



SPEC CPU2006 SPECint_rate_base2006 performance and power consumption on multiprocessor Intel- and AMD-based servers

Executive summary

Intel® Corporation (Intel) commissioned Principled Technologies (PT) to measure the SPEC® CPU2006 performance and power consumption on multiprocessor servers running SUSE Linux Enterprise Server 10 using the following five processors:

- AMD* Opteron* 2384
- Intel® Xeon® processor X5570
- Intel Xeon processor E5450
- Intel Xeon processor 5160
- 64-bit Intel Xeon processor with 2MB L2 cache 3.6 GHz-based server (Intel Xeon processor 3.6 GHz)

SPEC CPU2006 is an industry-standard benchmark created by the Standard Performance Evaluation Corp. (SPEC) to measure a server's compute-intensive performance. The benchmark consequently stresses the CPU and memory subsystems of the system under test. (For more information on SPEC CPU2006 and other SPEC benchmarks, see www.spec.org.)

The SPEC CPU2006 benchmark consists of two benchmark suites, each of which focuses on a different aspect of compute-intensive performance. CINT2006 measures and compares compute-intensive integer performance, while CFP2006 measures and compares

KEY FINDINGS

- The Intel Xeon processor X5570-based server delivered 98.2 percent higher performance than the AMD Opteron 2384-based server. It also delivered 73.8 percent higher performance than the Intel Xeon processor E5450-based server, 240.4 percent higher performance than the Intel Xeon processor 5160-based server, and 1,041.4 percent higher performance than the Intel Xeon processor 3.6 GHz-based server (see Figure 1).
- The Intel Xeon processor X5570-based server delivered 68.0 percent more performance/watt than the AMD Opteron 2384-based server. It also delivered 84.7 percent higher performance/watt than the Intel Xeon processor E5450-based server, 330.1 percent higher performance/watt than the Intel Xeon processor 5160-based server, and 1,044.2 percent higher performance/watt than the Intel Xeon processor 3.6 GHz-based server (see Figure 2).

compute-intensive floating-point performance. A "rate" version of each, which runs multiple instances of the benchmark to assess server performance, is also available. For this report, we ran only the CINT2006 SPECint_rate_base2006 benchmark.

Figure 1 shows the SPECint_rate_base2006 performance of each server. The Intel Xeon processor X5570-based server achieved a score of 226. This is a 98.2 percent performance increase over the AMD Opteron 2384-based server, which achieved a score of 114. The Intel Xeon processor X5570-based server also showed a 73.8 percent performance increase over the Intel Xeon processor E5450-based server, which achieved a score of 130; a 240.4 percent performance increase

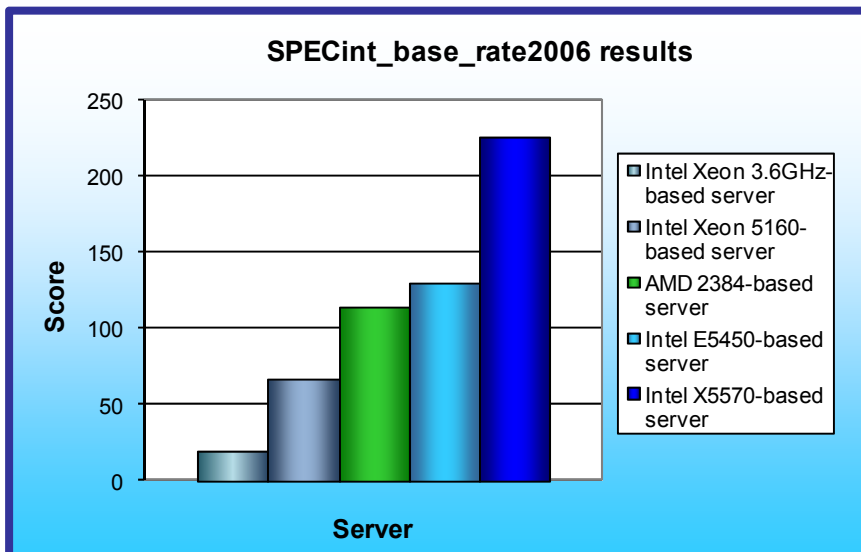


Figure 1: SPECint_rate_base2006 results of the test servers. Higher numbers are better.

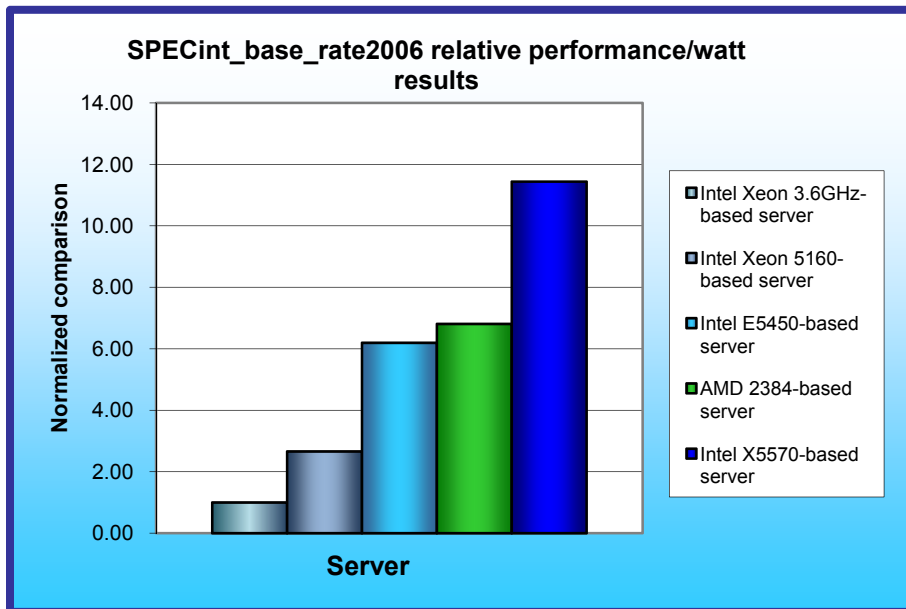


Figure 2: Normalized performance/watt results of the test servers. Higher numbers are better.

average power consumption in watts during period of peak performance.

As Figure 2 illustrates, the Intel Xeon processor X5570-based server delivered 68.0 percent more performance/watt than the AMD Opteron 2384-based server. The Intel Xeon processor X5570-based server also delivered 84.7 percent more performance/watt than the Intel Xeon processor E5450-based server, 330.1 percent more performance/watt than the Intel Xeon processor 5160-based server, and 1,044.2 percent more performance/watt than the Intel Xeon processor 3.6 GHz-based server.

Workload

The SPEC CPU2006 workload includes two benchmark suites: CINT2006 and CFP2006. (Note: SPEC and SPECint are trademarks of the Standard Performance Evaluation Corporation.) We ran only the CINT2006 benchmark, which focuses on measuring and comparing compute-intensive integer performance. Specifically, we measured the SPECint_rate_base2006 results for the test servers with 4, 8, or 16 users.

Name	Application area
400.perlbench	Programming language
401.bzip2	Compression
403.gcc	C compiler
429.mcf	Combinatorial optimization
445.gobmk	Artificial intelligence: Go
456.hmmcr	Search gene sequence
458.sjeng	Artificial intelligence: chess
462.libquantum	Physics/quantum computing
464.h264ref	Video compression
471.omnetpp	Discrete event simulation
473.astar	Path-finding algorithms
483.xalancbmk	XML processing

Figure 3: The applications that make up the CINT2006 benchmark.

Generally, the best SPECint_rate_base2006 score occurs using the same number of users as execution units for a given server. The optimum user count for our testing was 4 users on the Intel Xeon processor 3.6 GHz- and Intel Xeon processor 5160-based servers, 8 users on the Intel Xeon processor E5450- and AMD Opteron 2384-based servers, and 16 users on the Intel Xeon processor X5570-based server. The difference in user counts between the servers is due to the different number or execution units (logical or physical processors) on those servers.

Figure 3 lists the 12 applications that compose the CINT2006 benchmark. SPEC wrote nine of the applications in C and three (471.omnetpp, 473.astar, 483.xalancbmk) in C++.

over the Intel Xeon processor 5160-based server, which achieved a score of 66.4; and a 1,041.4 percent performance increase over the Intel Xeon processor 3.6 GHz-based server, which achieved a score of 19.8.

Figure 2 illustrates the performance/watt for each of the five servers. In this chart, we normalized the results for each system to the system with the lowest performance/watt, which we set to 1.00. By normalizing, we make each data point in these charts a comparative number, with higher results indicating better performance/watt. Thus, higher numbers are better.

To calculate the performance/watt, we used the following formula: benchmark score divided by

A CINT2006 run performs each of the 12 applications three times and reports the median for each. It also calculates the geometric mean of those 12 results to produce an overall score.

Test results

Figure 4 details the results of our tests with the optimum number of users for SPECint_rate_base2006. We determined the number of users based on the number of execution units in a given server. We used the same number of SPECint_rate_base2006 users as processor execution units, so there is a one-to-one ratio.

SPECint_rate_base2006 performs three runs of each benchmark in the test suite and records the median, so the final score is a median of three runs. Higher scores are better.

Figure 4 details the power consumption, in watts, of the test servers while idle and during the benchmark. Idle power is an average of a 2-minute power recording while the server was idle. Average power represents the duration of the benchmark run.

Server	SPECint_rate_base2006 results	Idle power	Average power
AMD Opteron 2384-based server	114.0	170.3	298.4
Intel Xeon processor X5570-based server	226.0	179.0	352.0
Intel Xeon processor E5450-based server	130.0	283.0	373.9
Intel Xeon processor 5160-based server	66.4	299.0	444.9
Intel Xeon processor 3.6 GHz-based server	19.8	194.4	352.9

Figure 4: SPECint_rate_base2006 results and power consumption, in watts, of the test servers while idle and during the benchmark.

Test methodology

Intel configured and provided the four Intel Xeon processor-based servers. PT purchased the AMD Opteron processor-based server.

To maximize performance for each server, we adjusted the BIOS settings as follows:

- Intel Xeon processor X5570-based server. We changed Turbo Mode to Disabled.
- Intel Xeon processor 5160- and Intel Xeon processor E5450-based servers. We changed the Hardware Prefetcher and Adjacent Cache Line Prefetch to Disabled.
- Intel Xeon processor 3.6 GHz-based server. We changed Hyper Threading Technology, Hardware Prefetcher, and Adjacent Cache Line Prefetch to Enabled and set Processor Power Management to GV3.

We used the default BIOS settings on the AMD Opteron 2384-based server.

We began by installing a fresh copy of SUSE Linux* Enterprise Server 10 Service Pack 2. We installed the default packages, the C/C++ Compilers and Tools and disabled the firewall. We made no additional changes to the default installation options.

After the base installation, we updated the kernel on the Intel Xeon processor X5570-based server. We updated from kernel 2.6.16.60-21-smp to 2.6.16.60-0.34-smp. This new kernel provided proper C state support in SUSE for the Intel Xeon processor X5570. This kernel did not add advantages for the other servers so we tested with kernel version 2.6.16.60-21-smp on those four systems.

Figure 5 summarizes some of the key aspects of the configurations of the server systems; Appendix A provides detailed configuration information.

Server	AMD Opteron 2384-based server	Intel Xeon processor X5570-based server	Intel Xeon processor E5450-based server	Intel Xeon processor 5160-based server	64-bit Intel Xeon processor with 2MB L2 cache 3.6 GHz-based server
Processor frequency	2.7 GHz	2.93 GHz	3.0 GHz	3.0 GHz	3.6 GHz
Front-side bus frequency (or transfer rate)	2.0 GT/s HyperTransport	6.4 GT/s (QPI)	1,333 MHz	1,333 MHz	800 MHz
Number of processor packages	2	2	2	2	2
Number of cores per processor package	4	4	4	2	1
Number of hardware threads per core	1	2	1	1	2
Motherboard	Supermicro H8DME-2	Supermicro X8DTN+	Supermicro X7DBE+	Supermicro X7DBE+	Supermicro X6DHT-G
Chipset	NVIDIA MCP55	Intel 5520	Intel 500P	Intel 5000P	Intel E7520
RAM	Kingston KVR800D2D4P 6/4G	Micron MT36JSZF512 72PY-G4D1AB	Kingston KVR667D2D4F 5/2G	Kingston KVR667D2D4F 5/2G	Samsung M312L5720DZ 3-CB3
Hard drive	Western Digital WD1601ABYS	Western Digital WD1600AAJS-00B4A0	Western Digital WD1600AAJS	Western Digital WD1600AAJS	Western Digital WD1600AAJS-11B4A0

Figure 5: Summary of some key aspects of the server configurations.

SPECCPU2006 configuration

For the Intel Xeon processor 3.6 GHz--based server, we followed SPEC's standard instructions for building the CINT2006 executables. After studying the best results for this benchmark on the SPEC Web site, we chose the following software tools:

- Intel C/C++ Compiler 9.1.053 for EM64T MicroQuill SmartHeap v8.1
- MicroQuill SmartHeap v8.1
- Binutils 2.19

For the Intel Xeon processor 5160- and Intel Xeon processor E5450-based servers, we followed SPEC's standard instructions for building the CINT2006 executables. After studying the best results for this benchmark on the SPEC Web site, we chose the following software tools:

- Intel C/C++ Compiler 11.0.074 for EM64T MicroQuill SmartHeap v8.1
- MicroQuill SmartHeap v8.1
- Binutils 2.19

Intel compiled and provided the SPEC CINT2006 executables, but followed SPEC's standard instructions for building the executables using the following software tools for the Intel Xeon processor X5570-based server:

- Intel C/C++ Compiler 11.0.080 for EM64T MicroQuill SmartHeap v8.1
- MicroQuill SmartHeap v8.1
- Binutils 2.18.50.0.7.20080502

For the AMD Opteron 8360 SE-based server, we followed SPEC's standard instructions for building the CINT2006 executables. After studying the best results for this benchmark on the SPEC Web site, we chose the following software tools:

- PGI Server Complete 8.0-3
- Pathscale Compiler Suite 3.2
- MicroQuill SmartHeap v8.1
- Binutils 2.19

The benchmark requires configuration files. Intel provided the configuration files used for the Intel Xeon processor X5570-based server. For the Intel processor 3.6 GHz-, Intel Xeon processor 5160-, Intel Xeon processor E5450-, and AMD Opteron 2384-based servers, we used configuration files from similar tests online. From the SPEC Web site, we chose the most recent (as of the testing for this report) SPEC CPU2006 results that used the above compiler. We copied the configuration files for those results and used them, with modifications to reflect the appropriate system information about the server under test, in our testing. The configuration files we used appear in Appendix B.

We report only the base metrics for the SPECint_rate test. SPEC requires the base metrics for all reported results and sets compilation guidelines that testers must follow in building the executables for such tests.

To begin the benchmark, we performed the following steps:

- Open a command prompt.
- Change to the cpu2006 directory.
- Type `./shrc` at the command prompt.
- Type `runspec -c <config file name> -r <#> -T base -v 10 int` where
 - `<config file name>` = name of the configuration file
 - `<#>` = number of users (we used 4, 8, and 16 users depending on the server)

When the run completes, the benchmark puts the results in the directory `/cpu2006/result`. The result file names are of the form `CINT2006.<number>.<suffix>`. The suffixes are `html`, `asc`, `raw`, and `pdf`. The number is three digits and associates a result file with its log, e.g., `CINT2006.002.asc` and `log.002`.

Appendix C provides the SPECint_rate_base2006 output results for each of the five test servers.

Power measurement procedure

To record each server's power consumption during each test, we used an Extech Instruments* (www.extech.com) 380803 Power Analyzer/Datalogger. We connected the power cord from the server under test to the Power Analyzer's output load power outlet. We then plugged the power cord from the Power Analyzer's input voltage connection into a power outlet.






We used the Power Analyzer's Data Acquisition Software (version 2.11) to capture all recordings. We installed the software on a separate Intel processor-based PC, which we connected to the Power Analyzer via an RS-232 cable. We captured power consumption at 1-second intervals.

To gauge the idle power usage, we recorded the power usage for 2 minutes while each server was running the operating system but otherwise idle.

We then recorded the power usage (in watts) for each server during the testing at 1-second intervals. To compute the average power usage, we averaged the power usage during the time the server was producing its peak performance results. We call this time the power measurement interval. See Figure 4 (idle and average peak power) for the results of these measurements.

Appendix A – Test system configuration information

This appendix provides detailed configuration information about each of the test server systems, which we list in alphabetical order in Figure 16.

Servers	AMD Opteron 2384-based server	Intel Xeon processor X5570-based server	Intel Xeon processor E5450-based server	Intel Xeon processor 5160-based server	64-bit Intel Xeon processor with 2MB L2 cache 3.6 GHz-based server
					
General processor setup					
Number of processor packages	2	2	2	2	2
Number of cores per processor package	4	4	4	2	1
Number of hardware threads per core	1	2	1	1	2
CPU					
Vendor	AMD	Intel	Intel	Intel	Intel
Name	Opteron 2384 (Shanghai)	Xeon X5570 (Nehalem)	Xeon E5450 (Harpertown)	Xeon 5160 (Woodcrest)	Xeon with 2MB L2 cache (Irwindale)
Stepping	2	5	C0	4	R0
Socket type	Socket F	LGA 1366	LGA 771	LGA 771	Socket 604
Core frequency	2.7 GHz	2.93 GHz	3.0 GHz	3.0 GHz	3.6 GHz
Front-side bus frequency (or transfer rate)	2.0 GT/s HyperTransport	6.4 GT/s (QPI)	1,333 MHz	1,333 MHz	800 MHz
L1 cache	64 KB + 64 KB (per core)	32 KB + 32 KB (per core)	32 KB + 32 KB (per core)	32 KB + 32 KB (per core)	12 KB + 16 KB
L2 cache	2 MB (4 x 512 MB)	1 MB (4 x 256 KB)	12 MB (2 x 6,144 MB)	4 MB (2 x 2,048 MB)	2 MB (1 x 2,048 MB)
L3 cache	1 x 6 MB	1 x 8 MB	N/A	N/A	N/A
Platform					
Vendor and model number	Supermicro A+ Server 2021M-T2R+V	Supermicro SuperServer	Supermicro SuperServer 6025B-TR+	Supermicro SuperServer 6025B-TR+	Supermicro 6024H-T
Motherboard model number	H8DME-2	X8DTN+	X7DBE+	X7DBE+	X6DHT-G
Motherboard chipset	NVIDIA MCP55	Intel 5520	Intel 5000P	Intel 5000P	Intel E7520
Motherboard revision number	01	1.02	B1	B1	01

Servers	AMD Opteron 2384-based server	Intel Xeon processor X5570-based server	Intel Xeon processor E5450-based server	Intel Xeon processor 5160-based server	64-bit Intel Xeon processor with 2MB L2 cache 3.6 GHz-based server
BIOS name and version	AMIBIOS 08.00.14 (10/13/08)	American Megatrends v4.6.3.2 (01/13/2009)	PhoenixBios v2.1 (06/23/08)	PhoenixBios v2.1 (06/23/08)	Phoenix Technologies LTD 6.00 (08/16/2007)
BIOS settings	Default	Turbo Mode Disabled	Hardware Prefetcher and Adjacent Cache Line Prefetch Disabled	Hardware Prefetcher and Adjacent Cache Line Prefetch Disabled	Hyper Threading Technology, Hardware Prefetcher and Adjacent Cache Line Prefetch Enabled Processor Power Management: GV3
Memory module					
Vendor and model number	Kingston KVR800D2D4P6 /4G	Micron MT36JSZF51272 PY-1G4D1AB	Kingston KVR667D2D4F5/ 2G	Kingston KVR667D2D4F5/ 2G	Samsung M312L5720DZ3-CB3
Type	DDR2 PC2-6400	DDR3 PC3-10600	DDR2 PC2-5300 FB-DIMM	DDR2 PC2-5300 FB-DIMM	DDR PC2700R
Speed (MHz)	800	1,333	667	667	333
Speed in the system currently running @ (MHz)	800	1,333	667	667	333
Timing/Laten cy (tCL-tRCD-iRP-tRASmin)	6-6-6-18	9-9-9-24	5-5-5-15	5-5-5-15	2.5-3-3-10
Size	4 GB	4 GB	2 GB	2 GB	2 GB
Number of RAM modules	8 x 4 GB	6 x 4 GB	8 x 2 GB	8 x 2 GB	8 x 2 GB
Chip organization	Double-sided	Double-sided	Double-sided	Double-sided	Double-sided
Hard disk					
Vendor and model number	Western Digital WD1601ABYS	Western Digital WD1600AAJS-00B4A0	Western Digital WD1600AAJS	Western Digital WD1600AAJS	Western Digital WD1600AAJS-11B4A0
Number of disks in system	1	1	1	1	1
Size	160 GB	160 GB	160 GB	160 GB	160 GB
Buffer size	16 MB	8 MB	8 MB	8 MB	16 MB
RPM	7,200	7,200	7,200	7,200	7,200 RPM
Type	SATA	SATA	SATA	SATA	SATA

Servers	AMD Opteron 2384-based server	Intel Xeon processor X5570-based server	Intel Xeon processor E5450-based server	Intel Xeon processor 5160-based server	64-bit Intel Xeon processor with 2MB L2 cache 3.6 GHz-based server
Controller	NVIDIA nForce SATA controller	Intel Corporation ICH10 SATA AHCI Controller	Intel ESB2 RAID Controller	Intel ESB2 RAID Controller	Adaptec Serial ATA HostRAID
Driver version	3.3	2.0	2.00ac7	2.00ac7	2.0
Operating system					
Name	SLES 10 SP2	SLES 10 SP2	SLES 10 SP2	SLES 10 SP2	SLES 10 SP2
Build number	#1 SMP Tue May 6	#1 SMP Tue May 6	#1 SMP Tue May 6	#1 SMP Tue May 6	#1 SMP Tue May 6
File system	Reiserfs	Reiserfs	Reiserfs	Reiserfs	Reiserfs
Kernel	2.6.16.60-0.21-smp	2.6.16.60-0.34-smp	2.6.16.60-0.21-smp	2.6.16.60-0.21-smp	2.6.16.60-0.21-smp
Language	English	English	English	English	English
Graphics					
Vendor and model number	ATI ES1000	ATI ES1000	ATI ES1000	ATI ES1000	ATI Rage XL SVGA PCI Video
Chipset	ES1000	ES1000	ES1000	ES1000	ATI Rage XL
Type	Integrated	Integrated	Integrated	Integrated	Integrated
Memory size	16 MB	32 MB	16 MB	16 MB	8 MB
Resolution	1,024 x 768	1,024 x 768	1,024 x 768	1,024 x 768	1,024 x 768
Network card/subsystem					
Vendor and model number	Intel Pro/1000	Intel 82576EB Gigabit Ethernet	Intel Pro/1000 EB	Intel Pro/1000 EB	Intel PRO/1000 MT
Type	Integrated	Integrated	Integrated	Integrated	Integrated
Optical drive					
Vendor and model number	Matshita DVD-ROM UJDA780	Slimtype DVD A DS8A1P	Matshita DVD-ROM SR-8178	Matshita DVD-ROM SR-8178	TEAC CD-224E-N
USB ports					
Number	4	4	4	4	2
Type	USB 2.0	USB 2.0	USB 2.0	USB 2.0	2.0
Power supplies					
Total number	2	2	2	2	1
Wattage of each	700W	805W	700W	700W	550W
Cooling fans					
Total number	3	3	3	3	4
Dimensions	3" x 3"	3" x 3"	3" x 3"	3" x 3"	3.25" x 3.25"
Voltage	12V	12V	12V	12V	12V
Amps	1.1A	1.1A	1.1A	1.1A	1.1A

Figure 6: Detailed system configuration information for the five test servers.

Appendix B – SPECint_rate_base2006 configuration files

This appendix contains the benchmark configuration files we used to test the servers.

SUSE Linux Enterprise Server 10: AMD Opteron 2384-based server

```
#####
# AMD SPEC CPU2006 V1.1 Configuration File for 64-bit Linux (Quad-Core)
#
#   CPU2006 Version           1.1
#   Compiler name/version:    PGI 7.2, Pathscale 3.2
#   Operating system version: 64-bit Linux
#   Hardware:                 AMD Opteron (Quad-core)
#   FP Base Pointer Size:     64-bit only
#   FP Peak Pointer Size:     32/64-bit
#   INT Base Pointer Size:    32/64-bit
#   INT Peak Pointer Size:    32/64-bit
#   Auto Parallization:       Yes
#
# Important! Please run with your stack size set to 'unlimited'.
# Failure to do so may cause 483.xalanbmk to get a stack overflow during
# execution.
# Using csh:  unlimit
# Using bash: ulimit -s unlimited
#
# Your LD_LIBRARY_PATH must be set to the location of the dependent
# runtime libraries. (Though this should be taken care of by the
# preENV_LD_LIBRARY_PATH found in the AMD909gh-sut.inc file.)
#
# The NCPUS environment variable must be set for the run. This
# is passed to runspec by using --define NPCUS=4 on the runspec
# command and is used to set an environment variable in the
# AMD909gh-sut.inc file.
#
# If building new binaries, please adjust the SMARTHEAP_DIR variable to
# the directory containing the Smartheap library.
#
#####
# Macro section
#####
# Modify this section to use the appropriate architecture flags
#define pgi_tp64 -tp barcelona-64
#define pgi_tp32 -tp barcelona-32

# Adjust the build jobs to the number of concurrent build processes
#define build_jobs 4

#####
# Header Section
#####
ext           = amd909gh
ignore_errors = no
tune         = base,peak
output_format = asc,pdf,raw,flags,cfg,html,csv

flagsurl000  = http://www.spec.org/cpu2006/flags/pgi80_linux_flags.xml
flagsurl001  = http://www.spec.org/cpu2006/flags/CPU2006_flags.20081112.xml
flagsurl002  = http://www.spec.org/cpu2006/flags/amd-platform-amd909gh.xml

size         = test,train,ref
check_md5   = yes
reportable  = yes
env_vars    = no
teeout      = yes
mean_anyway = yes
verbose     = 6

makeflags    = -j ${build_jobs}
```

```

#####
# Include file containing the SUT hardware information
# as well as the submit command, tester information and notes
#####
#include: AMD909gh-sut.inc
# ---- Begin inclusion of 'AMD909gh-sut.inc'
#####
#####
# Pre-Environment settings
#####

# These settings require using 4GB DIMMs - 4 per socket

# For rate runs, use the following below:
# In /etc/sysctl.conf, 1P configuration, use vm/nr_hugepages=3584
# In /etc/sysctl.conf, 2P configuration, use vm/nr_hugepages=7168
# In /etc/sysctl.conf, 4P configuration, use vm/nr_hugepages=14336

# Document amount of run time system hugepages setting
# What is vm/nr_hugepages is set to in /etc/sysctl.conf?
%define rt_sys_hp 7168

# Set number of sockets being used
%define num_sockets 2

# Set NCPUS environment variable for PGI autopar binaries
%ifdef %{NCPUS}
    preENV_NCPUS=%{NCPUS}
%endif

# Set location of run time libraries
preENV_LD_LIBRARY_PATH = /usr/cpu2006/SmartHeap_8.1/lib

# Need this for libhugetlbfs library usage for Pathscale binaries
preENV_HUGETLB_MORECORE=yes

# Run the sync command after setup
post_setup=sync

#####
# Submit Section
#####
# This config file is set to run these binaries on a 4P, 16 core system.
# If this does not match the description of your system, change this
# config file, specifically match the socket the memroy is bound (-m) to
# the physical cpu number. Use "man numactl" and "numactl --hardware" to
# better understand how to use this command.

submit0    = echo "$command" > run.sh
submit1    = if `test $$SPECCOPYNUM -eq 0 `; then numactl -m 0 --physcpubind=0 sh run.sh; fi
submit2    = if `test $$SPECCOPYNUM -eq 1 `; then numactl -m 0 --physcpubind=1 sh run.sh; fi
submit3    = if `test $$SPECCOPYNUM -eq 2 `; then numactl -m 0 --physcpubind=2 sh run.sh; fi
submit4    = if `test $$SPECCOPYNUM -eq 3 `; then numactl -m 0 --physcpubind=3 sh run.sh; fi
submit5    = if `test $$SPECCOPYNUM -eq 4 `; then numactl -m 1 --physcpubind=4 sh run.sh; fi
submit6    = if `test $$SPECCOPYNUM -eq 5 `; then numactl -m 1 --physcpubind=5 sh run.sh; fi
submit7    = if `test $$SPECCOPYNUM -eq 6 `; then numactl -m 1 --physcpubind=6 sh run.sh; fi
submit8    = if `test $$SPECCOPYNUM -eq 7 `; then numactl -m 1 --physcpubind=7 sh run.sh; fi
submit9    = if `test $$SPECCOPYNUM -eq 8 `; then numactl -m 2 --physcpubind=8 sh run.sh; fi
submit10   = if `test $$SPECCOPYNUM -eq 9 `; then numactl -m 2 --physcpubind=9 sh run.sh; fi
submit11   = if `test $$SPECCOPYNUM -eq 10 `; then numactl -m 2 --physcpubind=10 sh run.sh; fi
submit12   = if `test $$SPECCOPYNUM -eq 11 `; then numactl -m 2 --physcpubind=11 sh run.sh; fi
submit13   = if `test $$SPECCOPYNUM -eq 12 `; then numactl -m 3 --physcpubind=12 sh run.sh; fi
submit14   = if `test $$SPECCOPYNUM -eq 13 `; then numactl -m 3 --physcpubind=13 sh run.sh; fi
submit15   = if `test $$SPECCOPYNUM -eq 14 `; then numactl -m 3 --physcpubind=14 sh run.sh; fi
submit16   = if `test $$SPECCOPYNUM -eq 15 `; then numactl -m 3 --physcpubind=15 sh run.sh; fi

# SPEED
# For a speed run, the submit1 command above will be used.

```

```

#
use_submit_for_speed = 1

#####
# Tester information
#####
license_num      = 3184
prepared_by      = Principled Technologies, Inc.
tester           = Principled Technologies, Inc.
test_sponsor     = Intel Corp.
hw_vendor        = Supermicro Computer, Inc.
hw_model         = AMD Opteron 2384

#####
# Hardware information
#####
default=default=default=default:
hw_avail         = Nov-2008
hw_cpu_name      = AMD Opteron 2384
hw_cpu_mhz       = 2700
hw_fpu           = Integrated
hw_nchips        = 2
hw_ncores        = 8
hw_ncoresperchip = 4
hw_nthreadspercore = 1
hw_ncpuorder     = 1,2 chips
hw_pcache        = 64 KB I + 64 KB D on chip per core
hw_scache        = 512 KB I+D on chip per core
hw_tcache        = 6 MB I+D on chip per chip
hw_ocache        = None
hw_disk          = 1 x 160 GB SATA, 7200 RPM
hw_memory        = 32 GB (8x4 GB, DDR2 PC2-6400)
hw_other         = None
sw_file          = ReiserFS
sw_os000         = SuSE Linux Enterprise Server 10 (x86_64) SP2,
sw_os001         = Kernel 2.6.16-60.0.21-smp
sw_state         = Run level 3 (Full multiuser with network)

#####
# Notes
#####
notes_submit_000 = 'numactl' was used to bind copies to the cores

notes_os_000 = The libhugetlbfs libraries were installed using the
notes_os_005 = installation rpms that came with the distribution.
notes_os_010 =
notes_os_015 = 'ulimit -s unlimited' was used to set environment stack size
notes_os_020 = 'ulimit -l 2097152' was used to set environment locked pages in memory limit
notes_os_025 =
notes_os_030 = Set vm/nr_hugepages=7168 in /etc/sysctl.conf
notes_os_035 = mount -t hugetlbfs nodev /mnt/hugepages
notes_os_040 =

# Uncomment this if you used this for a speed run
#notes_os_045 = powersave -f was used to set the CPU frequency to its maximum.
# ---- End inclusion of '/root/work/cpu2006-amd909gh/config/AMD909gh-sut.inc'

##### Software Info #####
fp=default=default=default:
sw_base_ptrsize  = 64-bit
sw_peak_ptrsize  = 32/64-bit
sw_avail         = Feb-2009
sw_compiler000   = PGI Server Complete Version 8.0-3
sw_compiler001   = PathScale Compiler Suite Version 3.2
sw_other000      = binutils 2.19
sw_other001      = 32-bit and 64-bit libhugetlbfs libraries
sw_parallel_other = Yes

int=default=default=default:

```

```

sw_base_ptrsize = 32/64-bit
sw_peak_ptrsize = 32/64-bit
sw_avail        = Feb-2009
sw_compiler000  = PGI Server Complete Version 8.0-3
sw_compiler001  = PathScale Compiler Suite Version 3.2
sw_other000     = binutils 2.19
sw_other001     = 32-bit and 64-bit libhugetlbfs libraries
sw_other002     = SmartHeap 8.1 32-bit Library for Linux
sw_parallel_other = No

default=default=default=default:
#####
# Compiler selection
#####
CC          = pgcc
CXX         = pgcpp
FC          = pgf95

SMARTHEAP_DIR = /usr/cpu2006/SmartHeap_8.1/lib
HUGEPAGE64   = -Wl,-T/usr/share/libhugetlbfs/ldscripts/elf_x86_64.xBDT -L/usr/lib64 -lhugetlbfs
HUGEPAGE32   = -L/usr/lib -lhugetlbfs

#####
# Portability
#####
# Set for all base and peak benchmarks unless changed below
fp=default=default=default:
PORTABILITY = -DSPEC_CPU_LP64

int=default=default=default:
PORTABILITY = -DSPEC_CPU_LP64

#####
# INT Base Flags
#####
int=base=default=default:
COPTIMIZE   = -fastsse -Msmartalloc=huge -Mfprelaxed -Mipa=jobs:{build_jobs},fast,inline %{pgi_tp64}
CXXOPTIMIZE = -fastsse -Msmartalloc=huge -Mfprelaxed --zc_eh -Mipa=jobs:{build_jobs},fast,inline:6
%{pgi_tp32}
OPTIMIZE    = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi

#####
# FP Base Flags
#####
fp=base=default=default:
COPTIMIZE   = -fastsse -Msmartalloc=huge -Mfprelaxed -Mipa=jobs:{build_jobs},fast,inline %{pgi_tp64}
FOPTIMIZE   = -fastsse -Mfprelaxed -Msmartalloc=huge -Mipa=jobs:{build_jobs},fast,inline %{pgi_tp64}
CXXOPTIMIZE = -fastsse -Msmartalloc=huge -Mfprelaxed --zc_eh -Mipa=jobs:{build_jobs},fast,inline
%{pgi_tp64}
OPTIMIZE    = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi

#####
# INT Portability Flags
#####
400.perlbench=default=default=default:
CPORTABILITY = -DSPEC_CPU_LINUX_X64

403.gcc=peak=default=default:
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY =

429.mcf=peak=default=default:
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY =

462.libquantum=default=default=default:
CPORTABILITY = -DSPEC_CPU_LINUX

```

```

471.omnetpp=base=default=default:
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY      =

473.astar=default=default=default:
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY      =

483.xalancbmk=default=default=default:
CXXPORTABILITY   = -DSPEC_CPU_LINUX
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY      =

#####
# INT Peak Flags
#####
# PATHSCALE
400.perlbench=peak=default=default:
CC                = pathcc -march=barcelona
COPTIMIZE         = -Ofast -IPA:plimit=20000:field_reorder=on -LNO:opt=0 -WOPT:if_conv=0 -
CG:local_sched_alg=1
PASS1_CFLAGS      = -fb_create fbdata
PASS1_LDFLAGS     = -fb_create fbdata
PASS2_CFLAGS      = -fb_opt fbdata
PASS2_LDFLAGS     = -fb_opt fbdata $(HUGEPAGE64)
feedback          = 1
basepeak          = 0

# PATHSCALE
401.bzip2=peak=default=default:
CC                = pathcc -march=barcelona
COPTIMIZE=-O3 -OPT:alias=disjoint -OPT:Ofast:goto=off -INLINE:aggressive=on -CG:local_sched_alg=1 -
m3dnw $(HUGEPAGE64)
feedback          = 0
basepeak          = 0

# PATHSCALE
403.gcc=peak=default=default:
CC                = pathcc -march=barcelona
COPTIMIZE         = -Ofast -OPT:malloc_alg=1 -LNO:trip_count=256:prefetch Ahead=10 -CG:prefer_lru_reg=off -
m32
PASS1_CFLAGS      = -fb_create fbdata
PASS1_LDFLAGS     = -fb_create fbdata
PASS2_CFLAGS      = -fb_opt fbdata
PASS2_LDFLAGS     = -fb_opt fbdata
feedback          = 1
basepeak          = 0

# PATHSCALE
429.mcf=peak=default=default:
CC                = pathcc -march=barcelona
COPTIMIZE         = -O3 -ipa -INLINE:aggressive=on -CG:gcm=off -GRA:prioritize_by_density=on -m32
$(HUGEPAGE32)
feedback          = 0
basepeak          = 0

# PATHSCALE
445.gobmk=peak=default=default:
CC                = pathcc -march=barcelona
COPTIMIZE         = -O3 -OPT:alias=restrict -LNO:prefetch=1:ignore_feedback=off -CG:p2align=on
PASS1_CFLAGS      = -fb_create fbdata
PASS1_LDFLAGS     = -fb_create fbdata
PASS2_CFLAGS      = -fb_opt fbdata
PASS2_LDFLAGS     = -fb_opt fbdata $(HUGEPAGE64)
feedback          = 1
basepeak          = 0

# PGI
456.hmmmer=peak=default=default:
CC                = pgcc

```

```

COPTIMIZE      = -fastsse -Mvect=partial -Munroll=n:8 -Msmartalloc=huge -Msafeptr -Mprefetch=t0 -
Mfprelaxed    -Mipa=jobs:%{build_jobs},const,ptr,arg,inline  %{pgi_tp64}
OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
feedback      = 0
basepeak      = 0

# PATHSCALE
458.sjeng=peak=default=default:
CC            = pathcc -march=barcelona
%ifdef %{gnu3_fe}
  COPTIMIZE   = -O3 -ipa -LNO:ignore_feedback=off:full_unroll=10:fusion=0:fission=2 -
IPA:plimit=50000:pu_reorder=2 -CG:ptr_load_use=0 -OPT:unroll_times_max=8 -INLINE:aggressive=on
%else
  COPTIMIZE   = -O3 -ipa -LNO:ignore_feedback=off:full_unroll=10:fusion=0:fission=2 -IPA:pu_reorder=2 -
CG:ptr_load_use=0 -OPT:unroll_times_max=8 -INLINE:aggressive=on
%endif
PASS1_CFLAGS  = -fb_create fbdata
PASS1_LDFLAGS = -fb_create fbdata
PASS2_CFLAGS  = -fb_opt fbdata
PASS2_LDFLAGS = -fb_opt fbdata $(HUGEPAGE64)
feedback      = 1
basepeak      = 0

# PGI
462.libquantum=peak=default=default:
CC            = pgcc
COPTIMIZE     = -fastsse -Munroll=m:8 -Msmartalloc=huge -Mprefetch=distance:4 -Mfprelaxed -
Mipa=jobs:%{build_jobs},fast,inline,noarg %{pgi_tp64}
OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
feedback      = 0
basepeak      = 0

# PATHSCALE
464.h264ref=peak=default=default:
CC            = pathcc -march=barcelona
COPTIMIZE     = -O3 -IPA:plimit=20000 -OPT:alias=disjoint -LNO:prefetch=0 -
CG:ptr_load_use=0:push_pop_int_saved_regs=off:prefer_lru_reg=off
PASS1_CFLAGS  = -fb_create fbdata
PASS1_LDFLAGS = -fb_create fbdata
PASS2_CFLAGS  = -fb_opt fbdata
PASS2_LDFLAGS = -fb_opt fbdata $(HUGEPAGE64)
feedback      = 1
basepeak      = 0

# PGI
471.omnetpp=peak=default=default:
feedback      = 0
basepeak      = 1

# PGI
473.astar=peak=default=default:
CXX           = pgcpp
CXXOPTIMIZE   = -fastsse -O4 -Msmartalloc=huge -Msafeptr=global -Mfprelaxed --zc_eh  %{pgi_tp32}
OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
PASS1_CXXFLAGS = -Mpfi
PASS1_LDFLAGS = -Mpfi
PASS2_CXXFLAGS = -Mpfo -Mipa=jobs:%{build_jobs},fast,inline:6
PASS2_LDFLAGS = -Mpfo -Mipa=jobs:%{build_jobs},fast,inline:6
feedback      = 1
basepeak      = 0

# PATHSCALE
483.xalancbmk=peak=default=default:
CXX           = pathCC -march=barcelona
CXXOPTIMIZE   = -Ofast -INLINE:aggressive=on -m32
EXTRA_CXXLIBS = -L$(SMARTHEAP_DIR) -lsmartheap
feedback      = 0

```

```

basepeak      = 0

#####
# FP Portability
#####
435.gromacs=base=default=default:
LDPORTABILITY = -Mnomain

436.cactusADM=default=default=default:
LDPORTABILITY = -Mnomain

447.dealII=default=default:
#ifdef % {gnu3_fe}
    CXXPORTABILITY = -DSPEC_CPU_TABLE_WORKAROUND
#else
    CXXPORTABILITY =
#endif

447.dealII=peak=default:
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY =

454.calculix=default=default=default:
LDPORTABILITY = -Mnomain

450.soplex=peak=default=default:
# Needed to avoid -DSPEC_CPU_LP64 on 32-bit codes
PORTABILITY =

481.wrf=base=default=default:
CPORTABILITY = -DSPEC_CPU_CASE_FLAG -DSPEC_CPU_LINUX

481.wrf=peak=default=default:
FPORABILITY = -fno-second-underscore
CPORTABILITY = -DSPEC_CPU_LINUX

#####
# FP Peak Flags
#####
# PGI
410.bwaves=peak=default=default:
FC = pgf95
FOPTIMIZE = -fastsse -Msmartalloc -Mprefetch=nta -Mfprelaxed -Mipa=jobs: {build_jobs},fast,inline
% {pgi_tp64}
OPTIMIZE = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
feedback = 0
basepeak = 0

# PATHSCALE
416.gamess=peak=default=default:
FC = pathf95 -march=barcelona
FOPTIMIZE = -O2 -OPT:Ofast:ro=3:unroll_size=256
PASS1_FFLAGS = -fb_create fbdata
PASS1_LDFLAGS = -fb_create fbdata
PASS2_FFLAGS = -fb_opt fbdata
PASS2_LDFLAGS = -fb_opt fbdata $ (HUGEPAGE64)
feedback = 1
basepeak = 0

# PGI
433.milc=peak=default=default:
feedback = 0
basepeak = 1

# PGI
434.zeusmp=peak=default=default:
FC = pgf95
FOPTIMIZE = -fastsse -Mfprelaxed -Mprefetch=distance:8,t0 -Msmartalloc=huge,hugebss -
Mipa=jobs: {build_jobs},fast,inline % {pgi_tp64}

```

```

OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
feedback      = 0
basepeak      = 0

# PATHSCALE
435.gromacs=peak=default=default:
CC             = pathcc -march=barcelona
FC             = pathf95 -march=barcelona
COPTIMIZE     = -Ofast -OPT:rsqrt=2 -OPT:malloc_alg=1 $(HUGEPAGE64)
FOPTIMIZE     = -Ofast -OPT:rsqrt=2 -OPT:malloc_alg=1 $(HUGEPAGE64)
feedback      = 0
basepeak      = 0

# PGI
436.cactusADM=peak=default=default:
CC             = pgcc
FC             = pgf95
COPTIMIZE     = -fastsse -Mconcur -Msmartalloc=huge -Mfprelaxed -Mipa=jobs:${build_jobs},fast,inline
%{pgi_tp64}
FOPTIMIZE     = -fastsse -Mconcur -Mfprelaxed -Msmartalloc=huge -Mipa=jobs:${build_jobs},fast,inline
%{pgi_tp64}
OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
submit0       = echo "$command" > run.sh
submit1       = if `test $SPECCOPYNUM -eq 0 `; then numactl -m 0 --physcpubind=0,1,2,3 sh run.sh; fi
submit2       = if `test $SPECCOPYNUM -eq 1 `; then numactl -m 1 --physcpubind=4,5,6,7 sh run.sh; fi
submit3       = if `test $SPECCOPYNUM -eq 2 `; then numactl -m 2 --physcpubind=8,9,10,11 sh run.sh; fi
submit4       = if `test $SPECCOPYNUM -eq 3 `; then numactl -m 3 --physcpubind=12,13,14,15 sh run.sh; fi
copies        = ${num_sockets}
feedback      = 0
basepeak      = 0

# PGI
437.leslie3d=peak=default=default:
FC             = pgf95
FOPTIMIZE     = -fastsse -Mvect=fuse -Msmartalloc=huge -Mprefetch=distance:8,t0 -Mfprelaxed %{pgi_tp64}
OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
PASS1_FFLAGS  = -Mpfi=indirect
PASS1_LDFLAGS = -Mpfi=indirect
PASS2_FFLAGS  = -Mpfo=indirect -Mipa=jobs:${build_jobs},fast,inline
PASS2_LDFLAGS = -Mpfo=indirect -Mipa=jobs:${build_jobs},fast,inline
feedback      = 1
basepeak      = 0

# PGI
444.namd=peak=default=default:
CXX           = pgc++
CXXOPTIMIZE   = -fastsse -Munroll=n:4,m:8 -Msmartalloc=huge -Mnodepchk -Mfprelaxed --zc_eh %{pgi_tp64}
OPTIMIZE      = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
PASS1_CXXFLAGS = -Mpfi
PASS1_LDFLAGS = -Mpfi
PASS2_CXXFLAGS = -Mpfo -Mipa=jobs:${build_jobs},fast,inline
PASS2_LDFLAGS = -Mpfo -Mipa=jobs:${build_jobs},fast,inline
feedback      = 1
basepeak      = 0

# PATHSCALE
447.dealIII=peak=default=default:
CXX           = pathCC -march=barcelona
%ifdef %{gnu3_fe}
  CXXOPTIMIZE = -Ofast -INLINE:aggressive=on -LNO:opt=0 -OPT:alias=disjoint -fno-exceptions -m32
%else
  CXXOPTIMIZE = -Ofast -static -INLINE:aggressive=on -fno-exceptions -m32
  LDCXXFLAGS=-lm
%endif
feedback      = 0
basepeak      = 0

```



```

# PATHSCALE
450.soplex=peak=default=default:
CXX          = pathCC -march=barcelona
CXXOPTIMIZE  = -O3 -INLINE:aggressive=on -
OPT:IEEE_arith=3:IEEE_NaN_Inf=off:fold_unsigned_relops=on:malloc_alg=1 -CG:load_exe=0 -fno-exceptions -
m32
PASS1_CXXFLAGS = -fb_create fbdata
PASS1_LDFLAGS  = -fb_create fbdata
PASS2_CXXFLAGS = -fb_opt fbdata
PASS2_LDFLAGS  = -fb_opt fbdata $(HUGEPAGE32)
feedback      = 1
basepeak      = 0

# PATHSCALE
453.povray=peak=default=default:
CXX          = pathCC -march=barcelona
CXXOPTIMIZE  = -Ofast -INLINE:aggressive=on
PASS1_CXXFLAGS = -fb_create fbdata
PASS1_LDFLAGS  = -fb_create fbdata
PASS2_CXXFLAGS = -fb_opt fbdata
PASS2_LDFLAGS  = -fb_opt fbdata
feedback      = 1
basepeak      = 0

# PGI
454.calculix=peak=default=default:
CC          = pgcc
FC          = pgf95
COPTIMIZE   = -fastsse -Msmartalloc=huge -Mprefetch=t0 -Mpre -Mfprelaxed %{pgi_tp64}
FOPTIMIZE   = -fastsse -Msmartalloc=huge -Mprefetch=t0 -Mpre -Mfprelaxed %{pgi_tp64}
OPTIMIZE    = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
PASS1_CFLAGS = -Mpfi=indirect
PASS1_FFLAGS = -Mpfi=indirect
PASS1_LDFLAGS = -Mpfi=indirect
PASS2_CFLAGS = -Mpfo=indirect -Mipa=jobs:%{build_jobs},fast,inline
PASS2_FFLAGS = -Mpfo=indirect -Mipa=jobs:%{build_jobs},fast,inline
PASS2_LDFLAGS = -Mpfo=indirect -Mipa=jobs:%{build_jobs},fast,inline
feedback     = 1
basepeak     = 0

# PATHSCALE
459.GemsFDTD=peak=default=default:
FC          = pathf95 -march=barcelona
FOPTIMIZE   = -Ofast -LNO:fission=2:simd=2:prefetch Ahead=1 -CG:load_exe=0 -CG:prefer_lru_reg=off -
OPT:malloc_alg=1 $(HUGEPAGE64)
feedback     = 0
basepeak     = 0

# PATHSCALE
465.tonto=peak=default=default:
FC          = pathf95 -march=barcelona
FOPTIMIZE   = -Ofast -OPT:alias=no_f90_pointer_alias -LNO:blocking=off -CG:load_exe=1 -IPA:plimit=525
-OPT:malloc_alg=1 $(HUGEPAGE64)
feedback     = 0
basepeak     = 0

# PGI
470.lbm=peak=default=default:
CC          = pgcc
COPTIMIZE   = -fastsse -Msmartalloc=huge -Mprefetch=t0 -Mloop32 -Mfprelaxed -
Mipa=jobs:%{build_jobs},fast,inline %{pgi_tp64}
OPTIMIZE    = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
feedback     = 0
basepeak     = 0

# PATHSCALE
481.wrf=peak=default=default:

```

```

CC          = pathcc -march=barcelona
FC          = pathf95 -march=barcelona
COPTIMIZE   = -Ofast -LNO:blocking=off:prefetch Ahead=10 -LANG:copyinout=off -IPA:callee_limit=5000 -
GRA:prioritize_by_density=on -OPT:malloc_alg=1 -m3dnow $(HUGE_PAGE64)
FOPTIMIZE   = $(COPTIMIZE)
feedback    = 0
basepeak    = 0

```

```

# PGI
482.sphinx3=peak=default=default:
CC          = pgcc
COPTIMIZE   = -fastsse -Mfprelaxed -Msmartalloc %{pgi_tp64}
OPTIMIZE    = -Mvect=cachesize:6291456
EXTRA_LDFLAGS = -Bstatic_pgi
PASS1_CFLAGS = -Mpfi=indirect
PASS1_LDFLAGS = -Mpfi=indirect
PASS2_CFLAGS = -Mpfo=indirect -Mipa=jobs:{build_jobs},fast,inline
PASS2_LDFLAGS = -Mpfo=indirect -Mipa=jobs:{build_jobs},fast,inline
feedback    = 1
basepeak    = 0

```

SUSE Linux Enterprise Server 10: Intel Xeon processor X5570-based server

```

#####
# This is a sample config file. It was tested with:
#
#   Compiler name/version:      Intel Compiler 11.0
#   Operating system version:   64-Bit SUSE LINUX Enterprise Server 10 or later
#   Hardware:                   Intel processors supporting SSE4.2
#
#####
# SPEC CPU2006 Intel Linux64 config file
# July 2008 Intel Compiler 11.0 for Linux64
#####
action      = validate
tune        = base
ext         = cpu2006.1.1.ic11.0.linux64.sse42.rate.feb2009
PATHSEP     = /
check_md5=1
reportable=1

#
# These are listed as benchmark-tuning-extension-machine
#
default=default=default=default:
CC = icc
CXX = icpc
FC=ifort
OBJ = .o
SMARTHEAP_DIR = /spec/cpu2006.1.1/lib

# For UP systems, we need to know if the processors are ordered across cores first or in order
# If across cores, processors 0, 1, 2 and 3 are on distinct physical cores
# Otherwise, processors 0, 2, 4 and 6 are on distinct physical cores
#ifdef %{up-nhm}

#ifdef %{physicalfirst}
bind= 0, 1, 2, 3, 4, 5, 6, 7
#endif

#ifdef %{physicallogical}
bind= 0, 2, 4, 6, 1, 3, 5, 7
#endif

#endif

#ifdef %{dp-nhm}

#ifdef %{physicalfirst}

```

```

bind= 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
#endif

#ifdef %{physicallogical}
bind= 0, 2, 4, 6, 8, 10, 12, 14, 1, 3, 5, 7, 9, 11, 13, 15
#endif

#endif

#ifdef %{4p-nhm}

bind= <<EOT
 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,
26, 27, 28, 29, 30, 31,
32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57,
58, 59, 60, 61, 62, 63
EOT

#endif

submit          = numactl --localalloc --physcpubind=$BIND $command

#####
# Compiler options
# for Nehalem use -xSSE4.2
# for processors prior to dunnington, replace -xSSE4.1 with -xSSE3
#####

SSE              = -xSSE4.2
FAST             = $(SSE) -ipo -O3 -no-prec-div -static
FASTNOSTATIC    = $(SSE) -ipo -O3 -no-prec-div

#####
#
# portability & libraries
#
##### Portability Flags and Notes #####

400.perlbench=default:
CPORTABILITY=      -DSPEC_CPU_LINUX_IA32

403.gcc=default:
EXTRA_CFLAGS=      -Dalloca=_alloca

462.libquantum=default:
CPORTABILITY=      -DSPEC_CPU_LINUX

483.xalancbmk=default:
CXXPORTABILITY=    -DSPEC_CPU_LINUX

fp=default:
PORTABILITY = -DSPEC_CPU_LP64

435.gromacs=default=default=default:
LDPORTABILITY = -nofor_main

436.cactusADM=default=default=default:
LDPORTABILITY = -nofor_main

454.calculix=default=default=default:
LDPORTABILITY = -nofor_main

481.wrf=default=default=default:
CPORTABILITY = -DSPEC_CPU_CASE_FLAG -DSPEC_CPU_LINUX

# Tuning Flags
#####

```

```

#
# Base tuning default optimization
# Feedback directed optimization not allowed in baseline for CPU2006
# However there is no limit on the number of flags as long as the same
# flags are used in the same order for all benchmarks of a given language

471.omnetpp,473.astar,483.xalancbmk=default:
EXTRA_LIBS= -L$(SMARTHEAP_DIR) -lsmarheap
EXTRA_LDFLAGS= -Wl,-z,muldefs

int=base=default=default:
COPTIMIZE= $(FAST) -inline-alloc -opt-malloc-options=3 -opt-prefetch
CXXOPTIMIZE= $(FASTNOSTATIC) -opt-prefetch

fp=base=default=default:
OPTIMIZE= $(FAST)

#####
# Peak Tuning Flags int 2006 fast
#####
int=peak=default:
COPTIMIZE= -opt-prefetch
CXXOPTIMIZE= -opt-prefetch
PASS1_CFLAGS = -prof-gen
PASS2_CFLAGS = $(FAST) -prof-use
PASS1_CXXFLAGS = -prof-gen
PASS2_CXXFLAGS = $(FASTNOSTATIC) -prof-use
PASS1_LDCFLAGS = -prof-gen
PASS2_LDCFLAGS = $(FAST) -prof-use
PASS1_LDCXXFLAGS = -prof-gen
PASS2_LDCXXFLAGS = $(FASTNOSTATIC) -prof-use

400.perlbench=peak=default:
COPTIMIZE= -ansi-alias -opt-prefetch

401.bzip2=peak=default:
CC= /opt/intel/Compiler/11.0/080/bin/intel64/icc
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= -opt-prefetch -ansi-alias -auto-ilp32

403.gcc=peak=default:
COPTIMIZE = $(FAST) -inline-alloc -opt-malloc-options=3
feedback=0

429.mcf=peak=default:
COPTIMIZE= -opt-prefetch
#####
#####
%ifdef %{up-nhm}
copies=4
%endif
%ifdef %{dp-nhm}
copies=8
%endif
#####
#####

445.gobmk=peak=default:
COPTIMIZE= -O2 -ipo -no-prec-div -ansi-alias
PASS1_CFLAGS = -prof-gen
PASS2_CFLAGS = $(SSE) -prof-use
PASS1_LDCFLAGS = -prof-gen
PASS2_LDCFLAGS = $(SSE) -prof-use

456.hmmmer=peak=default:
CC= /opt/intel/Compiler/11.0/080/bin/intel64/icc
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= $(FAST) -unroll2 -ansi-alias -auto-ilp32
feedback=no

```

```

458.sjeng=peak=default:
CC= /opt/intel/Compiler/11.0/080/bin/intel64/icc
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= -unroll4 -auto-ilp32

462.libquantum=peak=default:
COPTIMIZE= $(FAST) -opt-malloc-options=3 -opt-prefetch
feedback=no

464.h264ref=peak=default:
COPTIMIZE= -unroll2 -ansi-alias

471.omnetpp=peak=default:
basepeak=yes

473.astar=peak=default:
CXX= /opt/intel/Compiler/11.0/080/bin/intel64/icpc
CXXPORTABILITY= -DSPEC_CPU_LP64
EXTRA_LIBS= -L$(SMARTHEAP_DIR) -lsmartheap64
CXXOPTIMIZE= -ansi-alias -opt-ra-region-strategy=routine -auto-ilp32

483.xalancbmk=peak=default:
basepeak=yes

#####
# Peak Tuning Flags for FP
#####
fp=peak=default:
COPTIMIZE= -auto-ilp32
CXXOPTIMIZE= -auto-ilp32
PASS1_CFLAGS = -prof-gen
PASS2_CFLAGS = $(FAST) -prof-use
PASS1_CXXFLAGS = -prof-gen
PASS2_CXXFLAGS = $(FAST) -prof-use
PASS1_FFLAGS = -prof-gen
PASS2_FFLAGS = $(FAST) -prof-use
PASS1_LDFLAGS = -prof-gen
PASS2_LDFLAGS = $(FAST) -prof-use

410.bwaves=peak=default:
OPTIMIZE= $(FAST) -opt-prefetch
feedback=0
#####
#####
#ifdef %{up-nhm}
copies=4
#endif
#ifdef %{dp-nhm}
copies=8
#endif
#####
#####

416.gamess=peak=default:
OPTIMIZE= -unroll2 -Ob0 -ansi-alias -scalar-rep-

433.milc=peak=default:
OPTIMIZE= -fno-alias
COPTIMIZE=

435.gromacs=peak=default:
OPTIMIZE= -opt-prefetch

436.cactusADM=peak=default:
OPTIMIZE= -unroll2 -opt-prefetch

437.leslie3d=peak=default:
PORTABILITY =
FC= ifort -m32
OPTIMIZE= -opt-malloc-options=3 -opt-prefetch

```

```

444.namd=peak=default:
CXXOPTIMIZE= -fno-alias -auto-ilp32

447.dealII=peak=default:
CXXOPTIMIZE= -unroll2 -ansi-alias -scalar-rep-

450.soplex=peak=default:
PORTABILITY =
CXX= icpc -m32
OPTIMIZE= -opt-malloc-options=3
CXXOPTIMIZE=
#####
#####
#ifdef %{up-nhm}
copies=4
#endif
#ifdef %{dp-nhm}
copies=8
#endif
#####
#####

453.povray=peak=default:
CXXOPTIMIZE= -unroll4 -ansi-alias

454.calculix=peak=default:
OPTIMIZE= $(FAST)
feedback=0

459.GemsFDTD=peak=default:
OPTIMIZE= -unroll2 -Ob0 -opt-prefetch

465.tonto=peak=default:
OPTIMIZE= -unroll4 -auto

470.lbm=peak=default:
OPTIMIZE= $(FAST) -opt-prefetch
feedback=no
#####
#####
#ifdef %{up-nhm}
copies=4
#endif
#ifdef %{dp-nhm}
copies=8
#endif
#####
#####

481.wrf=peak=default:
basepeak=yes

482.sphinx3=peak=default:
PORTABILITY =
CC= icc -m32
OPTIMIZE= $(FAST)
COPTIMIZE= -unroll2
feedback=no

#####
# (Edit this to match your system)
#####

default=default=default=default:
license_num      = 3184
test_sponsor     = Intel Corporation
hw_avail         = Feb-2009
sw_avail         = Feb-2009
tester           = Principled Technologies, Inc.

```

```

hw_cpu_name      = Intel Xeon X5570
hw_cpu_char     = 1600MHz system bus
hw_cpu_mhz      = 2930
hw_disk         = 1 x 160 GB 7200 RPM SATA
hw_fpu          = Integrated
hw_memory       = 24 GB (6 x 4 GB DDR3-10600)
hw_model        = Supermicro SuperServer
hw_ncpuorder    = 1,2 chip
hw_ncores       = 8
hw_nchips       = 2
hw_ncoresperchip = 4
hw_nthreadspercore = 2
hw_other        = None
hw_pcache       = 32 KB I + 32 KB D on chip per core
hw_scache       = 256 KB I+D on chip per core
hw_tcache       = 8 MB
hw_ocache       = None
hw_vendor       = Supermicro Computers, Inc.
prepared_by     = Principled Technologies, Inc.
sw_file         = ReiserFS
sw_os           = SuSe Linux SLES10 SP2
sw_state        = Run level 3 (multi-user)
notes_submit_000 = numactl was used to bind copies to the cores

```

```

int=default=default=default:
sw_compiler001  = Intel C++ Compiler 11.0 for Linux
sw_compiler002  = Build 20090131 Package ID: l_cproc_p_11.0.080
sw_base_ptrsize = 32-bit
sw_peak_ptrsize = 32/64-bit
sw_other001     = Microquill SmartHeap V8.1
sw_other002     = Binutils 2.18.50.0.7.20080502

```

```

fp=default=default=default:
sw_compiler001  = Intel C++ and Fortran Compiler 11.0 for Linux
sw_compiler002  = Build 20090131 Package ID: l_cproc_p_11.0.080, l_cprof_p_11.0.080
sw_base_ptrsize = 64-bit
sw_peak_ptrsize = 32/64-bit
sw_other001     = Binutils 2.18.50.0.7.20080502

```

SUSE Linux Enterprise Server 10: Intel Xeon processor E5450-based server

```

# Invocation command line:
# /usr/cpu2006/bin/runspec --rate 8 -c cpu2006.1.1.ic11.0.linux64.dgt.rate.cfg --flagsurl=Intel-ic11.0-
int-linux64-revA.xml,Intel-Linux64-Platform.xml --define stoakleydp8cores -T base -o asc int
# output_root was not used for this run
#####
#####
# This is a sample config file. It was tested with:
#
#   Compiler name/version:      Intel Compiler 11.0
#   Operating system version:    64-Bit SUSE LINUX Enterprise Server 10 or later
#   Hardware:                   Intel Core 2 and compatible Intel processors
#                               supporting Intel 64 and SSE4.1
#
#####
# SPEC CPU2006 Intel Linux64 config file
# July 2008 Intel Compiler 11.0 for Linux64
#####
action      = validate
tune        = base
ext         = cpu2006.1.1.ic11.0.linux64.dgt.rate
PATHSEP     = /
check_md5=1
reportable=1

#
# These are listed as benchmark-tuning-extension-machine
#

```

```

default=default=default=default:
CC = icc
CXX = icpc
FC=ifort
OBJ = .o
SMARTHEAP_DIR = /usr/cpu2006/SmartHeap_8.1/lib

submit= MYMASK=`printf '0x%x' \${(1<<\$SPECCOPYNUM)}`; taskset \${MYMASK} $command

#####
# Compiler options
# for Dunnington use -xSSE4.1
# for processors prior to dunnington, replace -xSSE4.1 with -xSSSE3
#####

SSE = -xSSE4.1
FAST = $(SSE) -ipo -O3 -no-prec-div -static
FASTNOSTATIC = $(SSE) -ipo -O3 -no-prec-div

#####
#
# portability & libraries
#
##### Portability Flags and Notes #####

400.perlbench=default:
CPORTABILITY= -DSPEC_CPU_LINUX_IA32

403.gcc=default:
EXTRA_CFLAGS= -Dalloca=_alloca

462.libquantum=default:
CPORTABILITY= -DSPEC_CPU_LINUX

483.xalancbmk=default:
CXXPORTABILITY= -DSPEC_CPU_LINUX

fp=default:
PORTABILITY = -DSPEC_CPU_LP64

435.gromacs=default=default=default:
LDPORTABILITY = -nofor_main

436.cactusADM=default=default=default:
LDPORTABILITY = -nofor_main

454.calculix=default=default=default:
LDPORTABILITY = -nofor_main

481.wrf=default=default=default:
CPORTABILITY = -DSPEC_CPU_CASE_FLAG -DSPEC_CPU_LINUX

# Tuning Flags
#####

#
# Base tuning default optimization
# Feedback directed optimization not allowed in baseline for CPU2006
# However there is no limit on the number of flags as long as the same
# flags are used in the same order for all benchmarks of a given language

471.omnetpp,473.astar,483.xalancbmk=default:
EXTRA_LIBS= -L$(SMARTHEAP_DIR) -lsmarheap
EXTRA_LDFLAGS= -Wl,-z,muldefs

int=base=default=default:
COPTIMIZE= $(FAST) -inline-alloc -opt-malloc-options=3 -opt-prefetch
CXXOPTIMIZE= $(FASTNOSTATIC) -opt-prefetch

```



```

fp=base=default=default:
OPTIMIZE= $(FAST) -opt-prefetch

#####
# Peak Tuning Flags int 2006 fast
#####
int=peak=default:
COPTIMIZE= $(FAST) -opt-prefetch
CXXOPTIMIZE= $(FASTNOSTATIC) -opt-prefetch
PASS1_CFLAGS = -prof-gen
PASS2_CFLAGS = -prof-use
PASS1_CXXFLAGS = -prof-gen
PASS2_CXXFLAGS = -prof-use
PASS1_LDFLAGS = -prof-gen
PASS2_LDFLAGS = -prof-use

400.perlbench=peak=default:
COPTIMIZE= $(FAST) -ansi-alias -opt-prefetch

401.bzip2=peak=default:
CC= /opt/intel/Compiler/11.0/042/bin/intel64/icc -L/opt/intel/Compiler/11.0/042/ipp/em64t/lib -
I/opt/intel/Compiler/11.0/042/ipp/em64t/include
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= $(FAST) -opt-prefetch -ansi-alias

403.gcc=peak=default:
COPTIMIZE = $(FAST) -inline-calloc -opt-malloc-options=3
feedback=0

429.mcf=peak=default:
COPTIMIZE= $(FAST) -opt-prefetch

445.gobmk=peak=default:
COPTIMIZE= $(SSE) -O2 -ipo -no-prec-div -ansi-alias

456.hmmer=peak=default:
CC= /opt/intel/Compiler/11.0/042/bin/intel64/icc -L/opt/intel/Compiler/11.0/042/ipp/em64t/lib -
I/opt/intel/Compiler/11.0/042/ipp/em64t/include
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= $(FAST) -unroll2 -ansi-alias
feedback=no

458.sjeng=peak=default:
COPTIMIZE= $(FAST) -unroll4

462.libquantum=peak=default:
COPTIMIZE= $(FAST) -opt-malloc-options=3 -parallel -par-runtime-control -opt-prefetch
feedback=no
copies=1
submit=

464.h264ref=peak=default:
COPTIMIZE= $(FAST) -unroll2 -ansi-alias

471.omnetpp=peak=default:
CXXOPTIMIZE= $(FASTNOSTATIC) -ansi-alias -opt-ra-region-strategy=block

473.astar=peak=default:
CXXOPTIMIZE= $(FASTNOSTATIC) -ansi-alias -opt-ra-region-strategy=routine

483.xalancbmk=peak=default:
basepeak=yes

#####
# Peak Tuning Flags for FP
#####
fp=peak=default:
OPTIMIZE= $(FAST)
COPTIMIZE= -auto-ilp32

```

```

CXXOPTIMIZE= -auto-ilp32
PASS1_CFLAGS = -prof-gen
PASS2_CFLAGS = -prof-use
PASS1_CXXFLAGS = -prof-gen
PASS2_CXXFLAGS = -prof-use
PASS1_FFLAGS = -prof-gen
PASS2_FFLAGS = -prof-use
PASS1_LDFLAGS = -prof-gen
PASS2_LDFLAGS = -prof-use

410.bwaves=peak=default:
OPTIMIZE= $(FAST) -opt-prefetch
feedback=0
#####
#####
%ifdef %{stoakleydp8cores}
copies=4
submit= if `test $SPEC COPYPNUM -le 1`; then taskset -c $SPEC COPYPNUM $command ; else taskset -c `expr
$SPEC COPYPNUM + 4` $command ; fi
%endif
#####
#####

416.gamess=peak=default:
OPTIMIZE= $(FAST) -unroll2 -Ob0 -ansi-alias -scalar-rep-

433.milc=peak=default:
OPTIMIZE= $(FAST) -fno-alias
COPTIMIZE=

435.gromacs=peak=default:
OPTIMIZE= $(FAST) -opt-prefetch

436.cactusADM=peak=default:
OPTIMIZE= $(FAST) -unroll2 -opt-prefetch -parallel
copies=1
submit=

437.leslie3d=peak=default:
PORTABILITY =
FC= /opt/intel/Compiler/11.0/042/bin/ia32/ifort -L/opt/intel/Compiler/11.0/042/ipp/ia32/lib -
I/opt/intel/Compiler/11.0/042/ipp/ia32/include
OPTIMIZE= $(FAST) -opt-malloc-options=3 -opt-prefetch
#####
#####
%ifdef %{stoakleydp8cores}
copies=4
submit= if `test $SPEC COPYPNUM -le 1`; then taskset -c $SPEC COPYPNUM $command ; else taskset -c `expr
$SPEC COPYPNUM + 4` $command ; fi
%endif
#####
#####

444.namd=peak=default:
CXXOPTIMIZE= $(FAST) -fno-alias -auto-ilp32

447.dealII=peak=default:
CXXOPTIMIZE= $(FAST) -unroll2 -ansi-alias -scalar-rep-

450.soplex=peak=default:
PORTABILITY =
CXX= /opt/intel/Compiler/11.0/042/bin/ia32/icpc -L/opt/intel/Compiler/11.0/042/ipp/ia32/lib -
I/opt/intel/Compiler/11.0/042/ipp/ia32/include
OPTIMIZE= $(FAST) -opt-malloc-options=3
CXXOPTIMIZE=

453.povray=peak=default:
CXXOPTIMIZE= $(FAST) -unroll4 -ansi-alias

454.calculix=peak=default:

```

```

OPTIMIZE= $(FAST)
feedback=0

459.GemsFDTD=peak=default:
OPTIMIZE= $(FAST) -unroll2 -Ob0 -opt-prefetch

465.tonto=peak=default:
OPTIMIZE= $(FAST) -unroll4 -auto

470.lbm=peak=default:
OPTIMIZE= $(FAST) -opt-prefetch
feedback=no

#####
#####
%ifdef %{xeon3000}
copies=2
%endif

%ifdef %{wolfdaledp4cores}
copies=2
%endif

%ifdef %{bensleydp8cores}
copies=2
submit= if `test $SPEC COP YNUM -le 0`; then taskset -c $SPEC COP YNUM $command ; else taskset -c `expr
$SPEC COP YNUM + 4` $command ; fi
%endif

%ifdef %{stoakleydp8cores}
copies=4
submit= if `test $SPEC COP YNUM -le 1`; then taskset -c $SPEC COP YNUM $command ; else taskset -c `expr
$SPEC COP YNUM + 4` $command ; fi
%endif

%ifdef %{tgt4p16cores}
copies=8
submit= if `test $SPEC COP YNUM -le 3`; then taskset -c $SPEC COP YNUM $command ; else taskset -c `expr
$SPEC COP YNUM + 8` $command ; fi
%endif

%ifdef %{dgt4p24cores}
copies=12
submit= $[top]/mysubmit.pl $SPEC COP YNUM "$command"
%endif

%ifdef %{dgt4p16cores}
copies=8
submit= $[top]/mysubmit.pl $SPEC COP YNUM "$command"
%endif
#####
#####

481.wrf=peak=default:
basepeak=yes

482.sphinx3=peak=default:
PORTABILITY =
CC= /opt/intel/Compiler/11.0/042/bin/ia32/icc -L/opt/intel/Compiler/11.0/042/ipp/ia32/lib -
I/opt/intel/Compiler/11.0/042/ipp/ia32/include
COPTIMIZE= -unroll2
feedback=no

#####
#####
%ifdef %{wolfdaledp4cores}
copies=2
%endif

%ifdef %{stoakleydp8cores}

```

```

copies=4
submit= if `test $$SPECCOPYNUM -le 1`; then taskset -c $$SPECCOPYNUM $command ; else taskset -c `expr
$$SPECCOPYNUM + 4` $command ; fi
%endif

%ifdef %{tgt4p16cores}
copies=8
submit= if `test $$SPECCOPYNUM -le 3`; then taskset -c $$SPECCOPYNUM $command ; else taskset -c `expr
$$SPECCOPYNUM + 8` $command ; fi
%endif

%ifdef %{dgt4p24cores}
copies=12
submit= $[top]/mysubmit.pl $$SPECCOPYNUM "$command"
%endif

%ifdef %{dgt4p16cores}
copies=8
submit= $[top]/mysubmit.pl $$SPECCOPYNUM "$command"
%endif
#####
#####

#####
# (Edit this to match your system)
#####

default=default=default=default:
license_num      = 3184
test_sponsor     = Intel Corp.
hw_avail         = Sep-2008
# SLES 10 SP2 was released in May
sw_avail         = Jan-2009
test_date        = Feb-2009
tester           = Principled Technologies, Inc.
hw_cpu_name      = Intel Xeon E5450
hw_cpu_char      =
hw_cpu_mhz       = 3000
hw_disk          = 1 x 160 GB 7200 RPM SATA
hw_fpu           = Integrated
hw_memory        = 16 GB (8 x 2 GB DDR2-667)
hw_model000      = Supermicro SuperServer 6025B-TR+ (Intel Xeon E5450, 3.00 GHz)
hw_ncpuorder     = 1,2 chips
hw_ncores        = 8
hw_nchips        = 2
hw_ncoresperchip = 4
hw_nthreadspercore = 1
hw_other         = None
hw_pcache        = 32 KB I + 32 KB D on chip per core
hw_scache        = 12 MB I+D on chip per chip, 6 MB shared / 2 cores
hw_tcache        = None
hw_ocache        = None
hw_vendor        = Supermicro Computer, Inc.
prepared_by      = Principled Technologies, Inc.
sw_file          = ReiserFS
sw_os000         = SUSE Linux Enterprise Server 10 (x86_64) SP2,
sw_os001 = Kernel 2.6.16-60.0.21-smp
sw_state         = Run level 3 (multi-user)
sw_other000      = Microquill SmartHeap V8.1
sw_other001      = Binutils 2.19
notes_os_000     = 'ulimit -s unlimited' was used to set the stack size to unlimited prior to run
notes_plat_000   = BIOS Settings:
# Remove if the settings were default (default for HBW is Disabled)
notes_plat_005   = Hardware Prefetcher = Disabled (Default = Enabled)

```

```

notes_plat_010 = Adjacent Cache Line Prefetch = Disabled (Default = Enabled)

int=default=default=default:
sw_compiler000 = Intel C++ Compiler 11.0 for Linux
sw_compiler001 = Build 20080730 Package ID: l_cproc_b_11.0.074
sw_base_ptrsize = 32-bit
sw_peak_ptrsize = Not Applicable
notes_000 = taskset was used to bind processes to cores except
notes_005 = for 462.libquantum peak

fp=default=default=default:
sw_compiler001 = Intel C++ and Fortran Compiler 11.0 for Linux
sw_compiler002 = Build 20080730 Package ID: l_cc_b_11.0.074, l_fc_b_11.0.074
sw_base_ptrsize = 64-bit
sw_peak_ptrsize = 32/64-bit
notes_026 = taskset was used to bind processes to cores except
notes_027 = for 436.cactusADM peak
notes_028 = OMP_NUM_THREADS set to number of processors
notes_029 = KMP_AFFINITY set to "physical,0"
notes_030 = KMP_STACKSIZE set to 64M
%if defined(%{dgt4p24cores}) || defined(%{dgt4p16cores})
notes_submit_000 = For peak modules using 1/2 the number of available cores, each
notes_submit_005 = copy was assigned to a single L2 cache using a Perl script;
notes_submit_010 = the script can be found in the flags description file.
%endif

```

```

# The following section was added automatically, and contains settings that
# did not appear in the original configuration file, but were added to the
# raw file after the run.
default:
flagsurl000 = http://www.spec.org/cpu2006/flags/Intel-ic11.0-int-linux64-revA.20081001.xml
flagsurl001 = http://www.spec.org/cpu2006/flags/Intel-Linux64-Platform.20081014.00.xml

```

SUSE Linux Enterprise Server 10: Intel Xeon processor 5160-based server

```

# Invocation command line:
# /usr/cpu2006/bin/runspec --rate 4 -c cpu2006.1.0.ic10.linux64.int.may72007.cfg -T base -o asc int
# output_root was not used for this run
#####
#####
# This is a sample config file. It was tested with:
#
# Compiler name/version: Intel Compiler 11.0
# Operating system version: 64-Bit SUSE LINUX Enterprise Server 10 or later
# Hardware: Intel Core 2 and compatible Intel processors
# supporting Intel 64 and SSSE3
#
#####
# SPEC CPU2006 Intel Linux32 config file
# May 7 2007 Intel Compiler 10.0 for Linux32
#####
action = validate
tune = base
ext = cpu2006.1.0.ic10.linux64.int.may72007
PATHSEP = /
check_md5=1
reportable=1

#
# These are listed as benchmark-tuning-extension-machine
#
default=default=default=default:
CC = icc
CXX = icpc

```

```

FC = ifort
OBJ = .o
SMARTHEAP_DIR = /usr/cpu2006/SmartHeap_8.1/lib
sw_base_ptrsize = 32-bit
sw_peak_ptrsize = 32/64-bit

submit= MYMASK=`printf '0x%x' \${(1<<\$SPECCOPYNUM)}`; taskset \$MYMASK $command

#####
#
# portability & libraries
#
##### Portability Flags and Notes #####

400.perlbench=default:
CPORTABILITY=      -DSPEC_CPU_LINUX_IA32

403.gcc=default:
EXTRA_CFLAGS=      -Dalloca=_alloca

435.gromacs=default:
LDPORTABILITY=     -nofor_main

436.cactusADM=default:
LDPORTABILITY=     -nofor_main

454.calculix=default:
LDPORTABILITY=     -nofor_main

462.libquantum=default:
CPORTABILITY=      -DSPEC_CPU_LINUX

481.wrf=default:
CPORTABILITY=      -DSPEC_CPU_LINUX -DSPEC_CPU_CASE_FLAG

483.xalancbmk=default:
CXXPORTABILITY=    -DSPEC_CPU_LINUX

# Tuning Flags
#####

#
# Base tuning default optimization
# Feedback directed optimization not allowed in baseline for CPU2006
# However there is no limit on the number of flags as long as the same
# flags are used in the same order for all benchmarks of a given language

all_cpp=default:
EXTRA_LIBS=        -L$(SMARTHEAP_DIR) -lsmarheap
EXTRA_LDFLAGS=     -Wl,-z,muldefs

int-base=default:
COPTIMIZE=         -fast
CXXOPTIMIZE=       -xT -ipo -O3 -no-prec-div

#####
# Peak Tuning Flags int 2006 fast
#####
int=peak=default:
COPTIMIZE=         -fast
CXXOPTIMIZE=       -xT -ipo -O3 -no-prec-div
PASS1_CFLAGS =     -prof-gen
PASS2_CFLAGS =     -prof-use
PASS1_CXXFLAGS =   -prof-gen
PASS2_CXXFLAGS =   -prof-use
PASS1_LDFLAGS =    -prof-gen
PASS2_LDFLAGS =    -prof-use

400.perlbench=peak=default:
COPTIMIZE=         -fast -ansi-alias -prefetch

```

```

401.bzip2=peak=default:
CC= /opt/intel/Compiler/11.0/074/bin/intel64/icc -L/opt/intel/Compiler/11.0/074/lib/intel64 -
I/opt/intel/Compiler/11.0/074/include/intel64
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= -fast

403.gcc=peak=default:
basepeak=yes

429.mcf=peak=default:
COPTIMIZE= -fast -prefetch
feedback=0

445.gobmk=peak=default:
COPTIMIZE= -xT -O2 -ipo -no-prec_div -ansi-alias

456.hmmmer=peak=default:
CC= /opt/intel/Compiler/11.0/074/bin/intel64/icc -L/opt/intel/Compiler/11.0/074/lib/intel64 -
I/opt/intel/Compiler/11.0/074/include/intel64
CPORTABILITY= -DSPEC_CPU_LP64
COPTIMIZE= -fast -unroll2 -ansi-alias

458.sjeng=peak=default:
COPTIMIZE= -fast -unroll4

462.libquantum=peak=default:
COPTIMIZE= -fast -unroll4 -Ob0 -prefetch -opt-streaming-stores always

464.h264ref=peak=default:
COPTIMIZE= -fast -unroll2 -ansi-alias

471.omnetpp=peak=default:
CXXOPTIMIZE= -xT -O3 -ipo -no-prec_div -ansi-alias

473.astar=peak=default:
CXXOPTIMIZE= -xT -O3 -ipo -no-prec_div -ansi-alias

483.xalancbmk=peak=default:
basepeak=yes

#####
# (Edit this to match your system)
#####

default=default=default=default:
license_num      = 3184
test_sponsor     = Intel Corp.
hw_avail         = Jul-2008
sw_avail         = Jan-2009
test_date       = Feb-2009
tester          = Principled Technologies, Inc.
hw_cpu_name     = Intel Xeon 5160
hw_cpu_char     = 1333MHz system bus
hw_cpu_mhz      = 3000
hw_disk         = 1 x 160 GB SAS, 7200 RPM
hw_fpu          = Integrated
hw_memory       = 16 GB (8 x 2GB DDR2-5300F)
hw_model        = Supermicro SuperServer 6025B-TR+ (Intel Xeon 5160)
hw_ncpuorder    = 1,2 chips
hw_ncores       = 4
hw_nchips       = 2
hw_ncoresperchip = 2
hw_nthreadspercore = 1
hw_other        = None
sw_auto_parallel = No
hw_pcache       = 32 KB I + 32 KB D on chip per core
hw_scache       = 4 MB I+D on chip per chip
hw_tcache       = None
hw_ocache       = None

```

```

hw_vendor      = Supermicro Computer, Inc.
prepared_by   = Principled Technologies, Inc.
sw_file       = ReiserFS
sw_os         = SLES 10 (x86_64) SP2, 2.6.16-60.0.21-smp
sw_state      = Multi-user, run level 3
sw_other      = MicroQuill SmartHeap 8.1

int=default=default=default:
sw_compiler000 = Intel C++ Compiler for Linux version 11.0
sw_compiler001 = Build 20070426 Package ID: l_cc_p_11.0.074
sw_auto_parallel = No

fp=default=default=default:
sw_compiler001 = Intel C++ Compiler for Linux32 version 11.0
sw_compiler002 = Build 20070426 Package ID: l_cc_p_11.0.074
sw_compiler003 = Intel Fortran Compiler for Linux32 version 11.0
sw_compiler004 = Build 20070426 Package ID: l_fc_p_11.0.074
sw_auto_parallel = No

```

```

# The following section was added automatically, and contains settings that
# did not appear in the original configuration file, but were added to the
# raw file after the run.
default:
flagsurl000= http://www.spec.org/cpu2006/flags/Intel-ic10-ia32-intel64-linux-flags-file-
20070525.xml.20070828.xml
notes_000 = taskset utility used to bind CPU(s) to processes

```

SUSE Linux Enterprise Server 10: Intel Xeon processor 3.6 GHz-based server

```

# Invocation command line:
# /usr/cpu2006/bin/runspec -c cpu2006-ic91.em64t.cfg -r 4 -v 7 -T base int
# output_root was not used for this run
#####
#####
# SPEC CPU2006 Intel Linux64 config file
# Feb 2009 Intel Compiler 9.1 for Linux64
#####

action      = validate
tune        = base
ext         = ic91.em64t
size        = ref
flagsurl000 = NEC-ic91-linux-flags.xml
output_format = all

check_md5   = 1
mean_anyway = 1
reportable  = 1
verbose     = 6

#####
# File to be included into config file for SPEC CPU2006

# This include file specifies the "System Under Test" (Hardware,
# OS, BIOS, etc.)
# Any information about the binaries is elsewhere.

# To avoid conflicts in the numbering of the notes,
# please use only 900 and higher numbers.

#####
# System information #
# If some remarks about BIOS or Firmware are needed, place them here. #
#####

```



```

default=default=default=default:
notes_os_000 = 'ulimit -s unlimited' was used to set the stacksize to unlimited prior to run
notes_os_005 = '/usr/bin/taskset' used to bind processes to CPUs
notes_000 =
notes_005 = The system bus runs at 800 MHz
notes_010 =
notes_015 =

```

```

#####
# Description Hardware and Software #
#####

```

```

default=default=default=default:
hw_vendor = Supermicro Computer, Inc.
hw_model000 = Supermicro 6024H-T, Intel Xeon processor,
hw_model001 = 3.6 GHz
hw_cpu_name = Intel Xeon 3.6GHz
hw_cpu_char = 800 MHz system bus, Hyperthreading Enabled
hw_cpu_mhz = 3600
hw_fpu = Integrated
#
hw_nchips = 2
hw_ncores = 2
hw_ncoresperchip = 1
hw_nthreadspercore = 2
#
hw_ncpuorder = 1,2 chips
hw_pcache = 12 K micro-ops I + 16 KB D on chip per chip
hw_scache = 2 MB I+D on chip per chip
hw_tcache = None
hw_ocache = None
hw_memory = 16 GB (8x2 GB DDR PC2700R)
hw_disk000 = Western Digital WD1600AAJS-11B4A0
hw_disk001 = (SATA, 160GB, 7200rpm)
hw_other = None

sw_file = ReiserFS
sw_state = Multiuser, Runlevel 3

license_num = 3184
test_sponsor = Intel Corp.
tester = Principled Technologies, Inc.
hw_avail = Oct-2006
prepared_by = Principled Technologies, Inc.

```

```

default=default=default=default:
CC = icc
CXX = icpc
FC = ifort
OBJ = .o

```

```

SMARTHEAP_DIR = /usr/cpu2006/SmartHeap_8.1/lib

```

```

submit= MYMASK=`printf '0x%x' \${(1<<\$SPECCOPYNUM)}`; /usr/bin/taskset \${MYMASK} $command

```

```

#####
# portability & libraries #
#####

```

```

fp=default=default=default:
PORTABILITY = -DSPEC_CPU_LP64

```

```

400.perlbench=default=default=default:
CPORTABILITY = -DSPEC_CPU_LINUX_X64

```

```

462.libquantum=default=default=default:
CPORTABILITY = -DSPEC_CPU_LINUX

```

```

483.xalancbmk=default=default=default:

```

```

CXXPORTABILITY = -DSPEC_CPU_LINUX

435.gromacs=default=default=default:
LDPORTABILITY = -nofor_main

436.cactusADM=default=default=default:
LDPORTABILITY = -nofor_main

454.calculix=default=default=default:
LDPORTABILITY = -nofor_main

481.wrf=default=default=default:
CPORTABILITY = -DSPEC_CPU_CASE_FLAG -DSPEC_CPU_LINUX

#####
# Baseline Tuning Flags #
#####

int=base=default=default:
COPTIMIZE= -fast
CXXOPTIMIZE= -xP -O3 -ipo -no-prec-div
EXTRA_CXXLIBS= -L$(SMARTHEAP_DIR) -lsmartheap

fp=base=default=default:
OPTIMIZE= -fast

#####
# Peak Tuning Flags #
#####

int=peak=default=default:
OPTIMIZE= -fast
EXTRA_LIBS= -L$(SMARTHEAP_DIR) -lsmartheap

PASS1_CFLAGS = -prof_gen
PASS2_CFLAGS = -prof_use
PASS1_CXXFLAGS = -prof_gen
PASS2_CXXFLAGS = -prof_use
PASS1_LDFLAGS = -prof_gen
PASS2_LDFLAGS = -prof_use

fp=peak=default=default:
OPTIMIZE= -fast

PASS1_CFLAGS = -prof_gen
PASS2_CFLAGS = -prof_use
PASS1_FFLAGS = -prof_gen
PASS2_FFLAGS = -prof_use
PASS1_CXXFLAGS = -prof_gen
PASS2_CXXFLAGS = -prof_use
PASS1_LDFLAGS = -prof_gen
PASS2_LDFLAGS = -prof_use

400.perlbench=peak=default=default:
EXTRA_LIBS=

401.bzip2=peak=default=default:
CC=/opt/intel/cce/9.1.053/bin/icc -I/opt/intel/cce/9.1.053/include -L/opt/intel/cce/9.1.053/lib
PORTABILITY=-DSPEC_CPU_LP64
EXTRA_LIBS=
feedback=0

403.gcc=peak=default=default:
basepeak=yes

456.hmmmer=peak=default=default:
CC=/opt/intel/cce/9.1.053/bin/icc -I/opt/intel/cce/9.1.053/include -L/opt/intel/cce/9.1.053/lib
PORTABILITY=-DSPEC_CPU_LP64
EXTRA_LIBS=

```

```

462.libquantum=peak=default=default:
CC=/opt/intel/cce/9.1.053/bin/icc -I/opt/intel/cce/9.1.053/include -L/opt/intel/cce/9.1.053/lib
PORTABILITY=-DSPEC_CPU_LP64
EXTRA_LIBS=

471.omnetpp=peak=default=default:
OPTIMIZE=-xP -O3 -ipo -no-prec-div

483.xalancbmk=peak=default=default:
basepeak=yes

410.bwaves=peak=default=default:
basepeak=yes

416.gamess=peak=default=default:
basepeak=yes

433.milc=peak=default=default:
PORTABILITY=
CC=/opt/intel/cc/9.1.053/bin/icc -I/opt/intel/cc/9.1.053/include -L/opt/intel/cc/9.1.053/lib

434.zeusmp=peak=default=default:
PORTABILITY=
FC=/opt/intel/fc/9.1.052/bin/fort -I/opt/intel/fc/9.1.052/include -L/opt/intel/fc/9.1.052/lib
feedback=0

436.cactusADM=peak=default=default:
basepeak=yes

437.leslie3d=peak=default=default:
basepeak=yes

444.namd=peak=default=default:
basepeak=yes

450.soplex=peak=default=default:
PORTABILITY=
CXX=/opt/intel/cc/9.1.053/bin/icpc -I/opt/intel/cc/9.1.053/include -L/opt/intel/cc/9.1.053/lib

459.GemsFDTD=peak=default=default:
basepeak=yes

470.lbm=peak=default=default:
PORTABILITY=
CC=/opt/intel/cc/9.1.053/bin/icc -I/opt/intel/cc/9.1.053/include -L/opt/intel/cc/9.1.053/lib

481.wrf=peak=default=default:
basepeak=yes

482.sphinx3=peak=default=default:
PORTABILITY=
CC=/opt/intel/cc/9.1.053/bin/icc -I/opt/intel/cc/9.1.053/include -L/opt/intel/cc/9.1.053/lib
feedback=0

#####
# Used Compilers and OS #
#####

int=default=default=default:
sw_compiler000 = Intel C++ Compiler for IA32/EM64T applications
sw_compiler001 = Version 9.1 - Build 20080314,
sw_compiler002 = Package-ID: l_cc_p_9.1.053
sw_other      = Smart Heap Library, Version 8.1
sw_base_ptrsize = 32-bit
sw_peak_ptrsize = 32/64-bit

fp=default=default=default:
sw_compiler000 = Intel C++ Compiler for IA32/EM64T application,
sw_compiler001 = Version 9.1 - Build 20070215,
sw_compiler002 = Package-ID: l_cc_p_9.1.053

```

```
sw_compiler003 = Intel Fortran Compiler for IA32/EM64T application,  
sw_compiler004 = Version 9.1 - Build 20070215,  
sw_compiler005 = Package ID: l_fc_p_9.1.052  
sw_other      = None  
sw_base_ptrsize = 64-bit  
sw_peak_ptrsize = 32/64-bit
```

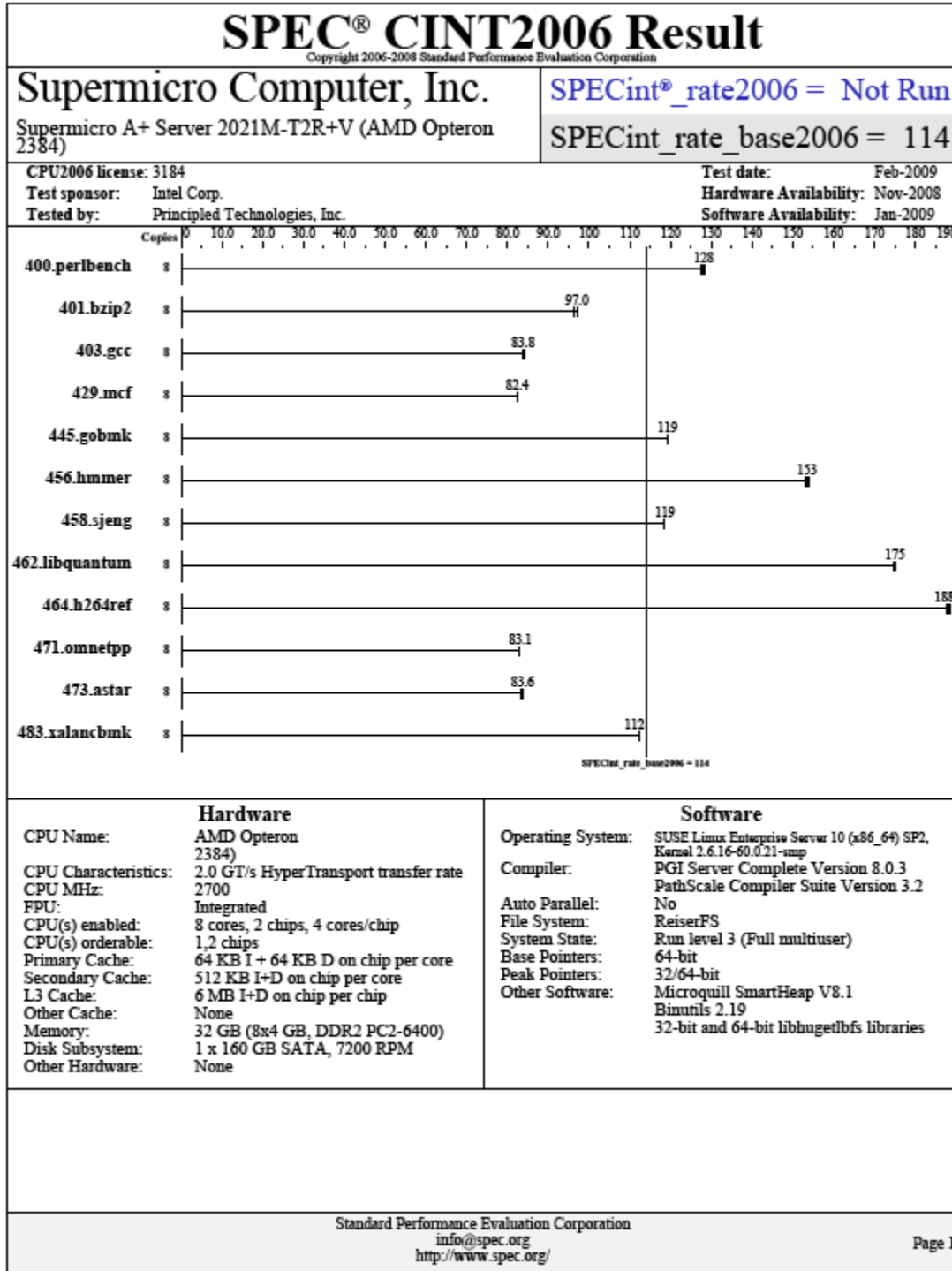
```
default=default=default=default:
```

```
sw_os000      = SUSE Linux Enterprise Server 10 (x86_64) SP2,  
sw_os001      = Kernel 2.6.16-60.0.21-smp  
sw_avail      = Apr-2008
```

```
#####  
# End of config-file #  
#####
```

Appendix C – SPECint_rate_base2006 output

This appendix provides the output of the benchmark for each of the test servers.
SUSE Linux Enterprise Server 10: AMD Opteron 2384-based server



SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro A+ Server 2021M-T2R+V (AMD Opteron 2384)

SPECint_rate_base2006 = 114

CPU2006 license: 3184

Test sponsor: Intel Corp.

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Nov-2008

Software Availability: Jan-2009

Results Table

Benchmark	Base							Peak						
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
400.perlbench	8	610	128	612	128	609	128							
401.bzip2	8	803	96.2	793	97.3	796	97.0							
403.gcc	8	768	83.8	769	83.7	763	84.3							
429.mcf	8	886	82.4	886	82.4	886	82.4							
445.gobmk	8	704	119	704	119	704	119							
456.hmmr	8	485	154	488	153	487	153							
458.sjeng	8	816	119	817	119	816	119							
462.libquantum	8	947	175	948	175	947	175							
464.h264ref	8	942	188	938	189	943	188							
471.omnetpp	8	602	83.1	602	83.1	602	83.0							
473.astar	8	670	83.9	671	83.6	673	83.5							
483.salanckmk	8	492	112	491	112	492	112							

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Submit Notes

The config file option 'submit' was used.

General Notes

taskset utility used to bind CPU(s) to processes

Base Compiler Invocation

C benchmarks:
pgcc

C++ benchmarks:
pgcpp

Base Portability Flags

400.perlbench: -DSPEC_CPU_LP64 -DSPEC_CPU_LINUX_X64
 401.bzip2: -DSPEC_CPU_LP64
 403.gcc: -DSPEC_CPU_LP64
 429.mcf: -DSPEC_CPU_LP64
 445.gobmk: -DSPEC_CPU_LP64
 456.hmmr: -DSPEC_CPU_LP64

Continued on next page

Standard Performance Evaluation Corporation
 info@spec.org
 http://www.spec.org/

Page 2

SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro A+ Server 2021M-T2R+V (AMD Opteron 2384)

SPECint_rate_base2006 = 114

CPU2006 license: 3184

Test sponsor: Intel Corp.

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Nov-2008

Software Availability: Jan-2009

Base Portability Flags (Continued)

458.sjeng: -DSPEC_CPU_LP64
462.libquantum: -DSPEC_CPU_LP64 -DSPEC_CPU_LINUX
464.h264ref: -DSPEC_CPU_LP64
483.xalanbmk: -DSPEC_CPU_LINUX

Base Optimization Flags

C benchmarks:

-Mvect-cachesize:6291456 -fastsse -Msmartalloc-huge -Mfprelaxed
-Mipa-fast -Mipa-inline -tp barcelona-64 -Bstatic_pgi

C++ benchmarks:

-Mvect-cachesize:6291456 -fastsse -Msmartalloc-huge -Mfprelaxed
--zc_eh -Mipa-fast -Mipa-inline:6 -tp barcelona-32 -Bstatic_pgi

Base Other Flags

C benchmarks:

-Mipa-jobs:4

C++ benchmarks:

-Mipa-jobs:4

SPEC and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. All other brand and product names appearing in this result are trademarks or registered trademarks of their respective holders.

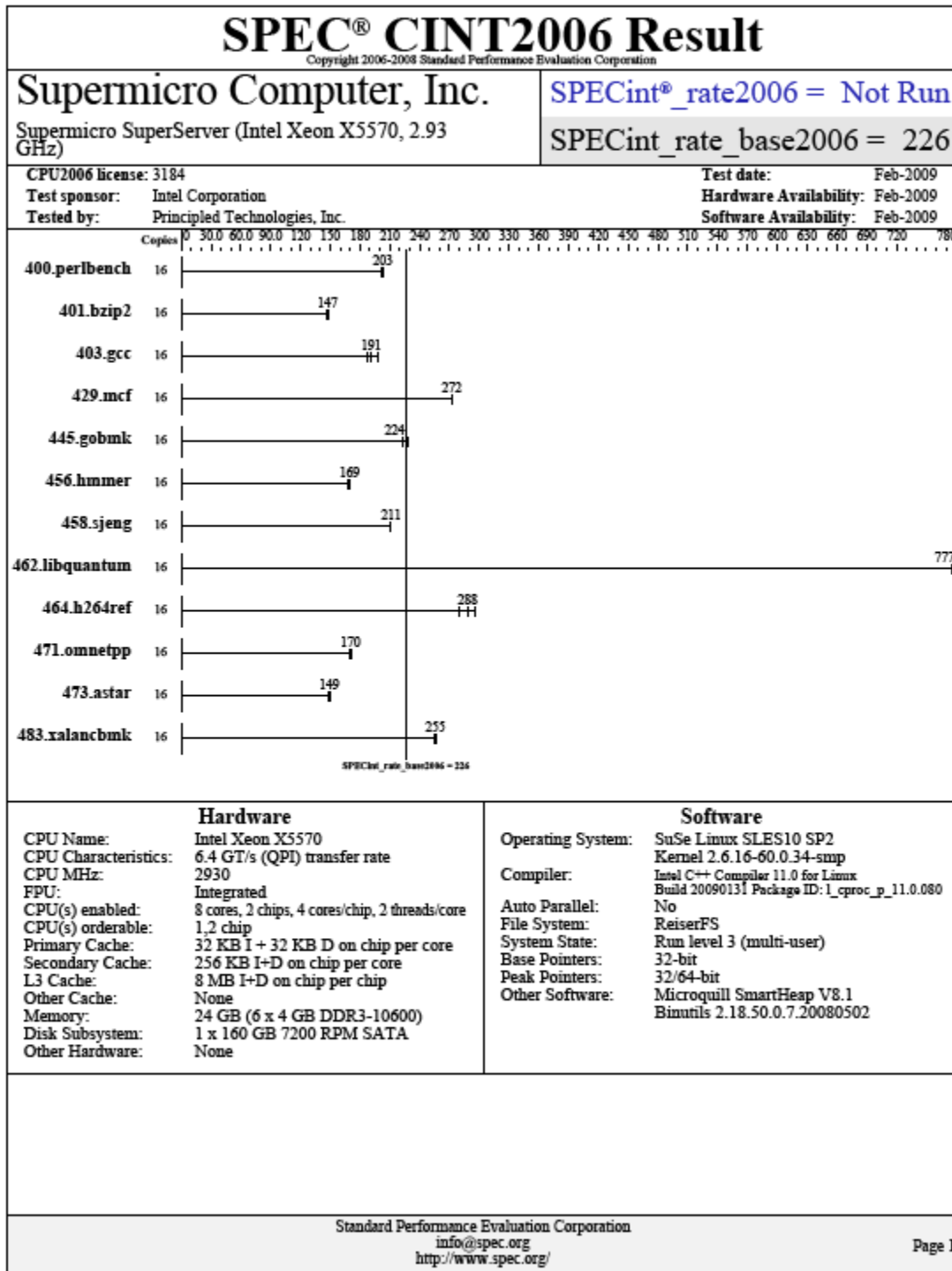
For questions about this result, please contact the tester.
For other inquiries, please contact webmaster@spec.org.

Tested with SPEC CPU2006 v1.1.
Report generated on Tue Mar 3 16:01:44 2009 by SPEC CPU2006 PS/PDF formatter v6128.

Standard Performance Evaluation Corporation
info@spec.org
<http://www.spec.org/>

Page 3

SUSE Linux Enterprise Server 10: Intel Xeon processor X5570-based server



SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro SuperServer (Intel Xeon X5570, 2.93 GHz)

SPECint_rate_base2006 = 226

CPU2006 license: 3184

Test sponsor: Intel Corporation

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Feb-2009

Software Availability: Feb-2009

Results Table

Benchmark	Base							Peak						
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
400.perlbench	16	768	203	776	201	771	203							
401.bzip2	16	1059	146	1047	147	1048	147							
403.gcc	16	651	198	674	191	686	188							
429.mcf	16	536	272	536	272	535	273							
445.gobmk	16	736	228	752	223	751	224							
456.hammer	16	881	169	880	170	887	168							
458.sjeng	16	917	211	917	211	918	211							
462.libquantum	16	427	777	426	779	427	777							
464.h264ref	16	1194	297	1230	288	1265	280							
471.omnetpp	16	587	170	586	171	587	170							
473.astar	16	753	149	755	149	755	149							
483.xalanbmk	16	429	257	433	255	432	255							

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Submit Notes

The config file option 'submit' was used.
numactl was used to bind copies to the cores

Base Compiler Invocation

C benchmarks:
icc

C++ benchmarks:
icpc

Base Portability Flags

400.perlbench: -DSPEC_CPU_LINUX_IA32
462.libquantum: -DSPEC_CPU_LINUX
483.xalanbmk: -DSPEC_CPU_LINUX

Base Optimization Flags

C benchmarks:
-xSSE4.2 -ipo -O3 -no-prec-div -static -inline-calloc
-opt-malloc-options-3 -opt-prefetch

Continued on next page

Standard Performance Evaluation Corporation
info@spec.org
http://www.spec.org/

Page 2

SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro SuperServer (Intel Xeon X5570, 2.93 GHz)

SPECint_rate_base2006 = 226

CPU2006 license: 3184

Test sponsor: Intel Corporation

Test date: Feb-2009

Tested by: Principled Technologies, Inc.

Hardware Availability: Feb-2009

Software Availability: Feb-2009

Base Optimization Flags (Continued)

C++ benchmarks:

-xSSE4.2 -ipo -O3 -no-prec-div -opt-prefetch -w1, -z, muldefs
-L/spec/cpu2006.1.1/lib -lsmartheap

Base Other Flags

C benchmarks:

403.gcc: -Dalloca-_alloca

SPEC and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. All other brand and product names appearing in this result are trademarks or registered trademarks of their respective holders.

For questions about this result, please contact the tester.
For other inquiries, please contact webmaster@spec.org.

Tested with SPEC CPU2006 v1.1.
Report generated on Fri Mar 6 15:55:13 2009 by SPEC CPU2006 PS/PDF formatter v6128.

Standard Performance Evaluation Corporation
info@spec.org
<http://www.spec.org/>

Page 3

SUSE Linux Enterprise Server 10: Intel Xeon processor E5450-based server

SPEC® CINT2006 Result <small>Copyright 2006-2008 Standard Performance Evaluation Corporation</small>																													
Supermicro Computer, Inc.	SPECint®_rate2006 = Not Run																												
Supermicro SuperServer 6025B-TR+ (Intel Xeon E5450, 3.00 GHz)	SPECint_rate_base2006 = 130																												
CPU2006 license: 3184	Test date: Feb-2009																												
Test sponsor: Intel Co.	Hardware Availability: Sep-2008																												
Tested by: Principled Technologies, Inc.	Software Availability: Jan-2009																												
<table border="1"> <caption>Performance Data from Chart</caption> <thead> <tr> <th>Benchmark</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>400.perlbenc</td><td>138</td></tr> <tr><td>401.bzip2</td><td>104</td></tr> <tr><td>403.gcc</td><td>103</td></tr> <tr><td>429.mcf</td><td>107</td></tr> <tr><td>445.gobmk</td><td>132</td></tr> <tr><td>456.hammer</td><td>131</td></tr> <tr><td>458.sjeng</td><td>139</td></tr> <tr><td>462.libquantum</td><td>173</td></tr> <tr><td>464.h264ref</td><td>263</td></tr> <tr><td>471.omnetpp</td><td>73.7</td></tr> <tr><td>473.astar</td><td>82.4</td></tr> <tr><td>483.xalanbmk</td><td>131</td></tr> <tr><td>SPECint_rate_base2006</td><td>130</td></tr> </tbody> </table>		Benchmark	Value	400.perlbenc	138	401.bzip2	104	403.gcc	103	429.mcf	107	445.gobmk	132	456.hammer	131	458.sjeng	139	462.libquantum	173	464.h264ref	263	471.omnetpp	73.7	473.astar	82.4	483.xalanbmk	131	SPECint_rate_base2006	130
Benchmark	Value																												
400.perlbenc	138																												
401.bzip2	104																												
403.gcc	103																												
429.mcf	107																												
445.gobmk	132																												
456.hammer	131																												
458.sjeng	139																												
462.libquantum	173																												
464.h264ref	263																												
471.omnetpp	73.7																												
473.astar	82.4																												
483.xalanbmk	131																												
SPECint_rate_base2006	130																												
Hardware	Software																												
CPU Name: Intel Xeon E5450 CPU Characteristics: 1333MHz system bus CPU MHz: 3000 FPU: Integrated CPU(s) enabled: 8 cores, 2 chips, 4 cores/chip CPU(s) orderable: 1,2 chips Primary Cache: 32 KB I + 32 KB D on chip per core Secondary Cache: 12 MB I+D on chip per chip, 6 MB shared / 2 cores L3 Cache: None Other Cache: None Memory: 16 GB (8 x 2 GB DDR2-667) Disk Subsystem: 1 x 160 GB 7200 RPM SATA Other Hardware: None	Operating System: SUSE Linux Enterprise Server 10 (x86_64) SP2, Kernel 2.6.16-60.0.21-smp Compiler: Intel C++ Compiler 11.0 for Linux Build 20080730 Package ID: i_cproc_b_11.0.074 Auto Parallel: No File System: ReiserFS System State: Run level 3 (multi-user) Base Pointers: 32-bit Peak Pointers: Not Applicable Other Software: Microquill SmartHeap V8.1 Binutils 2.19																												
Standard Performance Evaluation Corporation info@spec.org http://www.spec.org/																													
Page 1																													

SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro SuperServer 6025B-TR+ (Intel Xeon E5450, 3.00 GHz)

SPECint_rate_base2006 = 130

CPU2006 license: 3184

Test sponsor: Intel Co.

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Sep-2008

Software Availability: Jan-2009

Results Table

Benchmark	Base							Peak						
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
400.perlbench	8	496	158	<u>496</u>	<u>158</u>	499	157							
401.bzip2	8	723	107	741	104	<u>739</u>	<u>104</u>							
403.gcc	8	622	104	625	103	<u>624</u>	<u>103</u>							
429.mcf	8	681	107	<u>682</u>	<u>107</u>	682	107							
445.gobmk	8	<u>554</u>	<u>152</u>	554	152	554	151							
456.hmmr	8	492	152	<u>493</u>	<u>151</u>	493	151							
458.sjeng	8	<u>608</u>	<u>159</u>	608	159	608	159							
462.libquantum	8	959	173	955	174	<u>959</u>	<u>173</u>							
464.h264ref	8	673	263	674	263	<u>673</u>	<u>263</u>							
471.omnetpp	8	678	73.7	680	73.5	<u>679</u>	<u>73.7</u>							
473.astar	8	<u>681</u>	<u>82.4</u>	683	82.3	681	82.5							
483.salanckmk	8	422	131	423	131	<u>423</u>	<u>131</u>							

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Submit Notes

The config file option 'submit' was used.

Operating System Notes

'ulimit -s unlimited' was used to set the stack size to unlimited prior to run

Platform Notes

BIOS Settings:
 Hardware Prefetcher - Disabled (Default - Enabled)
 Adjacent Cache Line Prefetch - Disabled (Default - Enabled)

General Notes

taskset was used to bind processes to cores except for 462.libquantum peak

Base Compiler Invocation

C benchmarks:

icc

C++ benchmarks:

icpc

Standard Performance Evaluation Corporation
 info@spec.org
 http://www.spec.org/

Page 2

SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro SuperServer 6025B-TR+ (Intel Xeon E5450, 3.00 GHz)

SPECint_rate_base2006 = 130

CPU2006 license: 3184

Test sponsor: Intel Co.

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Sep-2008

Software Availability: Jan-2009

Base Portability Flags

400.perlbench: -DSPEC_CPU_LINUX_IA32
462.libquantum: -DSPEC_CPU_LINUX
483.xalanbmk: -DSPEC_CPU_LINUX

Base Optimization Flags

C benchmarks:
-xSSE4.1 -ipo -O3 -no-prec-div -static -inline-calloc
-opt-malloc-options-3 -opt-prefetch

C++ benchmarks:
-xSSE4.1 -ipo -O3 -no-prec-div -opt-prefetch -w1, -z, muldefs
-L/usr/cpu2006/SmartHeap_8.1/lib -lsmartheap

Base Other Flags

C benchmarks:
403.gcc: -Dalloca-_alloca

SPEC and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. All other brand and product names appearing in this result are trademarks or registered trademarks of their respective holders.

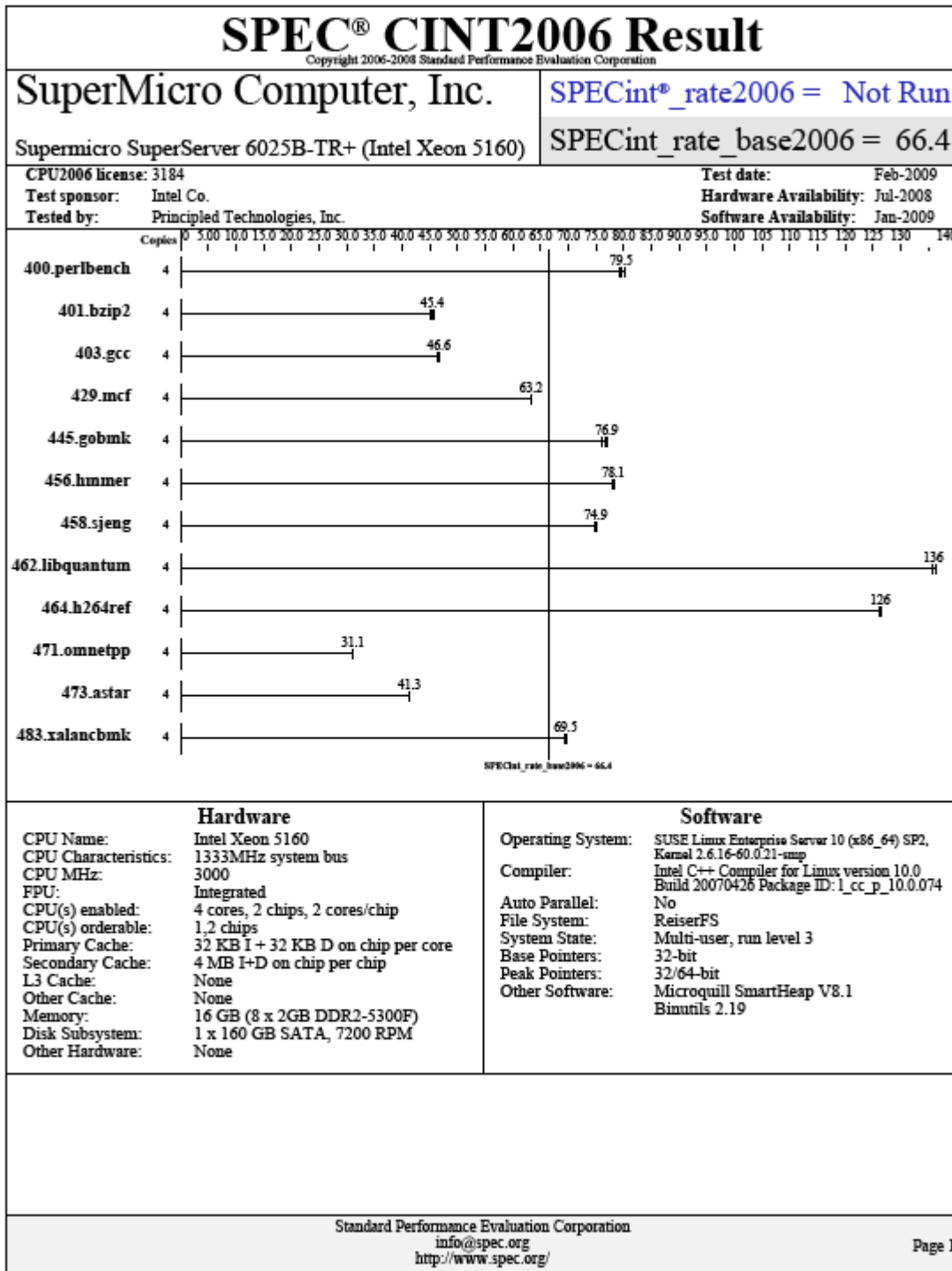
For questions about this result, please contact the tester.
For other inquiries, please contact webmaster@spec.org.

Tested with SPEC CPU2006 v1.1.
Report generated on Tue Mar 3 15:57:16 2009 by SPEC CPU2006 PS/PDF formatter v6128.

Standard Performance Evaluation Corporation
info@spec.org
<http://www.spec.org/>

Page 3

SUSE Linux Enterprise Server 10: Intel Xeon processor 5160-based server



SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

SuperMicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro SuperServer 6025B-TR+ (Intel Xeon 5160)

SPECint_rate_base2006 = 66.4

CPU2006 license: 3184

Test sponsor: Intel Co.

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Jul-2008

Software Availability: Jan-2009

Results Table

Benchmark	Base							Peak						
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
400.perlbench	4	488	80.1	493	79.3	<u>492</u>	<u>79.5</u>							
401.bzip2	4	<u>850</u>	<u>45.4</u>	846	45.6	854	45.2							
403.gcc	4	689	46.7	<u>691</u>	<u>46.6</u>	695	46.4							
429.mcf	4	<u>577</u>	<u>63.2</u>	576	63.3	577	63.2							
445.gobmk	4	546	76.9	<u>546</u>	<u>76.9</u>	551	76.2							
456.hmmer	4	478	78.1	<u>478</u>	<u>78.1</u>	477	78.2							
458.sjeng	4	645	75.0	646	74.9	<u>646</u>	<u>74.9</u>							
462.libquantum	4	607	137	<u>610</u>	<u>136</u>	611	136							
464.h264ref	4	700	126	<u>701</u>	<u>126</u>	701	126							
471.omnetpp	4	803	31.1	<u>805</u>	<u>31.1</u>	805	31.0							
473.astar	4	680	41.3	<u>680</u>	<u>41.3</u>	680	41.3							
483.xalanbmk	4	396	69.6	398	69.4	<u>397</u>	<u>69.5</u>							

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Submit Notes

The config file option 'submit' was used.

General Notes

taskset utility used to bind CPU(s) to processes

Base Compiler Invocation

C benchmarks:

icc

C++ benchmarks:

icpc

Base Portability Flags

400.perlbench: -DSPEC_CPU_LINUX IA32

462.libquantum: -DSPEC_CPU_LINUX

483.xalanbmk: -DSPEC_CPU_LINUX

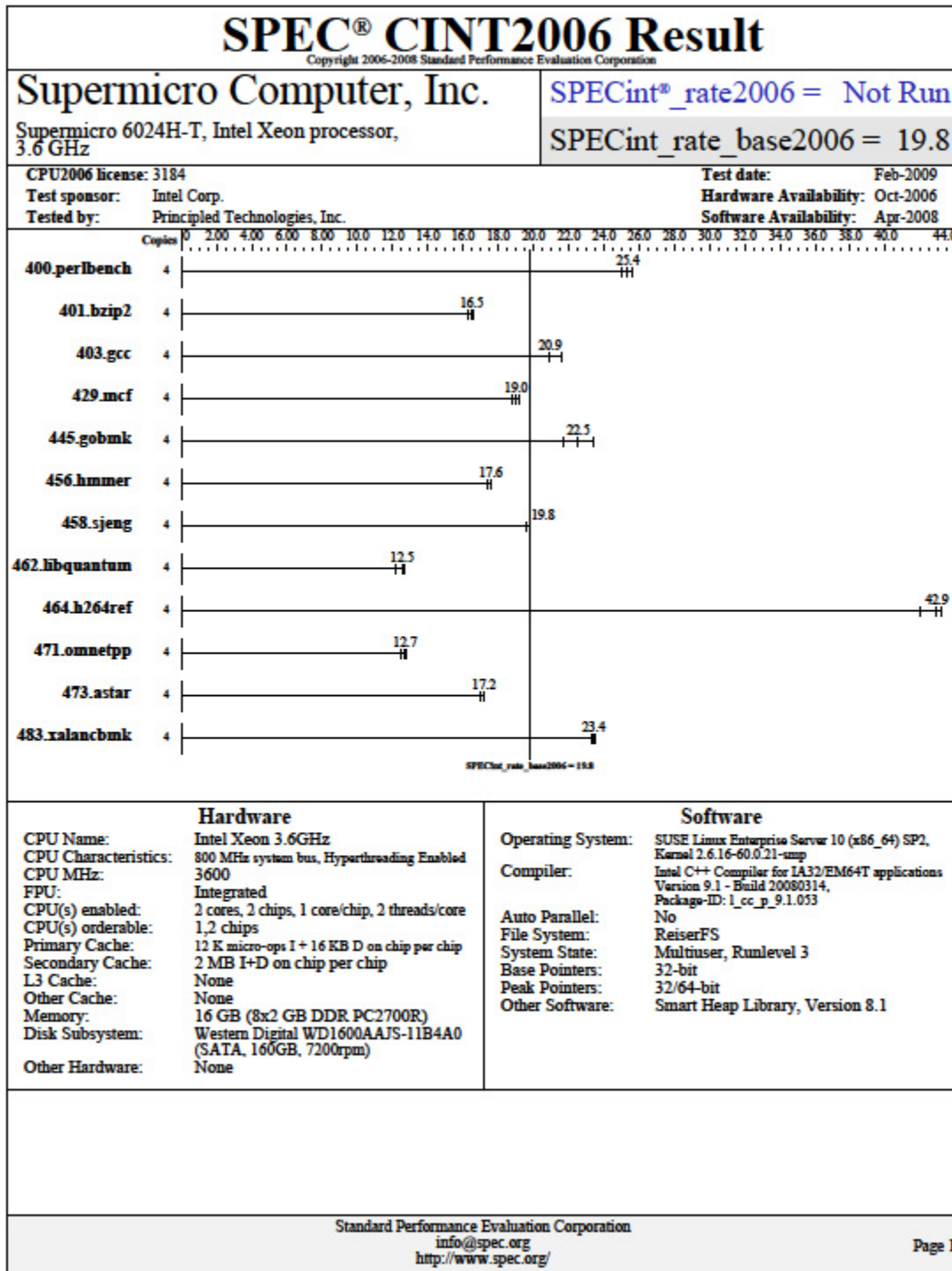
Standard Performance Evaluation Corporation

info@spec.org
http://www.spec.org/

Page 2

SPEC CINT2006 Result <small>Copyright 2006-2008 Standard Performance Evaluation Corporation</small>	
SuperMicro Computer, Inc.	SPECint_rate2006 = Not Run
Supermicro SuperServer 6025B-TR+ (Intel Xeon 5160)	SPECint_rate_base2006 = 66.4
CPU2006 license: 3184	Test date: Feb-2009
Test sponsor: Intel Co.	Hardware Availability: Jul-2008
Tested by: Principled Technologies, Inc.	Software Availability: Jan-2009
Base Optimization Flags	
<p>C benchmarks: -fast</p> <p>C++ benchmarks: -xT -ipo -O3 -no-prec-div -wL -z,muldefs -L/usr/cpu2006/SmartHeap_8.1/lib -lsmarheap</p>	
Base Other Flags	
<p>C benchmarks: 403.gcc: -Dalloca=_alloca</p>	
<p>SPEC and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. All other brand and product names appearing in this result are trademarks or registered trademarks of their respective holders.</p> <p>For questions about this result, please contact the tester. For other inquiries, please contact webmaster@spec.org.</p> <p>Tested with SPEC CPU2006 v1.1. Report generated on Tue Mar 3 20:37:23 2009 by SPEC CPU2006 PS/PDF formatter v6128.</p>	
Standard Performance Evaluation Corporation info@spec.org http://www.spec.org/	
Page 3	

SUSE Linux Enterprise Server 10: Intel Xeon processor 3.6 GHz-based server



SPEC CINT2006 Result

Copyright 2006-2008 Standard Performance Evaluation Corporation

Supermicro Computer, Inc.

SPECint_rate2006 = Not Run

Supermicro 6024H-T, Intel Xeon processor,
3.6 GHz

SPECint_rate_base2006 = 19.8

CPU2006 license: 3184

Test sponsor: Intel Corp.

Tested by: Principled Technologies, Inc.

Test date: Feb-2009

Hardware Availability: Oct-2006

Software Availability: Apr-2008

Results Table

Benchmark	Base								Peak							
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio		
400.perlbench	4	1562	25.0	1528	25.6	<u>1541</u>	<u>25.4</u>									
401.bzip2	4	2363	16.3	<u>2340</u>	<u>16.5</u>	2321	16.6									
403.gcc	4	<u>1538</u>	<u>20.9</u>	1542	20.9	1490	21.6									
429.mcf	4	1942	18.8	<u>1919</u>	<u>19.0</u>	1900	19.2									
445.gobmk	4	<u>1862</u>	<u>22.5</u>	1790	23.4	1935	21.7									
456.hmmer	4	2141	17.4	2123	17.6	<u>2124</u>	<u>17.6</u>									
458.sjeng	4	2468	19.6	2439	19.8	<u>2439</u>	<u>19.8</u>									
462.libquantum	4	6812	12.2	<u>6616</u>	<u>12.5</u>	6563	12.6									
464.h264ref	4	<u>2063</u>	<u>42.9</u>	2111	41.9	2048	43.2									
471.omnetpp	4	2000	12.5	<u>1969</u>	<u>12.7</u>	1958	12.8									
473.astar	4	1653	17.0	<u>1635</u>	<u>17.2</u>	1633	17.2									
483.salanckmk	4	1185	23.3	1173	23.5	<u>1178</u>	<u>23.4</u>									

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Submit Notes

The config file option 'submit' was used.

Operating System Notes

'ulimit -s unlimited' was used to set the stacksize to unlimited prior to run
'/usr/bin/taskset' used to bind processes to CPUs

General Notes

The system bus runs at 800 MHz

Base Compiler Invocation

C benchmarks:

icc

C++ benchmarks:

icpc

Standard Performance Evaluation Corporation

info@spec.org
http://www.spec.org/

Page 2

SPEC CINT2006 Result <small>Copyright 2006-2008 Standard Performance Evaluation Corporation</small>	
Supermicro Computer, Inc. Supermicro 6024H-T, Intel Xeon processor, 3.6 GHz	SPECint_rate2006 = Not Run SPECint_rate_base2006 = 19.8
CPU2006 license: 3184 Test sponsor: Intel Corp. Tested by: Principled Technologies, Inc.	Test date: Feb-2009 Hardware Availability: Oct-2006 Software Availability: Apr-2008
Base Portability Flags	
<pre> 400.perlbench: -DSPEC_CPU_LINUX_X64 462.libquantum: -DSPEC_CPU_LINUX 483.xalanbmk: -DSPEC_CPU_LINUX </pre>	
Base Optimization Flags	
C benchmarks: -fast C++ benchmarks: -xP -O3 -ipo -no-prec-div -L/usr/cpu2006/SmartHeap_0.1/lib -lsmarheap	
<p>SPEC and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. All other brand and product names appearing in this result are trademarks or registered trademarks of their respective holders.</p> <p>For questions about this result, please contact the tester. For other inquiries, please contact webmaster@spec.org.</p> <p>Tested with SPEC CPU2006 v1.1. Report generated on Mon Mar 9 17:13:06 2009 by SPEC CPU2006 PS/PDF formatter v6128.</p>	
Standard Performance Evaluation Corporation info@spec.org http://www.spec.org/	
Page 3	

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
Road, Suite 250 Durham, NC
27703 www.principledtechnologies.com [info@](mailto:info@principledtechnologies.com)
principledtechnologies.com

Principled Technologies is a registered trademark of Principled Technologies, Inc. Intel and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*All other product names are the trademarks of their respective owners.

Disclaimer of Warranties; Limitation of Liability: PRINCIPLED TECHNOLOGIES, INC. HAS MADE REASONABLE EFFORTS TO ENSURE THE ACCURACY AND VALIDITY OF ITS TESTING, HOWEVER, PRINCIPLED TECHNOLOGIES, INC. SPECIFICALLY DISCLAIMS ANY WARRANTY, EXPRESSED OR IMPLIED, RELATING TO THE TEST RESULTS AND ANALYSIS, THEIR ACCURACY, COMPLETENESS OR QUALITY, INCLUDING ANY IMPLIED WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE. ALL PERSONS OR ENTITIES RELYING ON THE RESULTS OF ANY TESTING DO SO AT THEIR OWN RISK, AND AGREE THAT PRINCIPLED TECHNOLOGIES, INC., ITS EMPLOYEES AND ITS SUBCONTRACTORS SHALL HAVE NO LIABILITY WHATSOEVER FROM ANY CLAIM OF LOSS OR DAMAGE ON ACCOUNT OF ANY ALLEGED ERROR OR DEFECT IN ANY TESTING PROCEDURE OR RESULT. IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC. BE LIABLE FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH ITS TESTING, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC.'S LIABILITY, INCLUDING FOR DIRECT DAMAGES, EXCEED THE AMOUNTS PAID IN CONNECTION WITH PRINCIPLED TECHNOLOGIES, INC.'S TESTING. CUSTOMER'S SOLE AND EXCLUSIVE REMEDIES ARE AS SET FORTH HEREIN.