



Red Hat Reference Architecture Series

Comparing the Performance of Red Hat® Enterprise Linux® 5 and Red Hat® Enterprise Linux® 6 using the AIM7 Multiuser Benchmark

| | |
|--------------------------------------|------------------------------------|
| AIM7 Benchmark | |
| EXT3 / EXT4 File Systems | |
| Red Hat® Enterprise Linux 5.5 | Red Hat® Enterprise Linux 6 |
| HP DL980 G7 (64 cores) | |

Version 1.1
December 2010





Comparing the Performance of Red Hat® Enterprise Linux® 5 and Red Hat® Enterprise Linux® 6 using the AIM7 Multiuser Benchmark

1801 Varsity Drive™
Raleigh NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588

Research Triangle Park NC 27709 USA

Linux is a registered trademark of Linus Torvalds. Red Hat, Red Hat Enterprise Linux and the Red Hat "Shadowman" logo are registered trademarks of Red Hat, Inc. in the United States and other countries.

UNIX is a registered trademark of The Open Group.

All other trademarks referenced herein are the property of their respective owners.

© 2010 by Red Hat, Inc. This material may be distributed only subject to the terms and conditions set forth in the Open Publication License, V1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

The information contained herein is subject to change without notice. Red Hat, Inc. shall not be liable for technical or editorial errors or omissions contained herein.

Distribution of modified versions of this document is prohibited without the explicit permission of Red Hat Inc.

Distribution of this work or derivative of this work in any standard (paper) book form for commercial purposes is prohibited unless prior permission is obtained from Red Hat Inc.

The GPG fingerprint of the security@redhat.com key is:
CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E



Table of Contents

| | |
|--|----|
| 1 Executive Summary..... | 4 |
| 2 Testbed Configuration..... | 6 |
| 2.1 Hardware..... | 6 |
| 3 AIM7 Multiuser Benchmark..... | 7 |
| 4 Test Methodology..... | 8 |
| 4.1 CPU Scheduler CFS and Ticketed Spinlocks..... | 8 |
| 4.2 VM Scalability: Split-LRU and Transparent Hugepages..... | 8 |
| 4.3 Disk I/O: BDI Flush, MPIO..... | 8 |
| 4.4 File System Scalability: EXT3 / EXT4 / XFS..... | 9 |
| 4.5 RHEL 6 tuned-adm Infrastructure..... | 9 |
| 5 Benchmark Results..... | 10 |
| 5.1 File Server Performance..... | 10 |
| 5.1.1 File Server Workfile..... | 11 |
| 5.2 Database Performance..... | 12 |
| 5.2.1 DBserver Workload Workfile..... | 13 |
| 5.3 Compute Performance..... | 14 |
| 5.3.1 Compute Server Workload Workfile..... | 15 |
| 5.4 Multiuser Shared Performance..... | 16 |
| 5.4.1 Multiuser Shared Workload Workfile..... | 17 |
| 6 Summary..... | 18 |



1 Executive Summary

This paper uses the AIM7 Multiuser Benchmark to compare the performance of Red Hat Enterprise Linux (RHEL) version 5.5 to the performance of RHEL version 6 for a variety of workload mixes. Figures 1, 2 and 3 show the improvement in the performance of RHEL 6 over RHEL 5.5.

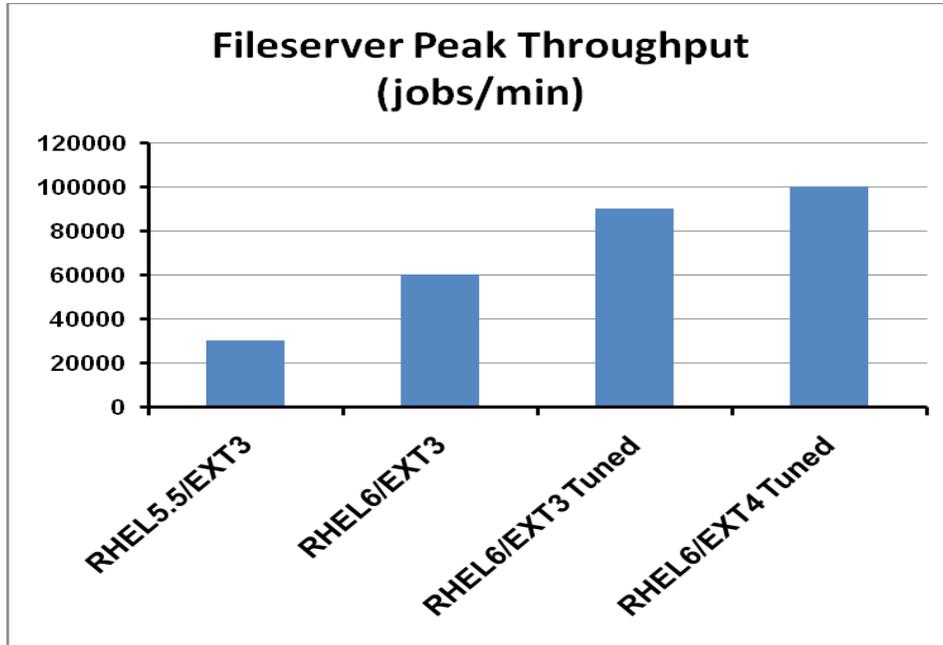


Figure 1: AIM7 File Server Performance

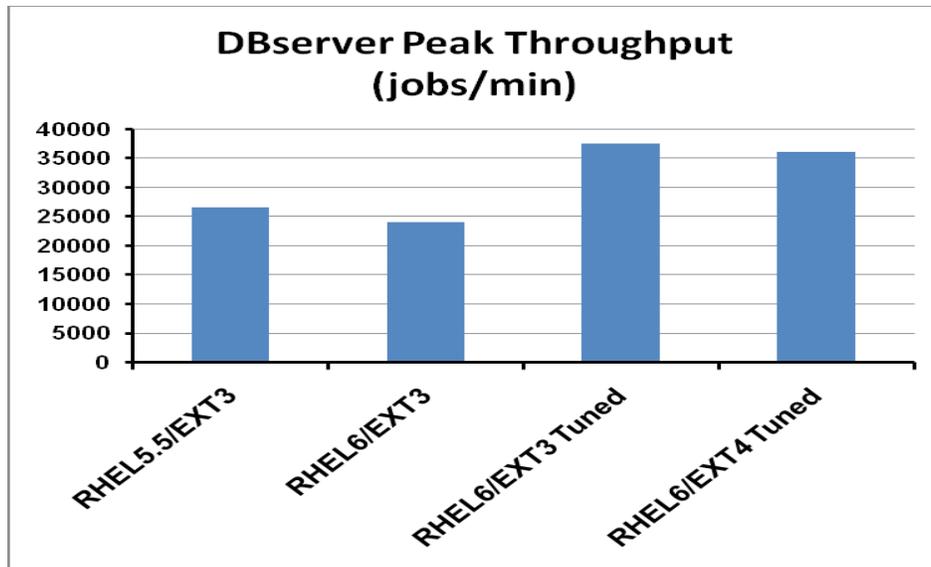


Figure 2: AIM7 Database Performance

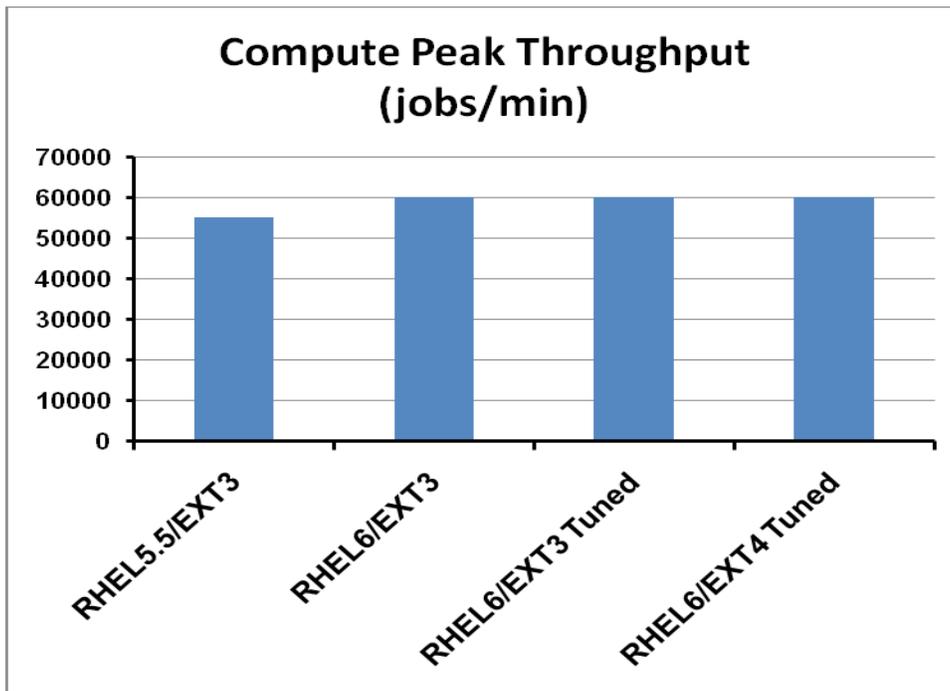


Figure 3: AIM7 Compute Performance

The advantage is clear for most AIM7 workloads and is most evident on workloads that are heavily weighted toward disk I/O such as the database and file server workloads. The results share performance improvements moving from the RHEL 5 default file system of EXT3 to the RHEL 6 default of EXT4.

The results also highlight the performance gains that can be achieved in tuning RHEL 6 to further optimize file server I/O loads for high-end x86_64 servers, in this case an HP DL980 64-core machine with 256 GB of memory and 15 dual port fiber channel HBAs connected to 20 P2000 G3 RAID controllers.



2 Testbed Configuration

The following configuration details describe the testbed configured by HP to fully characterize each dimension of performance.

2.1 Hardware

| Hardware | Specifications |
|---|--|
| 1 x HP DL980 G7 | Eight Socket, Intel® EX Xeon® 7560 64 (8x8) CPU, 2.27 GHz, 24GB cache (HT disabled) |
| | 256GB RAM (64 4GB DDR3 1333 MHz chips) |
| | 15 x StorageWorks 82Q PCIe dual port 8Gb FC HBA |
| 20 x HP StorageWorks P2000 G3 MSA Array | Single 8Gb Controller 24 x 146GB 15K 6Gb SAS Disks (480 total) Firmware: TS200R021 |
| 1 x HP StorageWorks 8/80 SAN Switch | Firmware: Fabric OS V6.2.0d 70 Populated Ports |

Table 1: Hardware Configuration



**Figure 4: HP DL980 G7,
64 Cores, 256 GB RAM**



3 AIM7 Multiuser Benchmark

The AIM Multiuser Benchmark, also called the AIM Benchmark Suite VII or AIM7, is a job throughput benchmark widely used by UNIX computer system vendors.

The original code was from AIM Technology, Inc., who licensed it to others. Caldera International, Inc., bought the license and released the source code for Suites VII and IX under the GPL.

AIM7 is a C program that forks many processes which represent jobs or users. Each job is composed of as much as 53 assorted tests blended to create a workload that exercises a different aspect of the operating system such as disk-file operations, process creation, user virtual memory operations, pipe I/O, and compute-bound arithmetic loops. The test proportions are specified via a *workfile* used to define the workload.

A complete AIM7 benchmark run is comprised of a series of independent runs of the selected workload at different requested loads, specified in terms of a number of jobs. Each individual run executes until all of its jobs have completed the set of randomly ordered tests specified by the workfile. A number of metrics describing the results at that load point are reported including the rate at which the system under test was able to complete the work, or the number of jobs completed per minute. The metric of greatest interest is peak system throughput, the throughput obtained at some requested load (in terms of a number of jobs per minute) that was greater than the throughput obtained for all other requested loads. I.E., a given system will have a peak number of tasks N at which the jobs per minute is maximized. Either N, or the value of the jobs per minute at N, is considered the peak system throughput.

The number of requested jobs per load point defaults to increasing by one, however using the *adaptive* option as was done in these tests, the number of requested jobs can increase by much more than one.

The AIM suite provides several examples of these workloads including simulations of databases, file servers, and compute servers. As mentioned the workload can be adjusted by altering test weight or modifying the test mix in the workfile.

This reference architecture characterizes the AIM7 mix for compute loads (CPU scalability), shared users (VM and file systems), database workload (mix weighted toward disks random I/O), and file server (mix weighted towards sequential and random disk I/O).



4 Test Methodology

The workloads will continue to add user processes where each process runs a mix of operations. A metric for the number of jobs per minute (jobs/min) represents the throughput for the system under test (SUT). A balanced system should allow server memory, disks, and file systems to be added to the SUT until the number of processes exceeds the number of jobs/min. This metric is called the *AIM7 crossover-point* or when sustained throughput equals the jobs/min. Historically this was considered an excellent measure of performance because many times a system's expandability does not match the hardware level and its ability for the OS to scale. The ramifications of not configuring an x86_64 server of this size include potential bottlenecks in CPU scheduling, virtual memory (VM) and disk scalability as well as file system limitations.

RHEL 6 has significantly improved upon scalability for large x86_64 systems such as the 64 processor, HP DL980 G7 based on 8 sockets with hyperthreads disabled.

4.1 CPU Scheduler CFS and Ticketed Spinlocks

The RHEL 6 scheduler uses ticketed spinlocks for scalability on x86_64 large SMP system and to ensure Completely Fair Scheduling (CFS) as well as avoiding process/NUMA node starvation.

4.2 VM Scalability: Split-LRU and Transparent Hugepages

RHEL 6 is NUMA aware and will place processes and their associated memory on a NUMA node to ensure lowest memory latency, best response time and therefore the highest possible throughput on the HP DL980 G7 server.

Additionally, RHEL 6 implements a new split LRU VM algorithm that separates the Linux page cache from anonymous memory and locking reduction done by developers from HP and Red Hat.

RHEL 6 also implements Transparent Hugepages (THP) to dynamically allocate x86_64 2MB pages when available compared to the base page size for x86_64 which architecturally is 4KB.

4.3 Disk I/O: BDI Flush, MPIO

RHEL 6 replaced pdflush for processing buffered writeback, opting to flush threads using Backing Device Information (BDI) allowing for linear scalability as LUNs are presented to the OS.

RHEL 6 continues to implement Linux native multipath (MPIO) for high availability.



4.4 File System Scalability: EXT3 / EXT4 / XFS

RHEL 6 will support standard EXT3 file systems and either EXT4 or XFS as enhancements in scalability for large file system volumes. EXT4 and XFS have improved logging and recovery for files greater than 1 TB which can be an order of magnitude faster than EXT3. This reference architecture focuses on EXT3 to compare against a RHEL 5.5 based system with EXT3. The file system is tuned to enhance performance using the I/O elevator=deadline scheduler and disabling the files system I/O barriers which are not needed for enterprise storage. Although not used to produce the results presented in this document, this tuning may be accomplished using the *tuned-adm* infrastructure described in the following section.

4.5 RHEL 6 *tuned-adm* Infrastructure

RHEL 5 introduced the utility *ktune* to adjust common system control (sysctl) parameters in RHEL 5 for optimizing CPU, memory, network and I/O for throughput or latency. In RHEL 6, Red Hat extended the utility to include:

- *tuned-adm* list
- available profiles:
 - latency-performance
 - enterprise-storage
 - default
 - throughput-performance
 - laptop-ac-powersave
 - laptop-battery-powersave
- optimizations in for latency / throughput and enterprise storage including:
 - adjusting the I/O elevator=deadline (versus CFQ default)
 - altering the powersave mode from OnDemand to Performance
 - setting the VM reclaim parameters for dirty_ratio back to the RHEL 5 value of 40 (RHEL 6 adjusted default to 20)
- additional optimization throughput and enterprise-storage also adjusts:
 - block device and LVM read ahead values increased by a factor of 4
 - scheduler tunable quantum back to RHEL 5 default of 10 milliseconds (RHEL 6 default quantum is 4 milliseconds)
- additional optimization for enterprise-storage includes remounting the file system using “-o barrier=0” (assumes enterprise storage). Future updates to RHEL 6 may do this automatically. See */proc/mount* to view the barrier settings on the server.



5 Benchmark Results

5.1 File Server Performance

RHEL 6 AIM7 file server results show the most dramatic improvement over RHEL 5.5 as the nature of the file server workload mix shows the impact of scaling I/O intensive jobs over the 120 mount points.

RHEL 5 achieves 90% of its peak throughput at 30,000 jobs/min with a relatively small load. Each stream will compete for kernel I/O resources and ultimately begin pushing I/O through the system memory.

The default performance of RHEL 6 peaks at approximately 69,000 jobs/min at 42,000 jobs. This highlights how RHEL 6 scales better than RHEL 5 on a difficult file server workload reaching a crossover-point at 65,000 jobs, approximately 225% beyond RHEL 5.5 at 29,000 jobs.

When tuning is applied, the I/O elevator is switched from CFQ to deadline (elevator=deadline) and the file systems I/O barrier code is disabled, a valid optimization for enterprise storage in RHEL 6. Figure 5 emphasizes how the results of tuned performance far exceed those of the defaults.

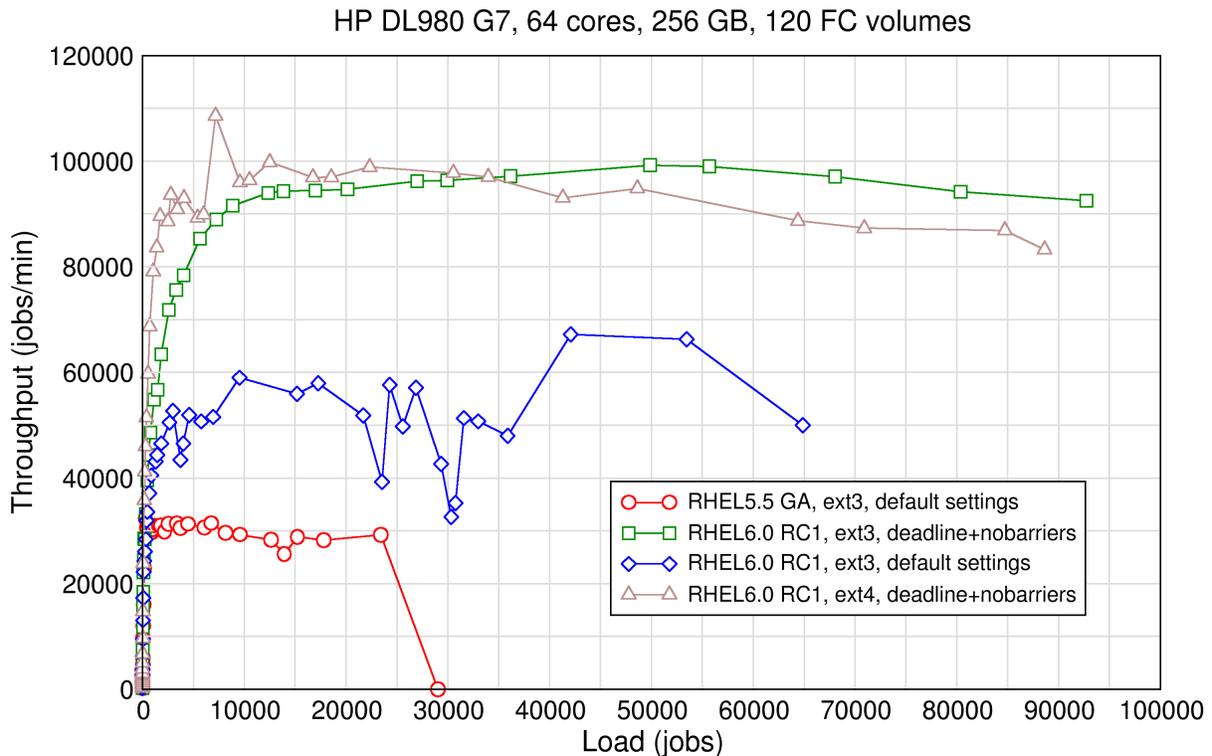


Figure 5: AIM7 File Server Performance Results



5.1.1 File Server Workfile

FILESIZE: 10M
POOLSIZE: 20M

| Weight | Tests |
|--------|---------------|
| 20 | add_int |
| 20 | add_long |
| 20 | add_short |
| 20 | creat-clo |
| 20 | dir_rtns_1 |
| 30 | disk_cp |
| 30 | disk_rd |
| 30 | disk_rr |
| 30 | disk_rw |
| 30 | disk_src |
| 30 | disk_wrt |
| 10 | div_int |
| 10 | div_long |
| 10 | div_short |
| 10 | jmp_test |
| 20 | link_test |
| 40 | mem_rtns_1 |
| 10 | mem_rtns_2 |
| 20 | misc_rtns_1 |
| 10 | mul_int |
| 10 | mul_long |
| 10 | mul_short |
| 20 | ram_copy |
| 10 | signal_test |
| 30 | sort_rtns_1 |
| 30 | string_rtns |
| 5 | sync_disk_cp |
| 5 | sync_disk_rw |
| 5 | sync_disk_wrt |
| 10 | tcp_test |
| 40 | udp_test |

Table 2: File Server Workfile



5.2 Database Performance

The RHEL 6 AIM7 database results show a larger improvement over RHEL 5.5 especially with the jobs/min increase in the tuned result. The nature of the database workload mix adds more random I/O intensive jobs over the 120 mount points. The RHEL 6 default is affected by the conservative setting of I/O barriers for EXT3 and EXT4 and reaches a peak at only 240,000 jobs/min. RHEL 5 does not use file system I/O barriers and achieves a 10% higher peak at 250,000 jobs/min.

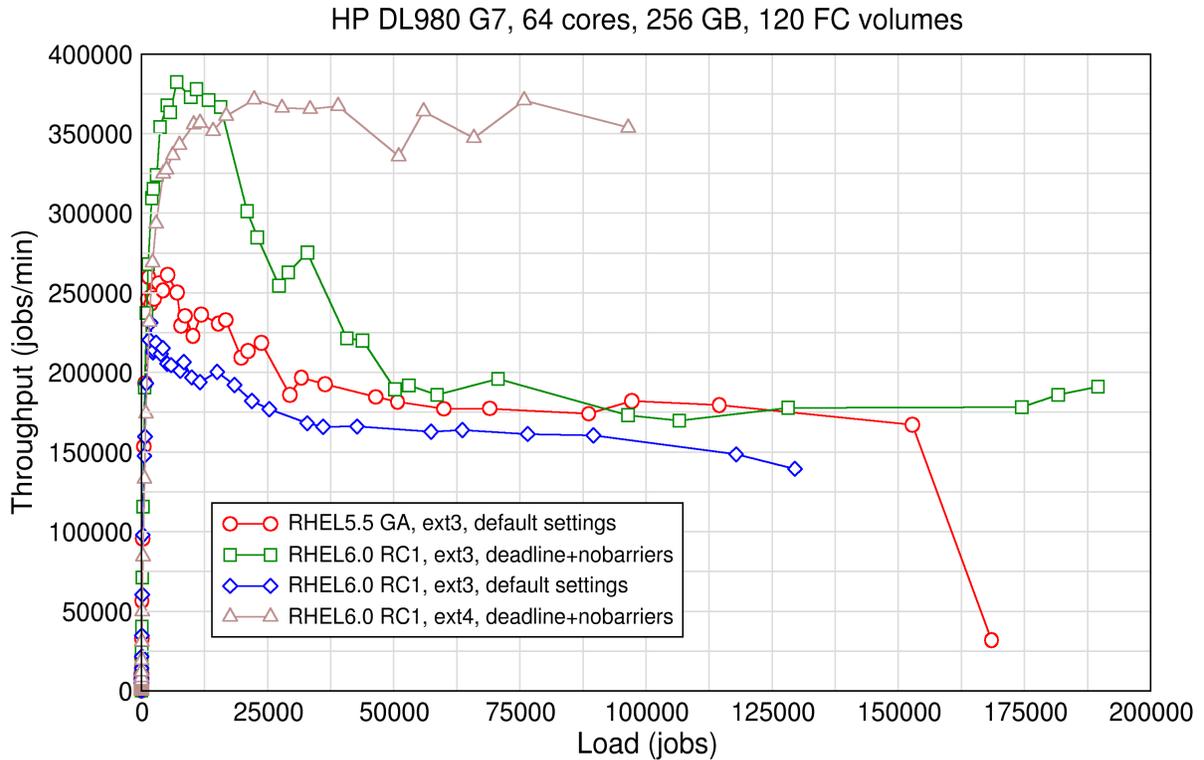


Figure 6: AIM7 Database Performance Results

When disabling the I/O barriers, RHEL 6 is able to take advantage of the new BDI code (generating and sustaining much higher I/O rates) and better scalability from ext4 with selective tuning.



5.2.1 DBserver Workload Workfile

FILESIZE: 1M

POOLSIZE: 25M

| Weight | Tests |
|--------|------------------|
| 20 | add_int |
| 20 | add_long |
| 20 | add_short |
| 40 | disk_rd |
| 40 | disk_rr |
| 10 | div_int |
| 10 | div_long |
| 10 | div_short |
| 10 | jmp_test |
| 40 | mem_rtns_1 |
| 40 | mem_rtns_2 |
| 10 | mul_int |
| 10 | mul_long |
| 10 | mul_short |
| 40 | page_test |
| 20 | ram_copy |
| 40 | shared_memory |
| 30 | sieve |
| 30 | sort_rtns_1 |
| 10 | stream_pipe |
| 30 | string_rtns |
| 30 | sync_disk_rw |
| 30 | sync_disk_update |

Table 3: DBserver Workfile



5.3 Compute Performance

The RHEL 6 AIM7 compute result is a good regression test for comparison to the established leadership result with RHEL 5.5. Both RHEL 5 and RHEL 6 schedule jobs across the 64-core machine and are aware of the eight nodes of Non-Uniform Memory Access (NUMA) such that the box is saturated within several thousand jobs. Note that each job is actually a Linux process running the workload mixture to competition. At 200,000 jobs (processes), RHEL 6 shows an improvement of approximately 9% by scheduling up to 600,000 jobs/min compared to RHEL 5.5 which was able to achieve 550,000 jobs/min.

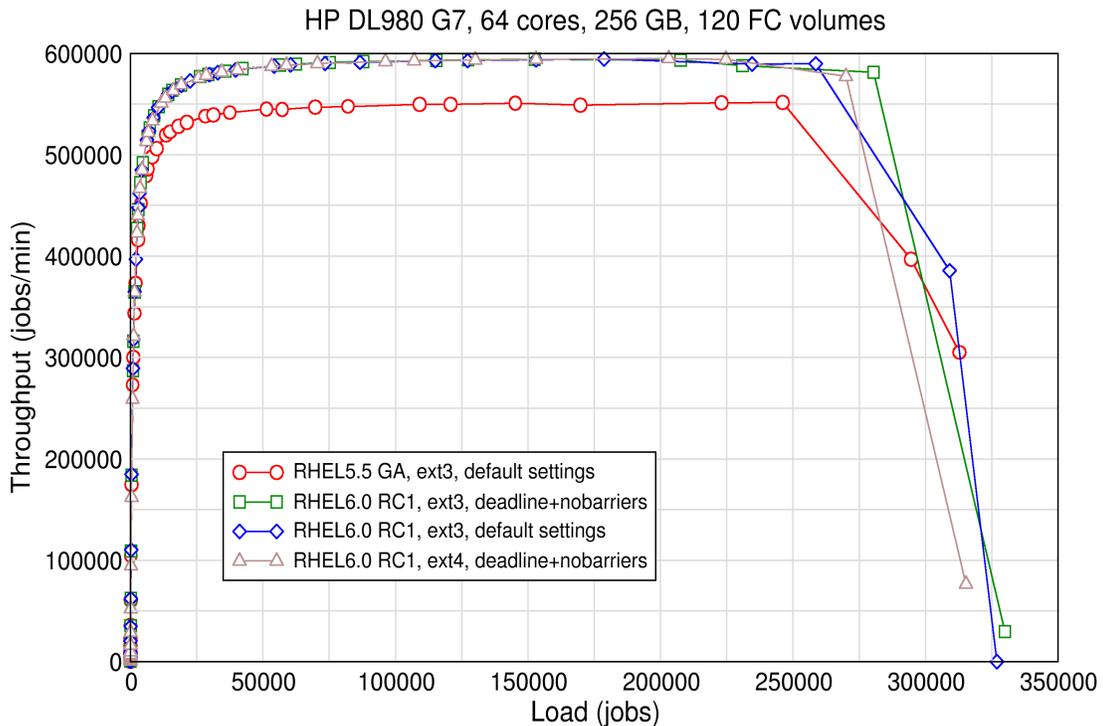


Figure 7: AIM7 Compute Performance Results

RHEL 6 benefits from the split LRU VM algorithm that separates the Linux page cache from anonymous memory and locking reduction as well as its ability to use THP, 2MB pages dynamically allocated at run-time if the virtual memory is available. THP will continue to honor a local NUMA memory policy that ensures a process uses fastest memory possible, memory that is local to the NUMA node if present at process creation. RHEL 5.5 can implement hugepages only if an application is coded to take advantage of them (e.g., large database or Java application).



5.3.1 Compute Server Workload Workfile

FILESIZE: 100K
POOLSIZE: 250M

| Weight | Tests |
|--------|---------------|
| 50 | add_double |
| 30 | add_int |
| 30 | add_long |
| 10 | array_rtns |
| 10 | disk_cp |
| 30 | disk_rd |
| 10 | disk_src |
| 20 | disk_wrt |
| 40 | div_double |
| 30 | div_int |
| 50 | matrix_rtns |
| 40 | mem_rtns_1 |
| 40 | mem_rtns_2 |
| 50 | mul_double |
| 30 | mul_int |
| 30 | mul_long |
| 40 | new_raph |
| 40 | num_rtns_1 |
| 50 | page_test |
| 40 | series_1 |
| 10 | shared_memory |
| 30 | sieve |
| 20 | stream_pipe |
| 30 | string_rtns |
| 40 | trig_rtns |
| 20 | udp_test |

Table 4: Compute Server Workfile



5.4 Multiuser Shared Performance

The RHEL 6 AIM7 shared mix result is similar to that of the compute mix but has less compute only tasks and instead features more operations that represent a software developer environment which exercise most Linux system calls and have an increased amount of disk and loop-back network I/O. For this load, RHEL 5.5 peaks at almost 350,000 jobs/min and reaches crossover at approximately the same level. The RHEL 6 default is again penalized by the default use of I/O barriers achieving only 240,000 jobs/min. The HP and Red Hat Performance teams will continue analysis of the results which is believed to be affected by the fact that the RHEL 6 process quantum has been reduced from ten milliseconds to four. While this helps reduce scheduler latency for some workloads, it has been shown that adjusting it back to ten milliseconds has a positive improvement in these AIM7 shared workload mixes.

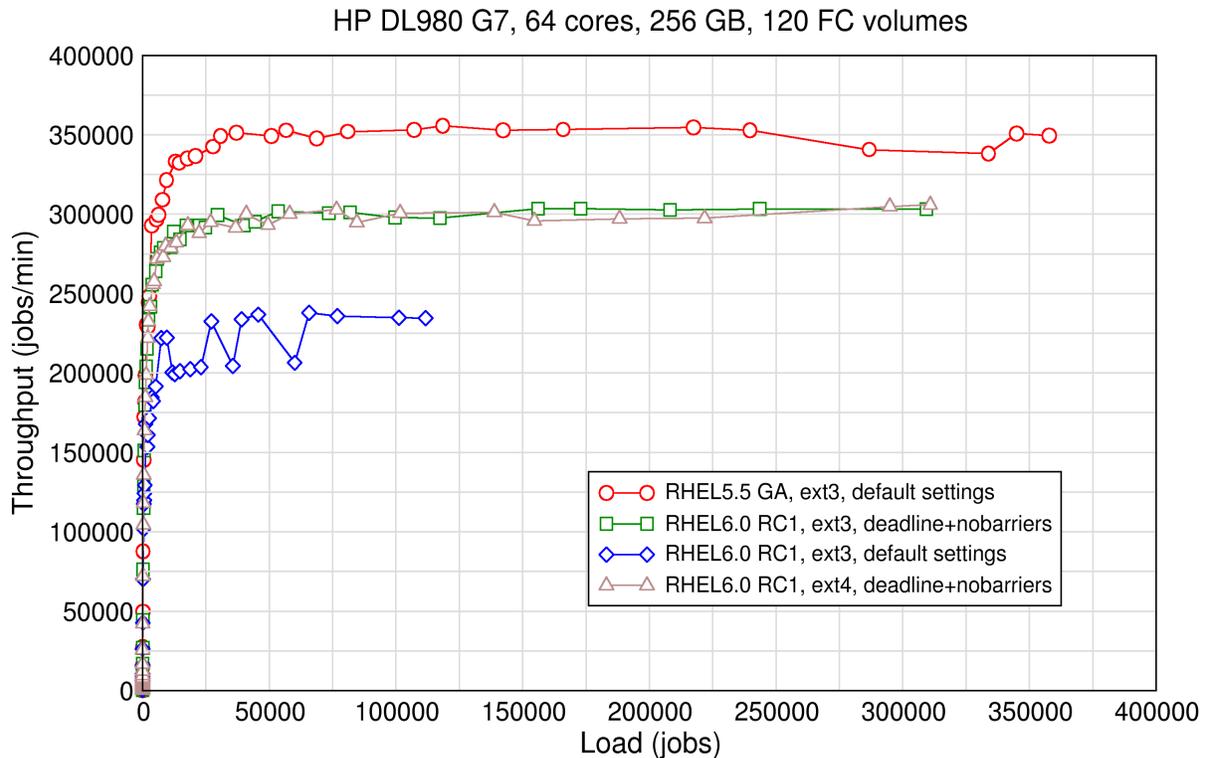


Figure 8: AIM7 Shared Performance Results



5.4.1 Multiuser Shared Workload Workfile

FILESIZE: 1M

POOLSIZE: 10M

| Weight | Tests |
|--------|-------------|
| 30 | add_double |
| 30 | add_float |
| 30 | add_int |
| 30 | add_long |
| 30 | add_short |
| 10 | array_rtns |
| 10 | brk_test |
| 10 | creat-clo |
| 10 | dgram_pipe |
| 10 | dir_rtns_1 |
| 20 | disk_cp |
| 20 | disk_rd |
| 20 | disk_rr |
| 20 | disk_rw |
| 20 | disk_src |
| 20 | disk_wrt |
| 10 | div_double |
| 10 | div_float |
| 10 | div_int |
| 10 | div_long |
| 10 | div_short |
| 20 | exec_test |
| 10 | fork_test |
| 10 | jmp_test |
| 10 | link_test |
| 10 | matrix_rtns |
| 10 | mem_rtns_1 |

| Weight | Tests |
|--------|---------------|
| 10 | mem_rtns_2 |
| 10 | misc_rtns_1 |
| 20 | mul_double |
| 20 | mul_float |
| 20 | mul_int |
| 20 | mul_long |
| 20 | mul_short |
| 10 | new_raph |
| 10 | num_rtns_1 |
| 10 | page_test |
| 10 | pipe_cpy |
| 10 | ram_copy |
| 10 | series_1 |
| 10 | shared_memory |
| 20 | shell_rtns_1 |
| 10 | sieve |
| 10 | signal_test |
| 10 | sort_rtns_1 |
| 10 | stream_pipe |
| 30 | string_rtns |
| 10 | sync_disk_cp |
| 10 | sync_disk_rw |
| 10 | sync_disk_wrt |
| 10 | tcp_test |
| 10 | trig_rtns |
| 10 | udp_test |

Table 5: Multiuser Shared Workfile



6 Summary

HP and Red Hat have been analyzing the performance of AIM7 workloads for more than five years of releases with RHEL 4, RHEL 5, and now RHEL 6. The feedback has led to a number of scalability improvements in RHEL and pushed upstream by both HP and Red Hat engineers.

This reference architecture summarizes how RHEL 6 performed in large HP server environments using default file systems and new enhancements from EXT3 to EXT4 by tuning the performance to alter I/O elevators from the default CFQ to Deadline (optimized for I/O latency) and by disabling unnecessary I/O barriers when enterprise storage is used. In doing so RHEL 6 achieved from 9% improvement in compute performance to as much as 320% improvement in file server peak throughput over the well established leadership results of RHEL 5.5.