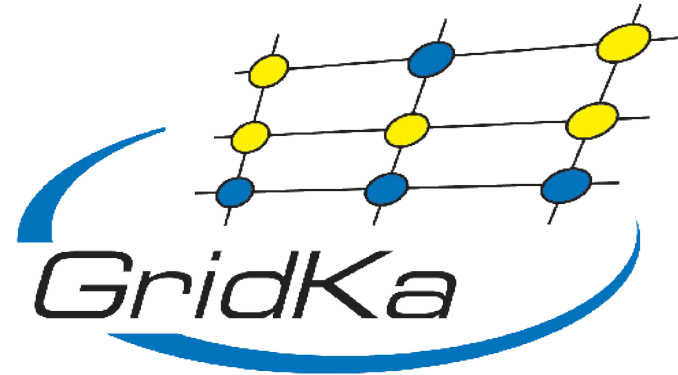


CPU Benchmarking at

– Update April 2007 –



Manfred Alef

Grid Computing Centre Karlsruhe (GridKa)

Forschungszentrum Karlsruhe
Institute for Scientific Computing
Hermann-von-Helmholtz-Platz 1
D-76344 Eggenstein-Leopoldshafen

<http://www.fzk.de>, <http://www.gridka.de>

firstname.lastname@iwr.fzk.de

Comparison of Measurements FZK/CERN:

- Measurements at FZK and CERN:
- Metric: SPECint_base2000
 - Operating system: Scientific Linux 3 or 4, i386
 - Compiler: gcc-3.4.x
 - Run 1 benchmark per core in parallel –
the total box performance is the sum of results
for each core

Comparison of Measurements FZK/CERN:

- Measurements at FZK (HEPiX at SLAC 2005):
 - Optimizing flags: **-O3 -funroll-loops -march**
+ a few portability flags *
("best" performance in GridKa environment)

* See the appendix of this talk for a listing of the portability flags

<http://www.slac.stanford.edu/conf/hepix05/talks/friday/alef.pdf>

Comparison of Measurements FZK/CERN:

- Measurements at CERN (HEPiX at JLab 2006):
 - Optimizing flags: **-O2 -pthread -fPIC**
+ a few portability flags *
(by LCG Architects Forum; used by LHC exp.)

* See the appendix of this talk for a listing of the portability flags

<https://indico.fnal.gov/materialDisplay.py?contribId=32&&sessionId=6&&materialId=slides&&confId=384>

Comparison of Measurements FZK/CERN:

→ Examples (SL4 **i386**, gcc-3.4.x):

	FZK	CERN*	diff.
Xeon 5160 (Woodcrest 3.0 GHz)	6979	5433	-22%
Opteron 2214 (2.21 GHz)	4471	2957	-34%
Xeon 2.4 GHz	1226	989	-19%

* Measurements by GridKa in order to compare benchmark results from the same hardware

Comparison of Measurements FZK/CERN:

→ What are the causes? (**i386**)

	X-5160	O-2214	X 2.4GHz
-O3 -funroll-loops -march	6979 $\pm 0\%$	4471 $\pm 0\%$	1226 $\pm 0\%$
-O3	6472 -7%	4007 -10%	1227 -0%
-O2	6361 -9%	3946 -12%	1135 -7%
-O3 -pthread	6504 -7%	4050 -9%	1126 -8%
-O2 -pthread	6328 -9%	3954 -12%	1135 -7%

Comparison of Measurements FZK/CERN:

→ What are the causes? (**i386**)

	X-5160	O-2214	X 2.4GHz
-O3 -funroll-loops -march	6979 $\pm 0\%$	4471 $\pm 0\%$	1226 $\pm 0\%$
-O3	6472 -7%	4007 -10%	1227 -0%
-O2	6361 -9%	3946 -12%	1135 -7%
-O3 -pthread	6504 -7%	4050 -9%	1126 -8%
-O2 -pthread	6328 -9%	3954 -12%	1135 -7%
-O3 -fPIC	5459 -22%	3000 -33%	981 -20%
-O2 -fPIC	5447 -22%	2951 -34%	984 -20%
-O2 -pthread -fPIC	5433 -22%	2957 -34%	989 -19%

Comparison of Measurements FZK/CERN:

→ What are the causes?

The -fPIC flag causes the major part of about 13% to 21% loss of performance *[when used on a i386 (32bit) system]*!

(The -fPIC – "position-independent code" – option is used to enable dynamic linking.)

First 64bit Results:

→ Comparing 32/64bit (SL4 **x86_64**, gcc-3.4.x):

	FZK	CERN	diff.
Xeon X5355 (Clovertown 2.66 GHz)	11307	10577	-6%
Xeon 5160 (Woodcrest 3.00 GHz)	6376	5833	-9%
Xeon 5148 (Woodcrest 2.33 GHz)	5674	5319	-6%
Opteron 2214 (2.21 GHz)	4388	3861	-12%
Opteron 270 (2.0 GHz)	4222	3762	-11%

First 64bit Results:

→ Comparing 32bit/64bit (**i386** / **x86_64**):

	X-5160		O-2214	
-O3 -funroll-loops -march	6979	6376	4471	4388
-O3	6472	6172	4007	4318
-O2	6361	6056	3946	4337
-O3 -pthread	6504	6120	4050	4285
-O2 -pthread	6328	6146	3954	4337
-O3 -fPIC	5459	5765	3000	3962
-O2 -fPIC	5447	5753	2951	3952
-O2 -pthread -fPIC	5433	5833	2957	3861

First 64bit Results:

→ Comparing 32bit/64bit (**i386** / **x86_64**):

	X-5160			O-2214		
-O3 -funroll-loops -march	6979	6376	±0%	4471	4388	±0%
-O3	6472	6172	-3%	4007	4318	-2%
-O2	6361	6056	-5%	3946	4337	-1%
-O3 -pthread	6504	6120	-4%	4050	4285	-2%
-O2 -pthread	6328	6146	-4%	3954	4337	-1%
-O3 -fPIC	5459	5765	-10%	3000	3962	-10%
-O2 -fPIC	5447	5753	-10%	2951	3952	-10%
-O2 -pthread -fPIC	5433	5833	-9%	2957	3861	-12%

Multi-Core Performance:

- How does the performance scale with the number of cores per system?

FZK	# cores	i386			x86_64		
		1	av.	diff.	1	av.	diff.
Xeon 2.40 GHz	2	688	613	-11%	—	—	
Xeon 5160	4	1995	1745	-13%	1784	1594	-11%
Opteron 2214	4	1155	1118	-3%	1072	1097	+1%
Xeon X5355	8				1883	1413	-25%
Barcelona	8	?	?	?	?	?	?

1: 1 benchmark run per box -O3 -funroll-loops -march

av.: average performance per core

Multi-Core Performance:

- How does the performance scale with the number of cores per system?

CERN	# cores	i386			x86_64		
		1	av.	diff.	1	av.	diff.
Xeon 2.40 GHz	2	541	495	-9%	—	—	
Xeon 5160	4	1489	1358	-9%	1538	1458	-5%
Opteron 2214	4	745	739	-1%	1029	965	-6%
Xeon X5355	8				1730	1322	-24%
Barcelona	8	?	?	?	?	?	?

1: 1 benchmark run per box -O2 -pthread -fPIC

av.: average performance per core

First SPEC CPU2006 Results:



SPEC CPU2000 i386				FZK				CERN			
				1	av.	rate		1	av.	rate	
Xeon 5160				1995	1745	-		1489	1358	-	
Opteron 2214				1155	1118	-		745	739	-	
SPEC CPU2006 i386				FZK				CERN			
				1	av.	rate		1	av.	rate	
Xeon 5160				13.4	10.4	10.1		11.9	9.4	9.2	
Opteron 2214				8.2	7.7	7.5		6.6	6.2	6.1	

1: 1 benchmark run per box

av.: average performance per core

rate: SPECint_rate_base2006 metric (#copies = #cores), per core

More details:

➔ <http://hepix.caspur.it/afs/hepix.org/project/ptrack/>

Comments, Questions?

Appendix: Portability Flags Used in SPEC CPU2000 Benchmark Runs (1):

→ Optimizing flags: -O3 -funroll-loops -march (i386 only):

186.crafty:

CPORTABILITY = -DLINUX_i386

252.eon:

CXXPORTABILITY = -DHAS_ERRLIST

253.perlbmk:

CPORTABILITY = -DSPEC_CPU2000_NEED_BOOL -DSPEC_CPU2000_LINUX_I386

254.gap:

CPORTABILITY = -DSYS_HAS_SIGNAL_PROTO -DSYS_HAS_MALLOC_PROTO
-DSYS_HAS_CALLOC_PROTO -DSYS_IS_USG -DSYS_HAS_IOCTL_PROTO
-DSYS_HAS_TIME_PROTO -DSYS_HAS_READ_PROTO -DSYS_HAS_WRITE_PROTO

Appendix: Portability Flags Used in SPEC CPU2000 Benchmark Runs (2):

- All other sets of optimizing flags, e.g. CERN's settings (i386 only), from <http://it-div-procurements.web.cern.ch/it-div-procurements/IT-3417/spec/cern.cfg>:

186.crafty:

CPORTABILITY = -DLINUX_i386

252.eon:

CXXPORTABILITY = -DHAS_ERRLIST -ffloat-store

253.perlbmk:

CPORTABILITY = -DSPEC_CPU2000_NEED_BOOL -DSPEC_CPU2000_LINUX_I386

254.gap:

CPORTABILITY = -DSYS_HAS_SIGNAL_PROTO -DSYS_HAS_MALLOC_PROTO

-DSYS_HAS_CALLOC_PROTO -DSYS_IS_USG -DSYS_HAS_IOCTL_PROTO

-DSYS_HAS_TIME_PROTO -DSYS_HAS_READ_PROTO -DSYS_HAS_WRITE_PROTO

int:

PORTABILITY = -DSPEC_CPU2000_LP64 -DSYS_HAS_ANSI *

- * The general integer PORTABILITY flag has caused some trouble, e.g. on systems running SL4. Some benchmarks have been started without these defines.

Appendix: Portability Flags Used in SPEC CPU2000 Benchmark Runs (3):

→ Measurements on x86_64:

186.crafty:

CPORTABILITY = -DLINUX_i386

252.eon:

CXXPORTABILITY = -DHAS_ERRLIST -DSPEC_CPU2000_LP64

253.perlbmk:

CPORTABILITY = -DSPEC_CPU2000_NEED_BOOL -DSPEC_CPU2000_LINUX_I386
-DSPEC_CPU2000_LP64

254.gap:

CPORTABILITY = -DSYS_HAS_SIGNAL_PROTO -DSYS_HAS_MALLOC_PROTO
-DSYS_HAS_CALLOC_PROTO -DSYS_IS_USG -DSYS_HAS_IOCTL_PROTO
-DSYS_HAS_TIME_PROTO -DSYS_HAS_READ_PROTO -DSPEC_CPU2000_LP64

255.vortex:

CPORTABILITY = -DSPEC_CPU2000_LP64