

# Announcements

- Deadline for midterm re-grades is Thursday
- Start on project #4 before spring break

# Steps of Project

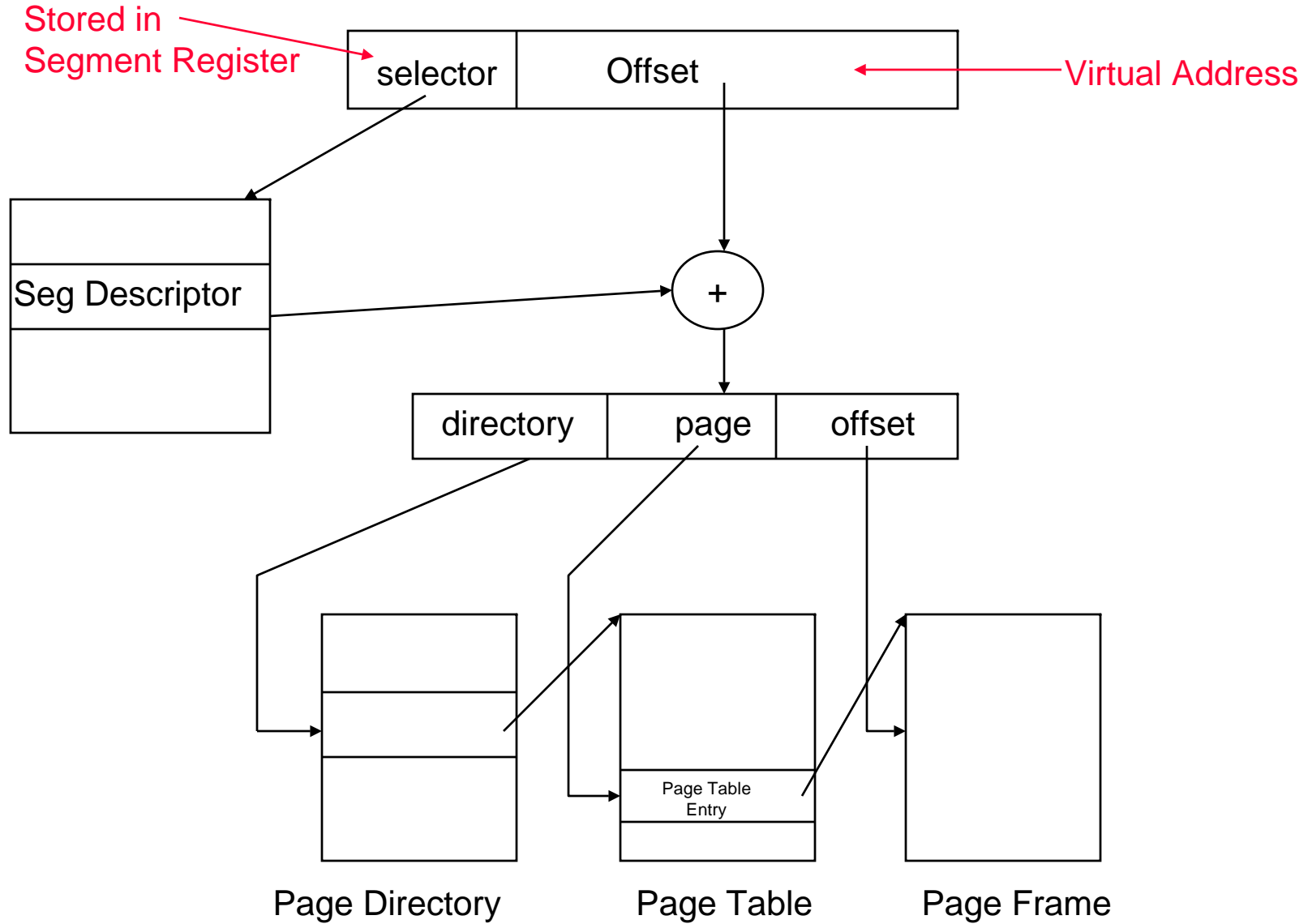
- Enable Paging
  - Map all of physical memory
    - Identity map
- Get separate page table for User Process
  - Map user pages at 2GB
  - Update Segment Info
  - Context switch PTBR
- Get page faults working

# Segmentation

- Segmentation is used to give each program several independent protected address spaces
  - each segment is an independent protected address space
  - access to segments is controlled by data which describes size, privilege level required to access, protection (whether segment is read-only etc)
  - segments may or may not overlap
    - disjoint segments can be used to protect against programming errors
    - separate code, data stack segments

- Disjoint Segments can be used to exploit expanded address space
  - In 16 bit architectures e.g. (8086 and 80x86 in V86 mode) each segment has only 16 bits of address space
  - In distributed networks consisting of multiple 32 bit machines, segmentation can be used to support single huge address space
- Segments can span identical regions of address space - *flat model*
  - Windows NT and Windows '95 use 4 Gbyte code segments, stack segments, data segments

# X86 Segmentation + Paging



# 64 bit processors

- Problem: 2 level page tables are too small
- Solution 1:
  - Use more levels & larger page size
    - Alpha:
      - 3 level
      - variable size pages
      - w8KB pages
        - 43 bits of virtual address
        - 13 bits page offset
        - $3 \times 10 = 30$  bits in page tables
      - w64KB pages
        - 55 bits of virtual address
        - 16 bits page offset
        - $3 \times 13 = 39$  bits in page tables

# Sparc & IBM Power 64 bit processors

- Ultra Sparc 64 bit MMU
  - 8KB, 16KB, 512KB, 4MB pages supported
  - Software TLB miss handler
  - 44 bit virtual address
- Power 4
  - Variable sized pages up to 16MB
  - Inverted page tables
  - TLB
    - 1024 entry 4-way set associate
  - TLB cache
    - Called ERAT
      - 128 entry 2-way set associative

# Other 64-bit Designs

- AMD-64
  - 54 bit physical memory
  - With 4KB pages
    - 48 bits of virtual address are used
    - 4KB pages
      - 12 bits page
      - $4 \times 9 = 36$  bits via 4-level page tables
    - 2MB pages
      - 21 bits page
      - $3 \times 9 = 27$  bits via 3-level page tables