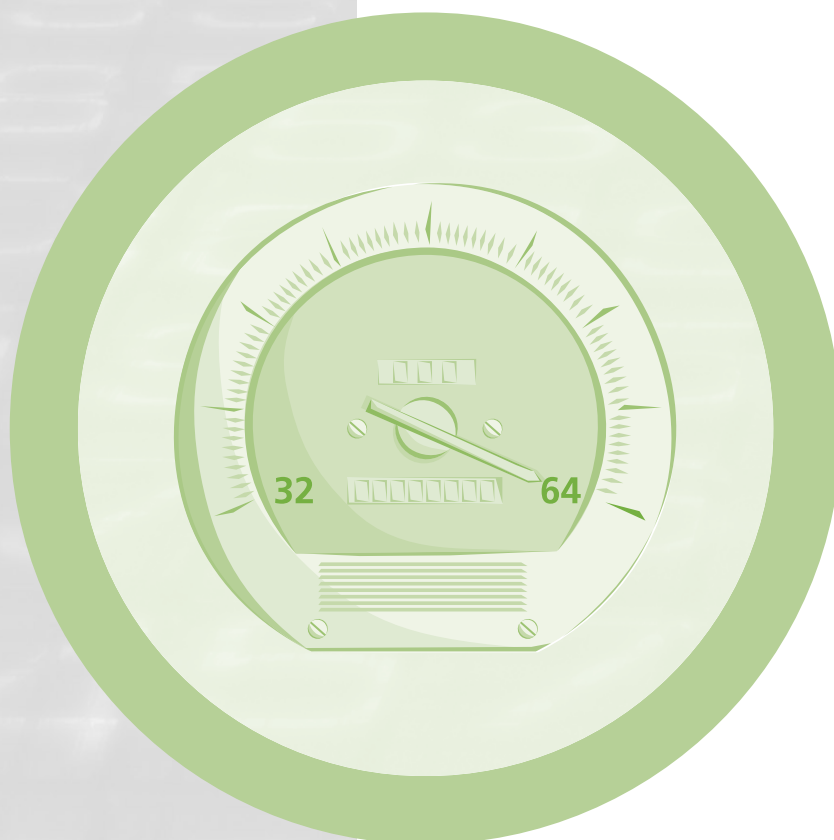


# SUSE

## LINUX ENTERPRISE SERVER 8 FOR AMD64



# SUSE LINUX ENTERPRISE SERVER 8 FOR AMD64

<b>INTRODUCTION</b>	.....	3
<b>AMD'S 64-BIT SYSTEMS</b>	.....	3
<b>LINUX ON AMD64 – SOME FUNDAMENTAL CHANGES REQUIRED</b>	.....	3
<b>SUSE LINUX ON AMD64</b>	Support for 32-bit Applications.....	4
	Large Memory Address space .....	5
	Large Files .....	6
	Selected AMD64 Optimizations for SuSE Linux Enterprise Server 8.....	6
	Optimized Routines.....	6
	System Calls are Expensive.....	6
	SMP vs. NUMA.....	6
	Shared Libraries .....	7
<b>CONCLUSION</b>	.....	7

## INTRODUCTION

AMD64 ushers in a new era for high performance and IT computing by extending the x86 instruction set to support 64-bit computing and allowing 32-bit and 64-bit applications to run – all without performance degradation. However, to be successful, the AMD64 architecture's scalability and its 64-bit enhancements requires an operating system that will exploit all new features while overcoming 32-bit limitations.

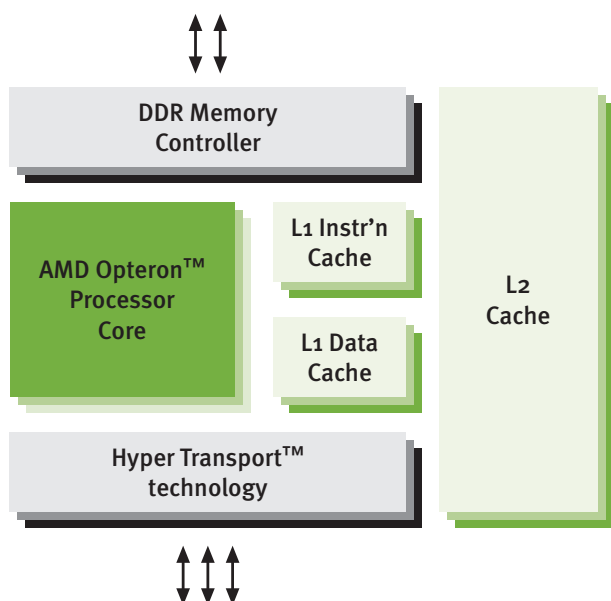
SuSE Linux AG worked hand-in-hand with AMD and the Linux community to develop SuSE Linux Enterprise Server 8 for AMD64. This paper gives a short introduction into the AMD64 processor family and the AMD64 specific implementation of SuSE Linux Enterprise Server.

## AMD'S 64-BIT SYSTEMS

In April of 2003 AMD released its first 64-bit processor, the AMD Opteron™, which is part of the AMD64 processor family. Around this new processor AMD has built a complete system architecture with many features that address high-end IT and high performance computing needs. Some of the key features of this architecture are:

- Allows native execution of existing 32-bit x86 programs.
- Executes native 64-bit AMD64 programs.
- Overcomes the limits of 32-bit memory addressing and allows access to 2<sup>40</sup> bits (1 Terabyte) of physical address space.
- 8 additional general purpose registers in 64-bit mode – 16 integer registers in all.

### AMD Opteron™ Processor Architecture



- Support for MMX, SSE, SSE2 and 3DNow!™ media instructions. In 64-bit mode, 8 additional registers are available so that SSE and SSE2 instructions can use up to 16 registers.
- A new instruction-pointer relative addressing mode
- A memory controller integrated into the processor for faster memory access. This eliminates the high latency memory subsystems typical with other types of processors.
- HyperTransport™ Technology, a high-performance point-to-point interconnect between CPUs and external devices. The HyperTransport is currently the fastest on-chip bus in production. Together with the per CPU memory controllers, it allows a highly scalable memory hierarchy.
- PCI-X bus for external devices with a speed of up to 133-MHz using the AMD 8131™ chipset. This provides high-speed bandwidth for networking and storage devices.
- Designed to be used in both multi-processor and a single-processor systems.

## LINUX ON AMD64 – SOME FUNDAMENTAL CHANGES REQUIRED

Linux is the favorite operating system for mid-range IT systems and for the high performance computing market. SuSE has been working with AMD since the summer of 2002 to create a version of Linux that runs on the AMD64 system architecture. Fortunately, SuSE has long been a leader in 64-bit Linux. We marshaled our best minds to take our expertise with large-scale Linux systems, and apply it to AMD64. Close cooperation between SuSE and AMD was critical. The AMD64 architecture had many new opportunities, but also presented new trade-offs in order to exploit those opportunities. With each design decision, SuSE and AMD collaborated to choose the option that was simultaneously stable and efficient.

Some of the specific changes made to the Linux system – and incorporated into the official sources of the respective projects – include:

- **A complete toolchain to generate and debug 64-bit AMD64 programs:**  
A basic element of every architecture is the toolchain to generate executable programs. For this the GNU Compiler Collection (GCC) and the GNU binutils were enhanced to generate 64-bit AMD64 programs. GCC currently supports the C, C++, Fortran77, Java, Ada95, and Objective-C languages. The two most important applications of the GNU binutils are assembler and linker.

SuSE also added general optimizations, like profile driven feedback optimizations, and code generation optimizations specific to AMD64. These were most often optimized through scheduling and instruction selection. Additionally debugging tools like the GNU Debugger (gdb) and strace were ported to AMD64.

- **GNU C Library changes for AMD64 processors:** The GNU C Library needed to be enhanced to support the AMD64 architecture. This included porting of architecture dependent files within the dynamic linker, development of code to interact with the Linux kernel (system calls), and development of thread support for AMD64. SuSE also included revised files that can be used for both AMD64 and 32-bit x86 development – a key requirement for migration of existing application servers.
- **Linux kernel enhancements to fully support AMD's new architecture:** The Linux kernel was enhanced to handle the AMD64 architecture. SuSE's engineers ported the 32-bit Linux code to 64-bit AMD64 and optimized it for the new hardware. Thus the SuSE/AMD64 kernel is fully native 64-bit. This allows it to access more virtual memory (up to 0.5 TB per process, 128 TB per system) than in 32-bit kernels.

Common drivers were ported to 64-bit mode and support for new AMD64 chipsets from AMD and its partners were added as well. An 32-bit emulation layer was added to allow 32-bit applications to execute seamlessly on the 64-bit kernel without code changes or even recompilation (install your binaries and go).

- **Graphical X11 interface (XFree86 project) for AMD64:** The XFree86 system consists of the X11 server, hardware drivers, the X11 libraries and a number of client applications like xterm. SuSE ported these components to AMD64 and also enhanced the X11 module loader to load 64-bit modules. The popular Gnome and KDE desktops were ported to give the user a familiar graphical desktop.

## SUSE LINUX ON AMD64

After SuSE Labs developers laid the groundwork of core Linux components on AMD64, SuSE developed a complete distribution for those systems. This distribution – SuSE Linux Enterprise Server 8 – takes advantage of the key features that are new in AMD Opteron.



SuSE Linux Enterprise Server 8 (powered by United-Linux 1.0) comes with over 700 packages to create a complete IT ready computing infrastructure. Nearly all of these applications were ported for AMD64 and include familiar components like Apache, Samba, and MySQL. The only 32-bit applications left on SuSE Linux Enterprise Server 8 are 32-bit closed-source versions like Acrobat Reader and IBM Java.

The changes to enable programs for AMD64 included changes to force recognition of the new architecture, general code cleanup for portability and changes to automatically link programs against the 64-bit libraries.

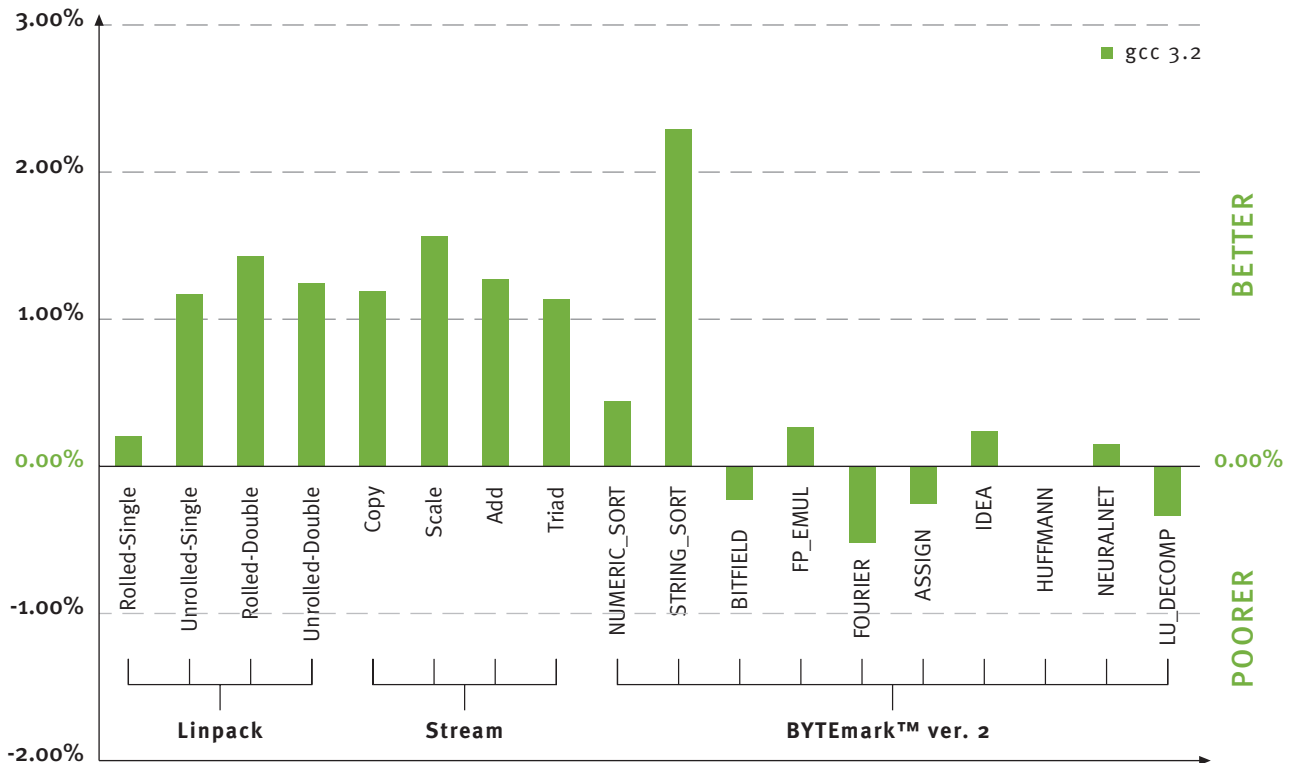
### Support for 32-bit Applications

To ease migration of existing 32-bit x86 systems, SuSE created a 64-bit kernel that can execute both 32-bit x86 and 64-bit AMD64 applications. The distribution comes with a number of 32-bit libraries enabling execution of 32-bit x86 programs without slowing performance. It is possible to completely replace a 32-bit server with a 64-bit server without changing applications.

The existing 32-bit programs will continue to run with the exception of certain system-level tools, like LVM tools, that need to operate in 64-bit mode. For almost any 32-bit application, the application's performance under the emulation layer matches or slightly exceeds its performance under a comparable 32-bit Linux kernel on the same AMD64 system.

To achieve this balance between 64-bit and 32-bit functionality, the 32-bit libraries are placed in the familiar /lib Linux directories without any changes. 64-bit libraries files will reside in /lib64, as is done on other Linux 64-bit architectures like zSeries, Sparc64 and PowerPC64.

**% Speed-up of 32-bit App on SuSE Linux Enterprise Server 8 for AMD64 relative to same 32-bit App on Linux 32 (all measured on Uniprocessor AMD Opteron™ system)**



All run-times measured on Uniprocessor AMD Opteron™ system

Dual operating modes are a critical element for IT organization. Users have the option of updating applications to 64-bit and move beyond 32-bit limitations, but at their leisure. Many IT organizations have third party applications that vendors have not yet ported to SuSE Linux Enterprise Server 8 for AMD64. With the support for execution of both 32- and 64-bit applications, users have the ability to move existing servers forward without any loss of functionality.

Due to architectural improvements in AMD Opteron, 32-bit x86 programs will run faster on AMD Opteron than on an AMD Athlon system with the same clock speed. If these 32-bit programs are written for portability, a simple recompilation of the applications will result in measurable throughput improvements since a 64-bit program is in generally faster due to:

- Eight extra registers in 64-bit mode that can be used to the toolchain's advantage.
- Improved calling conventions where function arguments are passed in registers and not on the stack.
- Fewer architectural instructions for the same source code. The code size of recompiled programs tends to stay within 10% of 32-bit x86 code. The main reason for the larger code is the longer instructions. The number of instructions itself decreases by

about 10%. Fewer instructions require less time to process and therefore generate faster programs.

- Usage of the SSE2 unit for floating point arithmetic (for C-language types float and double) with a flat 16 register file instead of the x87 FPU with its cumbersome register stack.
- Programs can use a greater address space and thus reduce I/O operations (see below).

The development toolchain was designed to let programmers generate both 32-bit and 64-bit code within the same environment. The default mode of operation of the GCC compiler is to produce 64-bit code. A compiler switch (-m32) enables generation of 32-bit x86 code. Developers therefore do not need an additional machine for development for 32-bit applications – they can instead switch to the new architecture and develop for x86 and AMD64 on the same computer.

**Large Memory Address space**

A 32-bit program running on an x86 system can only address 4 GB of memory. Under a 32-bit kernel the available address space is at most 2 - 3 GB (3.5 GB with a special kernel and static linking of an application) since the kernel also needs some of that memory.

However, the same 32-bit x86 application running under a 64-bit AMD64 kernel can access the full 4 GB of 32-bit memory to itself. This extra 0.5-2 GBs are critical for some currently memory bound applications. With a recompilation to 64-bit they can use the full 0.5 TB address space currently offered by the 64-bit kernel.

Currently, the 64-bit Linux kernel limits the address space for 64-bit programs to 512 GB. The limit within the architecture is larger than 512 GB, so future SuSE Linux products for AMD64 architecture will increase the amount of memory a process can address and own.

### Large Files

Due to historic reasons a 32-bit program by default can only operate on files with a size of up to 2 GB. To access larger files a recompilation is needed.

This restriction does not apply to a 64-bit application. Native 64-bit applications on SuSE Linux Enterprise Server 8 for AMD64 can address  $2^{63}$  bytes of file space (over 9,000,000,000,000,000 bytes, or 9 Exa-Bytes) if the underlying file system supports files of this size. Most applications needing this much space (mainly database servers) will likely continue to use raw partitions to avoid file system overhead.

### Selected AMD64 Optimizations for SuSE Linux Enterprise Server

There is not enough space in this white paper to discuss all of the optimizations SuSE made in SuSE Linux Enterprise Server 8 for the AMD64 architecture. However, we do want to discuss some of the more important and interesting enhancements. These enhancements highlight how the combination of SuSE Linux Enterprise Server 8 and AMD64 can boost your application's performance.

#### Optimized Routines

SuSE included in the common GNU C libraries some hand-optimized assembler code for high-usage routines such as string copying. SuSE also optimized the GCC compiler to inline some routines directly for AMD64 systems. In these optimizations, we analyzed those functions that are used most often and would thus benefit most applications.

In addition to these changes, SuSE Linux Enterprise Server 8 incorporated into libm a number of optimized elementary functions from the Numerical Algorithms Group (NAG) which yield better calculation throughput.

#### System Calls are Expensive

Every time a program has to call the Linux kernel, a system call must be performed. Since this involves a context-switch inside the CPU – switching from user mode to kernel mode – system calls are rather expensive in terms of time and CPU power. For Linux on AMD64 a number of optimizations were performed to reduce or eliminate these issues:

The AMD64 CPUs implements a “syscall” instruction that is faster than the traditionally used “int 0x80” instruction. The Linux kernel and C Library takes full advantage of syscall.

Calling conventions for a function are nearly the same as for the kernel. Thus SuSE pipelines the calling sequence and eliminates extra reshuffling of parameter.

In a number of programs, like databases and games, accessing the current wall-clock time is a time-critical operation. Linux on AMD64 is the first Linux architecture to use so-called “Virtual Syscalls”. Instead of doing a system call to get the time, the Linux kernel puts the time in a well-defined place that can be read transparently from the C library, thus avoiding a system call and context switch. This leads to several orders of magnitudes improvement for the *gettimeofday* system call.

#### SMP vs. NUMA

AMD64 users will typically rely heavily on its advanced multi-processing capabilities. For multiprocessing on AMD64, it is important to understand two technical issues: symmetric multi-processing (SMP) and non-uniform memory access (NUMA).

SMP is well understood: SMP systems have multiple processors in one box, and the work load is distributed over these multiple processors. AMD64's usage of the HyperTransport Technology interconnect greatly improves SMP scalability. In an SMP system a process can be placed on any CPU and if a process switches CPUs, it receives a minimal penalty since the CPU caches do not have the cached data of the processes.

Since each CPU has its own memory controller built in, multiprocessor AMD64 systems by definition are Non-Uniform-Memory-Architecture (NUMA) systems. Normally NUMA systems place severe restrictions on where programmers co-locate processes with their associated data and how to access data belonging to other processors. However, the AMD Opteron is designed in a way that developers do not need to worry about this.

AMD64 processors are initialized by the BIOS to locate their respective memories into one globally-addressable physical-memory space. Thus software can access any datum belonging to any processor with normal memory operations to one global address space. In addition, AMD64 processors automatically maintain cache-coherency across this global address space. Lastly, the throughput differences between remote and local memory access are not significant. The end result is that AMD64 systems look like normal SMP systems to users and programmers alike.

Nevertheless a number of NUMA optimizations are possible and we expect to develop more capabilities in future releases. SuSE Linux Enterprise Server 8 contains an experimental NUMA kernel – called `k_numa` – that includes our first optimizations and will help with specific, memory I/O intensive workloads.

For NUMA systems there is the additional cost that after switching CPUs, access to the process's memory is not local anymore but is remote. Some of the NUMA optimizations allocate memory local to the CPU the process is running on and try to avoid migration of processes between CPUs. SuSE is planning to make further enhancements in this area.

#### Shared Libraries

Shared libraries are essential to modern code development, but they can be difficult to deploy in some architectures.

Shared libraries can be loaded at any place in the address space so that they do not conflict with each other. Each program can use several shared libraries and most parts of the libraries can be marked as read-only, thus requiring that they only be loaded into memory once and reducing overall memory usage. To make library position independent, most architectures need an extra register (PIC register) so that global variables and jump targets can be addressed relative to that register.

This extra register is not needed on AMD64 for libraries smaller than 2GB. AMD64 introduces an addressing mode that is relative to the instruction pointer (called relative-instruction-pointer – or RIP – addressing). This mode is used for shared libraries. Using RIP gives shared libraries the same amount of registers as application programs (not one less) and additionally avoids the expensive setup of the extra PIC-register that contain the absolute address. For example on 32-bit x86, the setup code of every exported function in a shared library needs to setup the PIC-register.

## CONCLUSION

AMD has made a number of unique and powerful advancements with their AMD64 processor family. But to take full advantage of this architecture, and to deliver the broadest capabilities, an operating system must:

- Have native support of both 32- and 64-bit applications
- Take advantage of the unique aspects of the architecture
- Be tuned for maximum performance on AMD64

SuSE is proud to be the first operating system vendor to provide from the first day a fully functional Linux operating system for AMD64. Analysts and customers tell us that Linux is the choice for IT and for high performance computing, and with SuSE Linux Enterprise Server 8 for AMD64, all their forward looking server needs are met.

#### **For more information:**

<http://www.suse.com/amd64/>  
<http://www.amd.com/opteron/>

© SuSE Linux AG 2003

Linux is a registered trademark of Linus Torvalds.

UNIX is a registered trademark of The Open Group.

All other company, product, and service names or designations may be trademarks or service marks or registered trademarks or service marks of other companies around the world and shall be treated as such.