OPERATING SYSTEMS

Memory Management

Suresh Jamadagni
Department of Computer Science

OPERATING SYSTEMS

Intel 32 and 64-bit Architectures

Suresh Jamadagni
Department of Computer Science

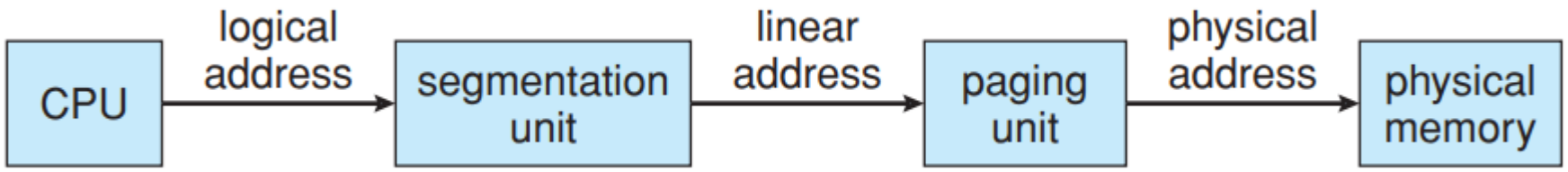
OPERATING SYSTEMS

Slides Credits for all the PPTs of this course

- The slides/diagrams in this course are an **adaptation**, **combination**, and **enhancement** of material from the following resources and persons:

1. Slides of Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne - 9th edition 2013 and some slides from 10th edition 2018
2. Some conceptual text and diagram from Operating Systems - Internals and Design Principles, William Stallings, 9th edition 2018
3. Some presentation transcripts from A. Frank – P. Weisberg
4. Some conceptual text from Operating Systems: Three Easy Pieces, Remzi Arpaci-Dusseau, Andrea Arpaci Dusseau

Logical to physical address translation in IA-32



- Memory management in IA-32 systems is divided into two components— segmentation and paging:
- The CPU generates logical addresses, which are given to the segmentation unit.
- The segmentation unit produces a linear address for each logical address.
- The linear address is then given to the paging unit, which in turn generates the physical address in main memory.
- The segmentation and paging units form the equivalent of the memory-management unit (MMU).

OPERATING SYSTEMS

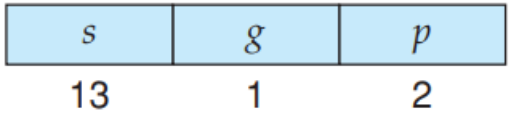
IA-32 Segmentation

- The IA-32 architecture allows a segment to be as large as 4 GB
- Maximum number of segments per process is 16 K.
- The logical address space of a process is divided into two partitions. The first partition consists of up to 8K segments that are private to that process. The second partition consists of up to 8 K segments that are shared among all the processes.
- Information about the first partition is kept in the **local descriptor table (LDT)** and information about the second partition is kept in the **global descriptor table (GDT).**
- Each entry in the LDT and GDT consists of an 8-byte segment descriptor with detailed information about a particular segment, including the base location and limit of that segment.

OPERATING SYSTEMS

IA-32 Segmentation (contd…)

- The logical address is a pair (selector, offset), where the selector is a 16-bit number
- s designates the segment number, g indicates whether the segment is in the GDT or LDT, and p deals with protection
- The offset is a 32-bit number specifying the location of the byte within the segment in question.

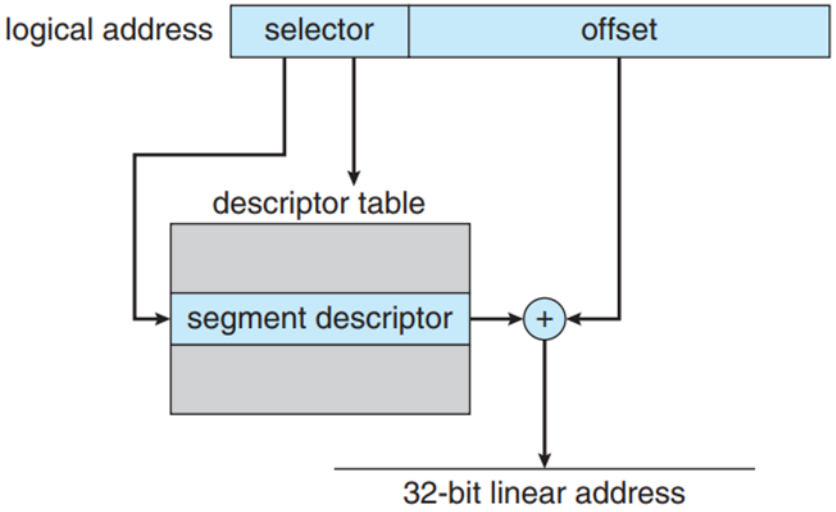


- The machine has **six segment registers**, allowing six segments to be addressed at any one time by a process.
- It also has **six 8-byte microprogram registers** to hold the corresponding descriptors from either the LDT or GDT. This cache lets the Pentium avoid having to read the descriptor from memory for every memory reference

OPERATING SYSTEMS

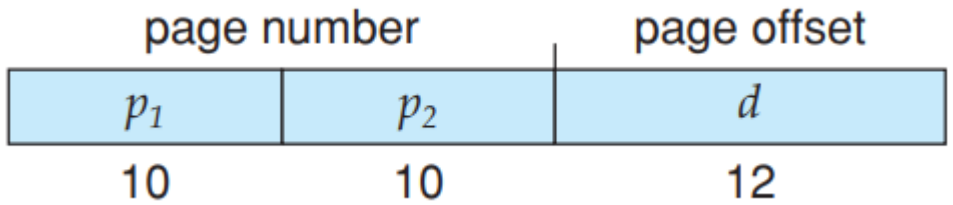
IA-32 Segmentation (contd…)

- The linear address on the IA-32 is 32 bits long and is formed as follows:
  - The segment register points to the appropriate entry in the LDT or GDT. The base and limit information about the segment in question is used to generate a linear address.
  - First, the limit is used to check for address validity. If the address is not valid, a memory fault is generated, resulting in a trap to the operating system.
  - If it is valid, then the value of the offset is added to the value of the base, resulting in a 32-bit linear address.



IA-32 Paging

- The IA-32 architecture allows a page size of either 4 KB or 4 MB.
- For 4-KB pages, IA-32 uses a two-level paging scheme in which the division of the 32-bit linear address is as follows:

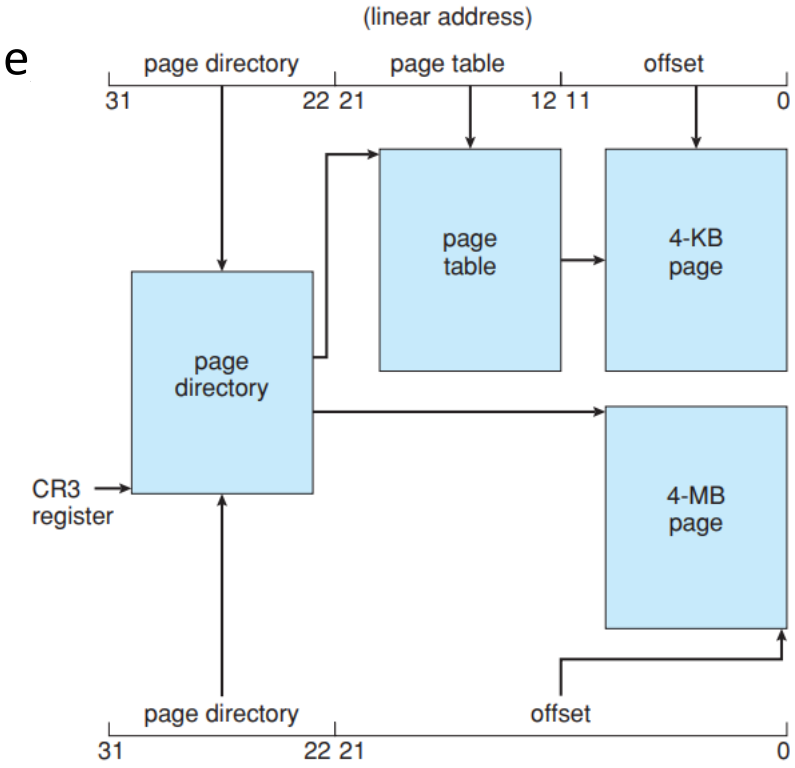


- The 10 high-order bits reference an entry in the outermost page table, which IA-32 terms the page directory. (The CR3 register points to the page directory for the current process.)
- The page directory entry points to an inner page table that is indexed by the contents of the innermost 10 bits in the linear address.
- The low-order bits 0–11 refer to the offset in the 4-KB page pointed to in the page table.

OPERATING SYSTEMS

IA-32 Paging

- One entry in the page directory is the Page Size flag, which if set, indicates that the size of the page frame is 4 MB and not the standard 4 KB.
- If this flag is set, the page directory points directly to the 4-MB page frame bypassing the inner page table; and the 22 low-order bits in the linear address refer to the offset in the 4-MB page frame



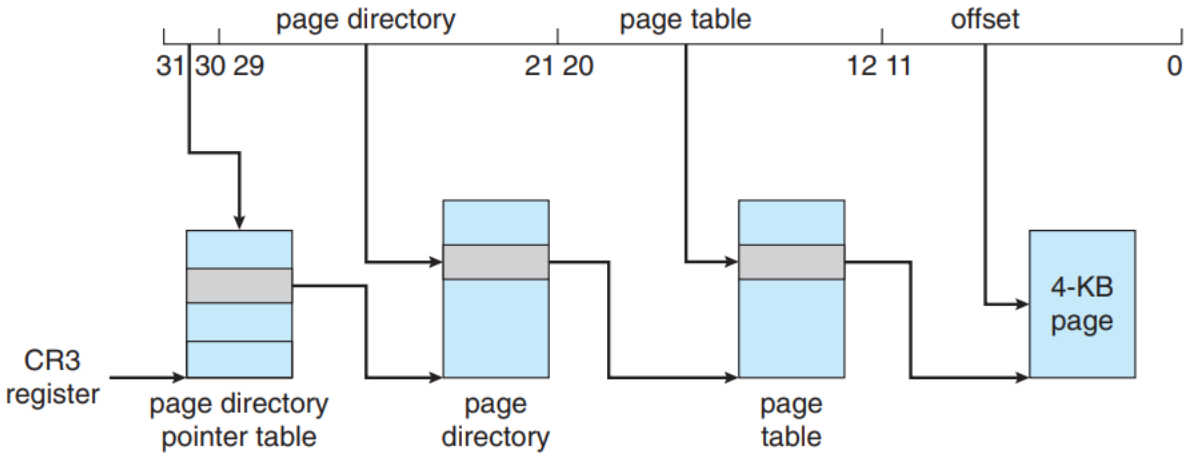
OPERATING SYSTEMS

IA-32 Paging

- To improve the efficiency of physical memory use, IA-32 page tables can be swapped to disk.
- An invalid bit is used in the page directory entry to indicate whether the table to which the entry is pointing is in memory or on disk.
- If the table is on disk, the operating system can use the other 31 bits to specify the disk location of the table.
- The table can then be brought into memory on demand.

OPERATING SYSTEMS

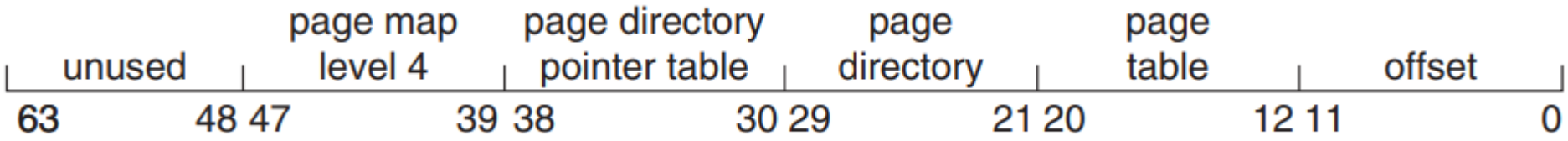
IA-32 Physical Address Extension



- Intel adopted a page address extension (PAE), which allows 32-bit processors to access a physical address space larger than 4 GB.
- The fundamental difference introduced by PAE support was that paging went from a two-level scheme to a **three-level scheme**, where the top two bits refer to a page directory pointer table.
- PAE also increased the page-directory and page-table entries from 32 to 64 bits in size, which allowed the base address of page tables and page frames to extend from 20 to 24 bits.
- Combined with the 12-bit offset, adding PAE support to IA-32 increased the address space to 36 bits, which supports up to 64 GB of physical memory.

OPERATING SYSTEMS

X86-64



- The x86-64 supported much larger logical and physical address spaces, as well as several other architectural advances.
- Support for a **64-bit address space yields an astonishing 264 bytes of addressable memory**—a number greater than 16 quintillion (or 16 exabytes).
- Even though 64-bit systems can potentially address this much memory, in practice far fewer than 64 bits are used for address representation in current designs.
- The x86-64 architecture currently provides a 48-bit virtual address with support for page sizes of 4 KB, 2 MB, or 1 GB using four levels of paging hierarchy
- Because this addressing scheme can use PAE, virtual addresses are 48 bits in size but support 52-bit physical addresses (4096 terabytes)

Additional reference

https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

THANK YOU

Suresh Jamadagni
Department of Computer Science Engineering

sureshjamadagni@pes.edu