# 64-BIT OPTERON SYSTEMS IN HIGH ENERGY AND ASTROPARTICLE PHYSICS

P. Wegner*, S. Wiesand#, DESY, 15738 Zeuthen, Germany

*Abstract*

64-Bit commodity clusters and farms based on AMD technology meanwhile have been proven to achieve a high computing power in many scientific applications. This report first gives a short introduction into the specialities of the AMD64 architecture and the characteristics of two-way Opteron systems. Results from measuring the performance and the behaviour of such systems in various Particle Physics applications as compared to the classical 32-Bit systems, as well as the recently launched EM64T systems from Intel, are presented. In addition, the compatibility of 32- and 64-Bit architectures and Linux operating system issues, as well as the impact on fabric management are discussed.

## INTRODUCTION TO AMD64/EM64T

64-bit systems are neither new nor revolutionary. Systems based on the HP-PA, SPARC, MIPS, Alpha, POWER and Itanium CPUs have been available and used in High Energy Physics for a decade at least. The Linux operating system has been available for these platforms for several years as well. What these architectures have in common is that they adhere to the RISC paradigm, and they can't execute x86 instructions. While x86 software emulation is available for the Alpha and Itanium CPUs, this solution by its nature is not very performant.

The 64-bit CPUs implementing AMD64 or EM64T (AMD's Opteron and Athlon64, as well as, more recently, Intel's latest Pentium 4 and Xeon models) however represent a true extension of the x86 CISC architecture: They can execute x86 instructions in hardware at full speed, thus providing a smooth path for migrating the plethora of existing HEP applications for the 32-bit x86 platform to the 64-bit world.

### *Execution Modes*

These CPUs can operate either in *legacy mode*, in which they present themselves as traditional x86 CPUs only allowing for a 32-bit operating system and applications and 4 GB of address space, or in *long mode*, requiring a 64-bit operating system and providing an extended address space while allowing to run both 32-bit and 64-bit applications.

### *Performance Enhancements in 64-bit mode*

Besides an extended address range - which at present is not full 64 bits but only 40 bits (1 TB) of physical memory and 48 bits (256 TB) of virtual memory - in long mode these CPUs provide a number of features that improve performance as compared to classical x86 CPUs:

- Twice the number (16 instead of 8) of SSE registers, with the same width of 128 bits as before.
- Twice the number (16 instead of 8) of general purpose registers, now 64 bits wide instead of just 32. The paucity of general purpose registers has always been a weakness of the x86 architecture.
- Additional addressing modes, among them a generally usable PC-relative addressing that reduces the penalty for position independent code, as it has to be used for shared libraries, from about 20% to about 8% [1].

The x87 FPU registers are still available, but in long mode must not be used by 64-bit applications since their content is not preserved across context switches in 64-bit mode.

### *Differences between AMD64 and EM64T*

These architectures are very similar and designed to be compatible, but some differences do exist:

- While both implement the SSE2 instructions, only AMD's CPUs implement 3dNow!, and only Intel's CPUs implement SSE3, which can be of advantage in complex number calculations.
- There are some more subtle differences in the instruction sets. These should not matter for authors of ordinary application software programmes.
- AMD's CPUs have an on chip memory interface, and memory can be attached to each CPU in a system. This means that the total memory bandwidth scales with the number of CPUs in a system. On the other hand, Intel's Xeon CPUs still have to compete for a single front side bus, and all memory traffic has to pass through the front side bus, the memory controller hub, and the data path between the MCH and the RAM. At present, the front side bus, the memory interfaces, and the Hypertransport [2] interconnect between AMD Opteron CPUs all have the same bandwidth of 6.4 GB/s.

### *NUMA (Non Uniform Memory Access)*

This latter feature is an advantage for Opteron based SMP systems over SMP systems built from Intel CPUs, and it does show in our performance comparisons for physics applications. For best efficiency however, it requires the operating system kernel to allocate memory as close as possible to the requesting process, and to

*Peter.Wegner@desy.de
#Stephan.Wiesand@desy.de

schedule processes as close as possible to their resident memory set, since access to non-local memory is possible with full bandwidth, but at increased latency. Recent Linux kernels provide this NUMA support, but not all Linux distributions ship a kernel with this feature yet.

## PERFORMANCE COMPARISONS

### *Hardware*

For our performance comparisons, we used the following hardware. Each system was equipped with two CPUs and an SCSI disk spinning at 10,000 RPM:

- An IBM e-server 325, equipped with 4 GB of RAM and AMD Opteron 246 CPUs. In the charts, this system appears with the label *Opteron 2.0 GHz*.
- A SUN Fire V20z, equipped with 4GB of RAM and Opteron 248 CPUs. In the charts, this system appears with the label *Opteron 2.2 GHz*.
- A Supermicro 7044H-X6R, equipped with 4 GB of RAM and Xeon 3.4 GHz CPUs. In the charts, this EM64T-capable system appears with the label *Xeon 3.4 GHz*. It should be noted that this was an early system of its kind and later production systems may perform slightly superior to the one we had access to.

The raw memory performance, as measured with the hdparm command under Linux (result for buffer-cache reads) of these systems was approximately the same, with the Xeon system being slightly faster than both Opteron systems.

For reference, we used these 32-bit systems:

- A SUN Fire V65x, equipped with 2 GB of RAM and Xeon 3.2 GHz CPUs. In the charts, this system appears with the label *Xeon 3.2 GHz*.
- A Supermicro 6023H, equipped with 1 GB of RAM and Pentium III CPUs clocked at 1.266 GHz. In the charts, this system appears with the label *Tualatin 1.266 GHz*.

### *Operating systems*

Both Opteron systems were running SuSE Linux 9.0 professional, both 32-bit systems SuSE Linux 8.2 professional. The EM64T system was installed with SuSE 9.1 professional, the only operating system used with a 2.6 kernel. Except for the EM64T system, all were running a full DESY production environment, including an AFS client.

### *ROOT stress benchmark*

We subjected the faster Opteron system, the EM64T system and the faster 32-bit system to the stress test suite coming with the ROOT framework [3]. On each system, we first ran a single process of this kind, then two of them at the same time.

The results of this comparison are shown in figure 1. Both 64-bit systems clearly excel, but the Opteron system is slightly faster. More importantly, loading the second Opteron CPU on a system very nearly doubles the total performance, while the Xeon system seems to be throttled by some kind of bottleneck. Most likely, this is due to the CPUs competing for memory access.

### *Sieglinde Performance*

Sieglinde is the Amanda [4] experiment's neutrino reconstruction and filtering software, written in C++ and using the ROOT framework. Amanda has developed a benchmark for this application, kindly made available to us by P. Nießen, University of Delaware, that processes a standardized set of input data and stores intermediate data in a MySQL database running on the same node. A native AMD64 build of the MySQL server software (version 4)

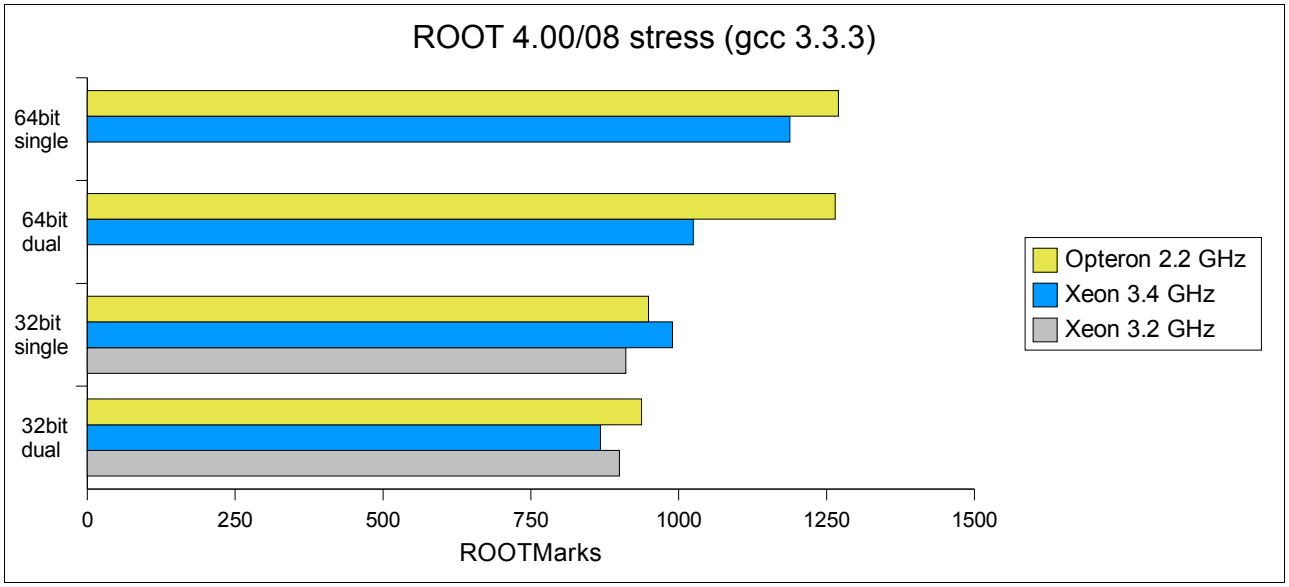


Figure 1: Results from the ROOT stress benchmark. 64-bit results are clearly superior to the 32-bit ones, even if the CPU clock ratio for the Xeons is taken into account. On the Opteron system, loading both CPUs yields almost twice the performance. This is not the case on the EM64T system, presumably due to a memory bottleneck.

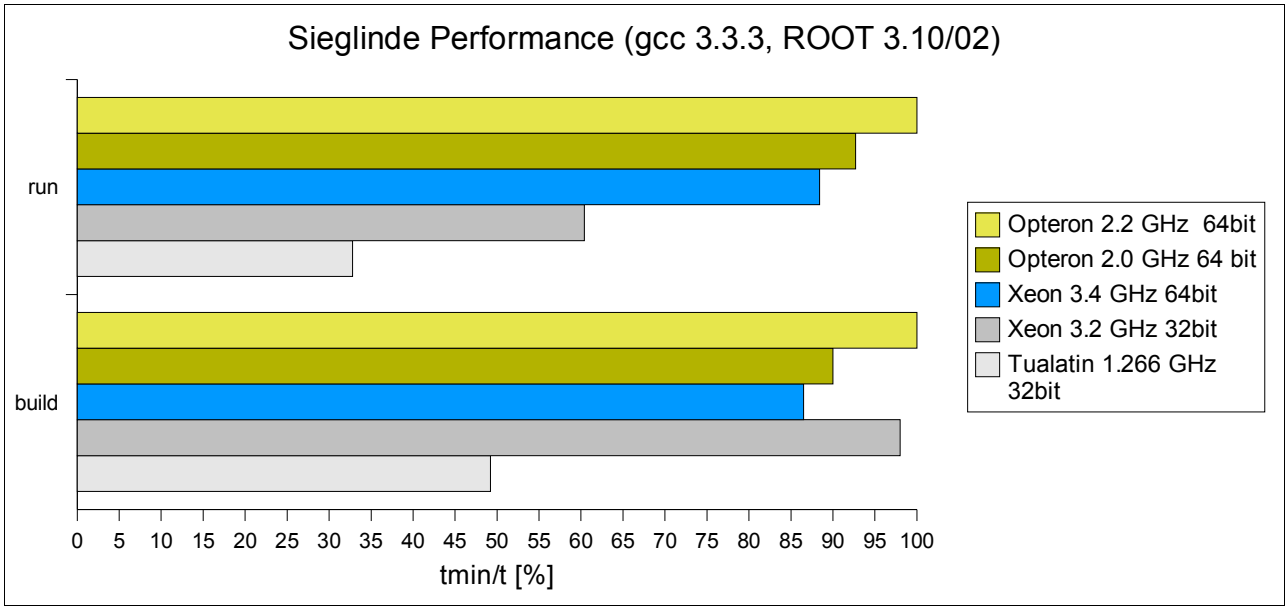


Figure 2: Performance of Sieglinde, the neutrino reconstruction and filtering software used by the Amanda experiment. This ROOT-based C++ application clearly favours the 64-bit systems. The bottom set of bars shows the build time performance, relevant for the length of development turnaround times, and illustrates that 64-bit builds are more expensive than 32-bit compilations. Recent Opteron system however catch up with the fastest 32-bit Xeon systems in this respect.

is available and was used where appropriate. Figure 2 shows that in runtime performance, again even the fastest 32-bit systems cannot compete with the new 64-bit systems. Again, a 2.2 GHz Opteron is slightly faster than the 3.4 GHz Xeon. Sieglinde also demonstrates that 64-bit compilations take considerably more CPU time than 32-bit ones: While building this application takes about ten minutes on our Pentium III test system, it's about half that time on both the 3.2 GHz Xeon and a 2.2 GHz Opteron, although the latter one performs much better according to the other data.

### *Pythia Performance*

This FORTRAN77 event generator [5] also performs better when used on the Opteron in 64-bit mode. Figure 3 also shows that for this application, Intel's FORTRAN compiler creates significantly faster code than g77 on the 32-bit Xeon platform. The commercial compilers

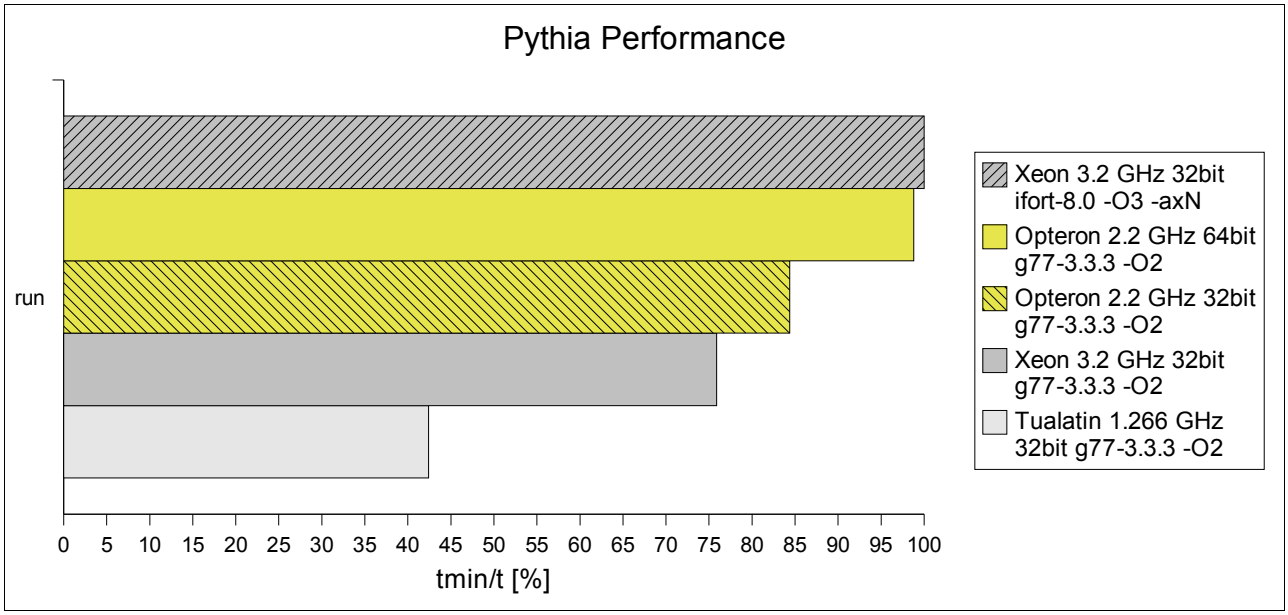


Figure 3: Performance of Pythia example 4, *Study of W mass shift at LEP2*. The Opteron system is significantly faster if used in 64bit mode. Substantial performance gains are possible on the 32-bit Xeon system by using Intel's compiler instead of gcc. This executable refused to run on the Opteron.

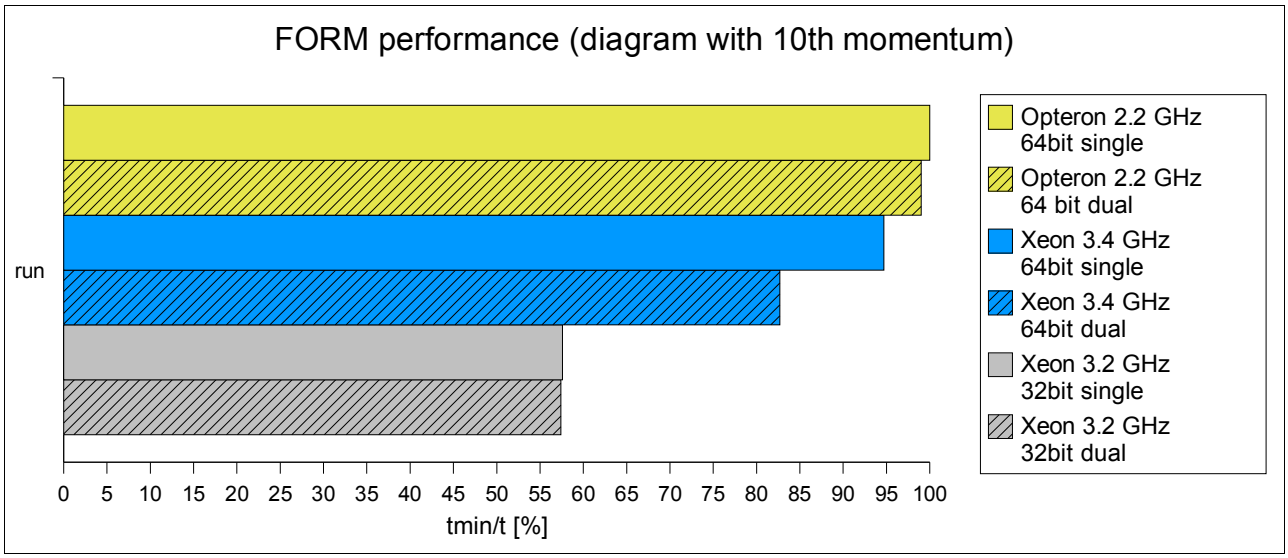


Figure 4: Performance of FORM, an application used in theoretical physics for symbolic formula manipulation to sum QCD diagrams. The 32-bit binary was built with Intel's C compiler, the 64-bit executable with gcc 3.3. FORM performs some I/O and extensively exercises the memory bandwidth.

available [6] from The Portland Group and Pathscale will have to be used on the Opteron for a fair comparison.

### *FORM Performance*

This symbolic formula manipulation software [7] tends to use large amounts of memory and hence exercises the memory bandwidth. As shown in figure 4, it performs much better on the 64-bit systems, presumably mostly due to their superior memory speed. Again, the Opteron architecture scales better than EM64T if both CPUs on an SMP system are used. The 64-bit executable was built by the author J. Vermaseren on one of our Opteron test systems (the source code is not available to the public).

### *Other Applications*

MPI based Lattice QCD calculations carried out by C. Urbach, FU Berlin, on Opteron and Xeon clusters [8] confirm our observations about the characteristics of Opteron SMP systems: The efficiency of dual Opteron CPU systems was about 90% while that of dual Xeon systems was about 60% only.

## USING LINUX ON AMD64/EM64T

### *Installation and Administration*

This turned out to be surprisingly simple. These systems behave the same way PC-like systems have always done. Boot loaders and installation methods are exactly the same as for 32-bit Linux. Technical staff does not have to acquire any new skills. Several mature Linux distributions are available, although not all of them support NUMA and have an extensive set of 32-bit compatibility packages yet. We expect this to improve quickly.

### *Porting Issues*

Problems were only encountered where source code assumes that the data types int, long, and pointer have the same size. While this is true on 32-bit Linux, the latter two are 64 bits wide instead of 32 as under a 64-bit operating system. This affects some libraries popular in High Energy Physics, most notably *cernlib*, which is still in use by the HERA experiments and many others but is no longer supported and will not be ported to any new platform by the former maintainers.

Since 64-bit applications must not use the x87 FPU registers, all floating point arithmetics has to be carried out using the SSE registers. This means that the 80-bit extended precision for intermediates, which is the default on x86 systems, is not available. Instead, standard IEEE 754 floating point precision applies to double precision calculations. This may be a problem for some software that relied on the extra bits for the mantissa in the past.

## REFERENCES

[1] J. Hubička, "Porting GCC to the AMD64 architecture", http://www.ucw.cz/~hubicka/papers/amd64.pdf.
[2] http://www.hypertransport.org.
[3] http://root.cern.ch.
[4] E. Andrés et al.: "Observation of high-energy neutrinos using Čerenkov detectors embedded deep in Antarctic ice", Nature 410 (2001) 441.
[5] T. Sjöstrand, P. Edén, C. Friberg, L. Lönnblad, G. Miu, S. Mrenna and E. Norrbin, Computer Phys. Commun. 135 (2001) 238, hep-ph/0010017.
[6] http://www.pgroup.com; http://www.pathscale.com.
[7] J.A.M. Vermaseren, "New features of FORM", math-ph/0010025.
[8] A. Heiss, "Infiniband for High Energy Physics, these proceedings".