Linux on AMD64

Michal Ludvig

michal@logix.cz

http://www.logix.cz/michal

x86 – your 32 bit CPU

- No significant changes since 80386 (1985!)
- LP32 architecture
  - can address 2^{32} bytes = 4GB per process
  - max 4GB of physical RAM
  - with ugly hacks up to 64GB RAM
    - still only 4GB/task!
- each process has its own virtual memory
  - code, libraries, data and kernel share the same address space

Going 64 bit – AMD64

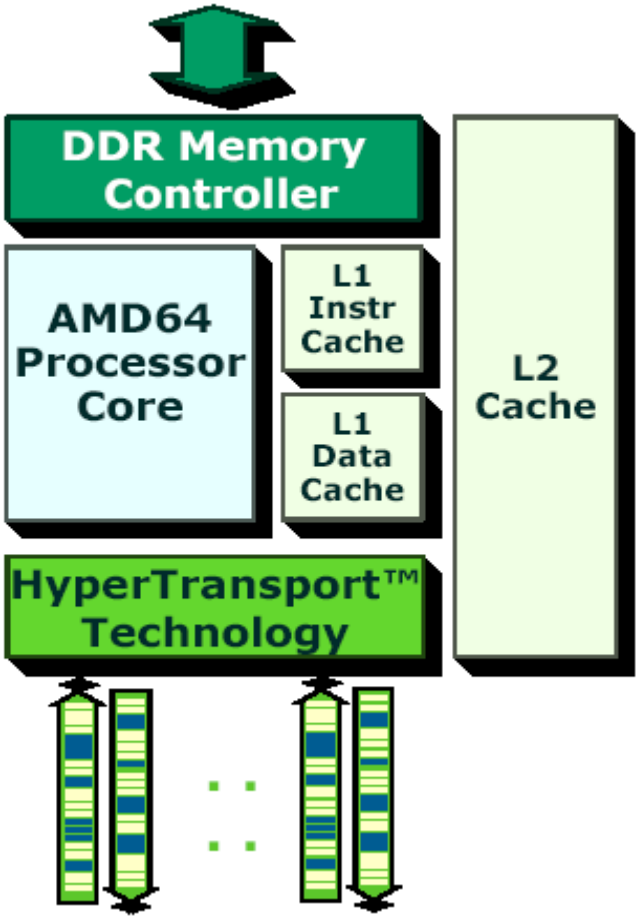
- LP64 – `long int` and pointers have 64 bits
  - huge addressable space: 2^{64} Bytes is ***a lot***
  - faster operations with large integers
- multiple operating modes
  - backward compatible *real* and *protected* modes
  - *long mode* for native 64 bit kernel and processes
    - 32b processes run in *legacy mode* under 64bit kernel
      - fast, native, not emulated like on Itanium

AMD Opteron and Athlon64 (I.)

- Improved core architecture
  - 64 bit *long mode*
  - Doubled number of GPRs
  - Supports **3Dnow!**, **SSE** and **SSE2** with 16 XMM/128b registers
  - NX pages against buffer overflows
- IOMMU unit
  - gives 32 bit PCI cards access to the whole 64 bit address space

AMD Opteron and Athlon64 (II.)

- integrated memory controller
  - can address L1, L2 cache and main memory at the same time
  - memory attached directly to the CPU
- multiple **HyperTransport** buses for connecting CPU↔CPU and CPU↔peripherals
  - up to **12.8 GB/s** full duplex



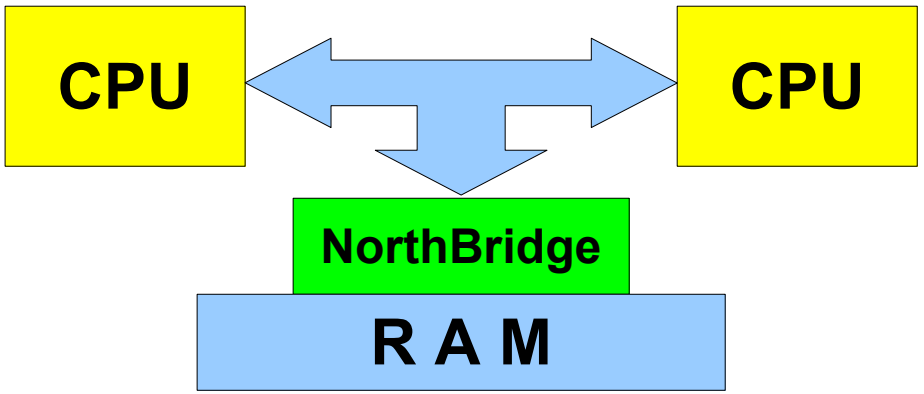
Intel EM64T

- New **Xeons** with 800MHz FSB
- Instruction-wise compatible with **AMD64**
- Doesn't have integrated memory controller
- Doesn't have **IOMMU**
  - 32 bit PCI cards may slow the system
- No **HyperTransport** – limited to single **FSB**
- **SSE3** instead of **3Dnow!**

MP systems – NUMA vs SMP

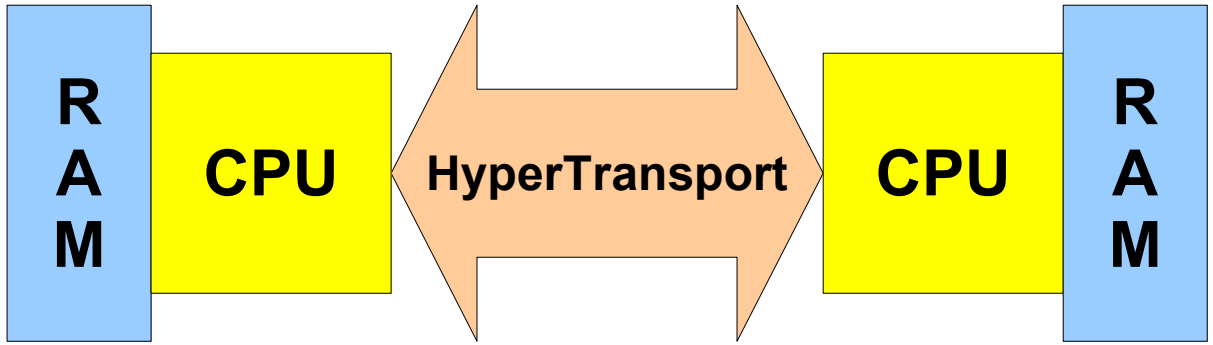
- **NUMA** – Non-Uniform Memory Access
  - each CPU has its own memory
  - indirect access to other CPU's memory

- **SMP** – Symmetric Multi-Processing
  - all CPUs access the same memory
  - single memory controller in *northbridge*



NUMA + SMP = SUMO

- **Sufficiently Uniform Memory Organization**
  - introduced in AMD64 MP systems
  - each CPU has directly attached memory
  - high speed access to other memory over **Hyper-Transport** bus
  - transparent to the operating system



Linux on x86-64

- kernel is fully ported
  - including all HW drivers, frame buffer, software suspend, power management, ...
  - 64bit kernel supports both 32bit and 64bit tasks
    - e.g. 32 bit Firefox runs just fine in 64 bit X11
- most userspace programs run in 64 bits
- redesigned ABI for faster run
  - calling conventions enjoy more registers

cat /proc/cpuinfo

```
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 15
model           : 12
model name      : AMD Athlon(tm) 64 Processor 3200+
stepping        : 0
cpu MHz         : 2211.110
cache size      : 512 KB
fpu             : yes
fpu_exception   : yes
cpuid level     : 1
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca
    cmov pat pse36 clflush mmx fxsr sse sse2 syscall nx mmxext lm 3dnowext
    3dnow
bogomips        : 4430.14
TLB size        : 1024 4K pages
clflush size    : 64
cache_alignment : 64
address sizes   : 40 bits physical, 48 bits virtual
power management: ts fid vid ttp
```

Applications

- System provides both 32 bit and 64 bit libs

32 bit system:

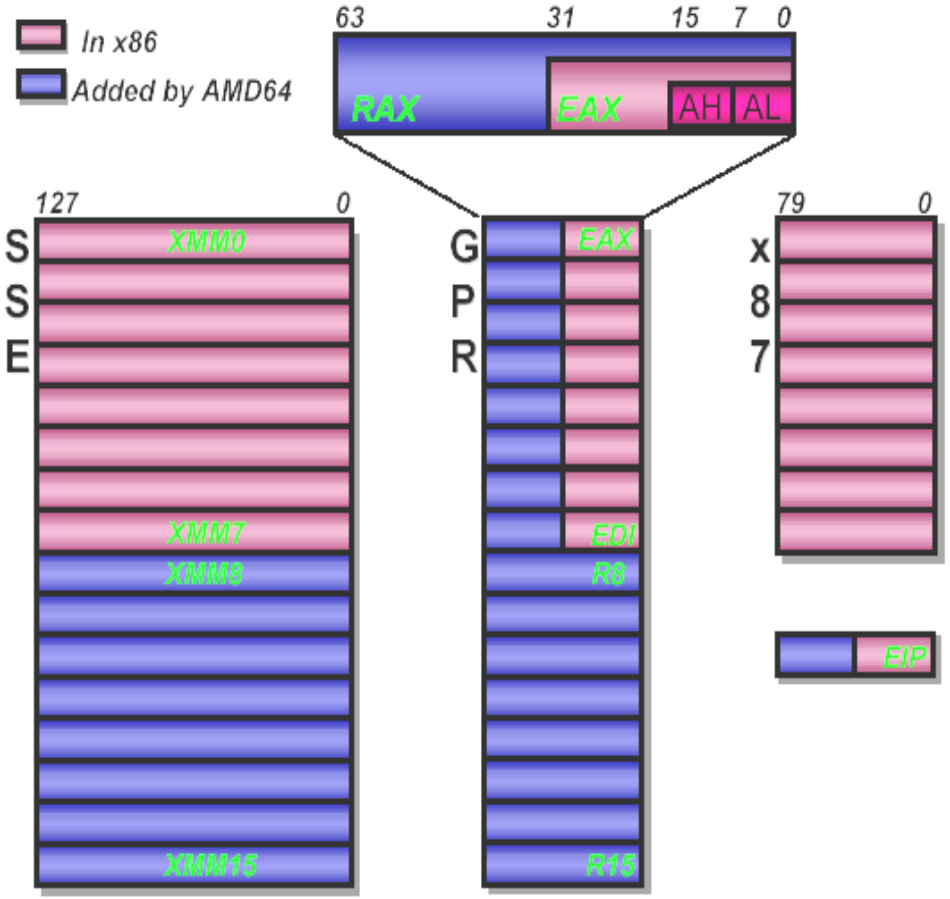
```
x86:~$ file /bin/echo
   /bin/echo: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), [...]
x86:~$ ldd /bin/echo
   libc.so.6 => /lib/tls/libc.so.6 (0x4002e000)
   /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

64 bit system:

```
x86-64:~ $ file /bin/echo
   /bin/echo: ELF 64-bit LSB executable, AMD x86-64, version 1 (SYSV), [...]
x86-64:~$ ldd /bin/echo
   libc.so.6 => /lib64/tls/libc.so.6 (0x00002aaaaabc2000)
   /lib64/ld-linux-x86-64.so.2 => /lib64/ld-linux-x86-64.so.2
                                                (0x00002aaaaaaab000)
```

Low level

- extended registers:
  AL→AX→EAX→RAX
- extra eight registers called **R8** to **R15**
- **GCC** can generate both 64 bit and 32 bit code
  - default is 64 bit code
  - ***gcc -m32*** for 32 bit output



Instruction set

- System calls via ***syscall***
  - no ***int 0x80*** anymore
- Extended AT&T assembly syntax

```
movl   $1, %eax          ; 32 bits
movq   $1, %rax          ; 64 bits
```

- RIP-relative addressing

```
movl   $1, symbol(%rip)
```

- REX prefix

Distributions

- ported by SUSE
- all patches given back to community
- installation and look&feel is the same as on x86
- most major distributions have AMD64/x86-64 variant
- SUSE still actively develops AMD64 kernel and toolchain
  - at least the kernel and GCC for AMD64 are a step ahead of other distributions

Porting Linux to AMD64 (I.)

- Close cooperation between SUSE and AMD
  - some CPU features requested by SUSE hackers
  - direct contact without the managers' overhead
  - famous dinners paid by AMD ;-)

Porting Linux to AMD64 (II.)

- mostly done on simulators (Virtutech ***Simics***)
- toolchain development heavily used cross compilation
- core porting – ca 10 developers in 2 years
  - binutils, GCC, kernel, glibc, GDB
- remaining >2000 packages was much easier task

Porting Linux to AMD64 (III.)

- first real ***Claw hammer*** CPU in SUSE
  - 800MHz
  - disabled caches and TLB
  - really slow but still much faster than ***Simics***
  - motherboard ***Solo***
- the only other AMD64 system outside AMD was in Microsoft

Useful links

- AMD64 developers guide (5 volumes)
- AMD64 Application Binary Interface
- Gentle Introduction to x86-64 Assembly
- AMD64 registers overview

Thank you