With 2 VMs, using a NUMA-preferred setting for each VM and aligning it to a single NUMA node helps improve performance.

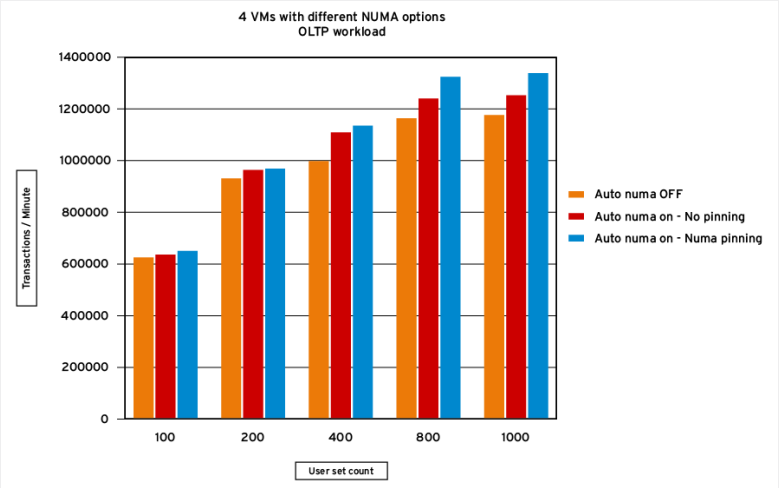
Figure 5 (NUMA and OLTP)



We see the best advantage with four VMs. Worth noting, though, is that Red Hat Enterprise Linux® 7 has autoNUMA enabled by default. This feature in the kernel automatically aligns workloads—including VMs—to NUMA nodes and continues to dynamically align NUMA as the workloads run.

Using manual pinning, however, you can see increased performance improvement.

Figure 6 (NUMA OLTP 2)

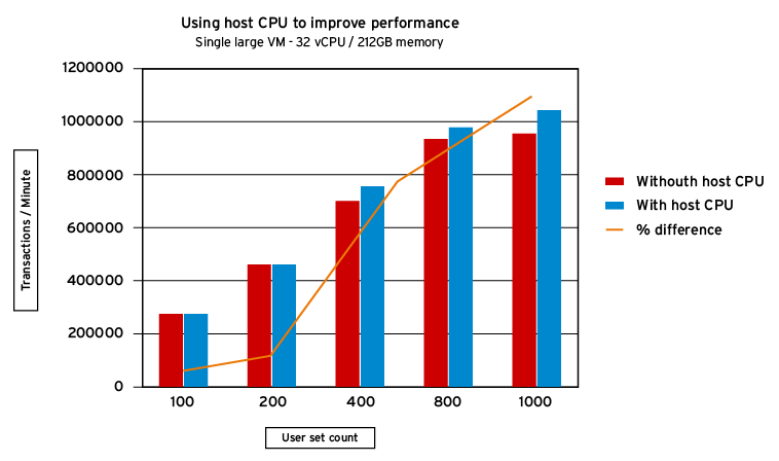


USING HOST CPU PASS-THROUGH FOR ADDED PERFORMANCE

For VMs, there is an option to "Pass-through host CPU" under the "Host" tab. This matches the vCPUs in the virtual machine to the host CPU exactly—instead of simply matching the features of the CPU family. This can be useful for CPU-centric workloads. If the workload is dependent on other factors like I/O, enabling this feature might not help performance.

This feature can be enabled only if VM migration is turned off—so using this feature largely depends on the use case. If the application can benefit from the performance gain and does not need to be highly available, this flag can be used. Figure 7 shows a single VM running the database workload with and without the flag. The performance difference is tracked on the second Y-axis using the yellow line on the graph. As the numbers show, performance increases with higher user count with the greatest CPU utilization.

Figure 7 (single large CPU)

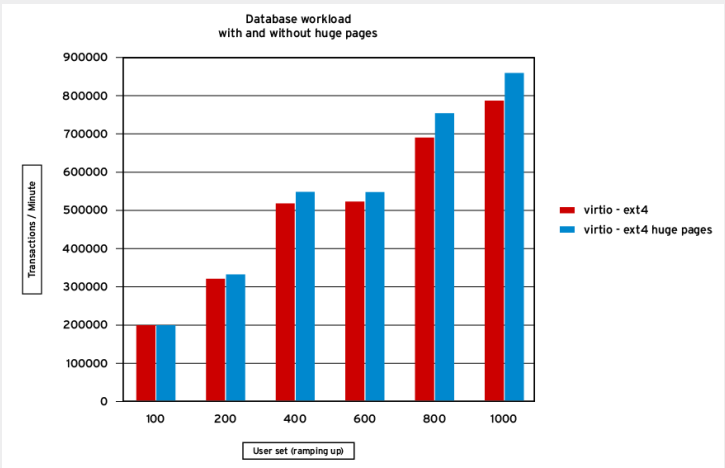


USING HUGE PAGES IN THE VM

Red Hat Enterprise Linux kernel supports 2MB-size huge pages. The VM is backed by transparent huge pages on the host. But users can manage non-transparent huge pages inside the VM with applications that require it. Using huge pages helps reduce the translation lookaside buffer (TLB) misses (which improves performance) and also wires the application memory to prevent swapping (beneficial when there is memory pressure in the virtual machine).

Figure 8 shows that using 2MB huge pages to back an Oracle SGA improved performance by approximately 10% at higher user count.

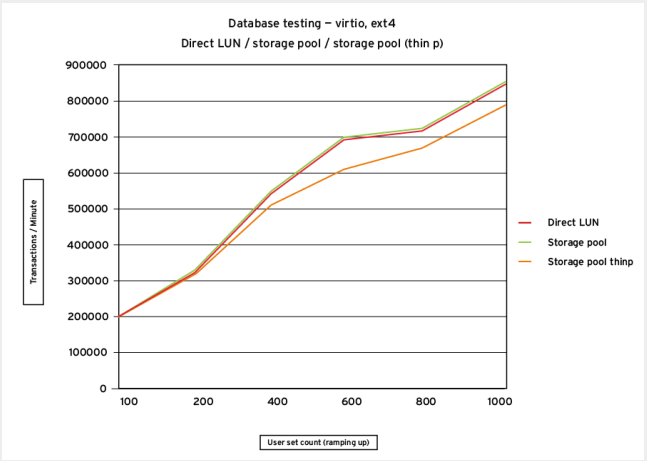
Figure 8 (huge pages)



USING PRE-ALLOCATED DISKS FOR PERFORMANCE

Storage can be attached to virtual machines as either pre-allocated or thin-provisioned disks. If the workload has a growing data set, pre-allocated disk performance can be significantly better than that of thin-provisioned disks, directly proportional to the rate of file growth. The example in Figure 9 shows that as the user set increases, so does the number of transactions per minute. However, the pre-allocated storage pool handles a higher number of transactions per minute as compared to the thin-provisioned storage.

Figure 9 (direct LUN)



ADDITIONAL DOCUMENTS

All Red Hat Enterprise Virtualization-related documentation:

https://access.redhat.com/documentation/en/red-hat-enterprise-virtualization/

Red Hat Enterprise Virtualization 3.6 Installation Guide

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Virtualization/3.6/html/Installation_Guide/index.html

Red Hat Enterprise Virtualization 3.6 Administration Guide

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Virtualization/3.6/html/Administration_Guide/index.html

ABOUT RED HAT

Red Hat is the world's leading provider of open source software solutions, using a community-powered approach to provide reliable and high-performing cloud, Linux, middleware, storage, and virtualization technologies. Red Hat also offers award-winning support, training, and consulting services. As a connective hub in a global network of enterprises, partners, and open source communities, Red Hat helps create relevant, innovative technologies that liberate resources for growth and prepare customers for the future of IT.

NORTH AMERICA
1 888 REDHAT1

EUROPE, MIDDLE EAST, AND AFRICA
00800 7334 2835
europe@redhat.com

ASIA PACIFIC
+65 6490 4200
apac@redhat.com

LATIN AMERICA
+54 11 4329 7300
info-latam@redhat.com

facebook.com/redhatinc
@redhatnews
linkedin.com/company/red-hat

redhat.com
inc0383299_0516