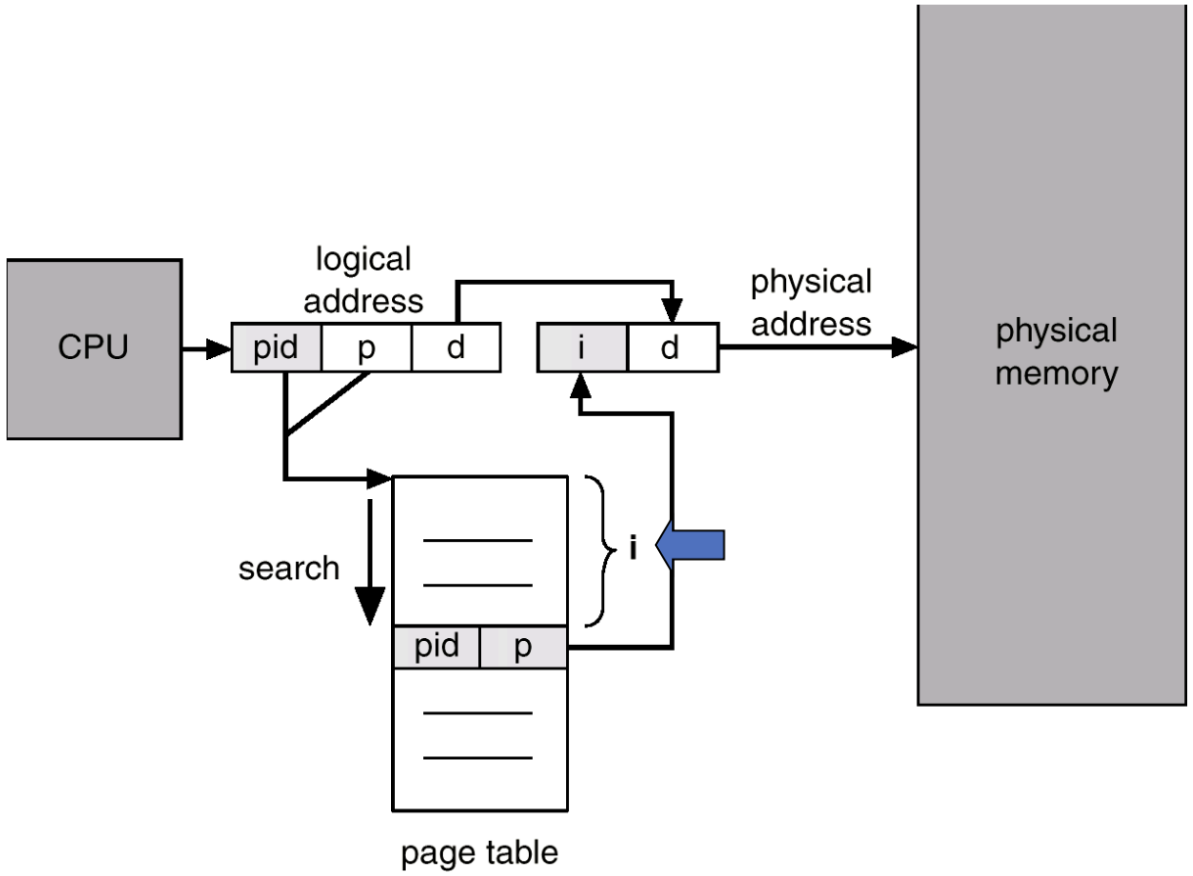
4주차_메모리 관리3

상태	진행 중
강의	CS 스터디
작성일	@2024년 2월 23일

Inverted Page Table

- page table이 매우 큰 이유
  - 모든 process 별로 그 logical address에 대응하는 모든 page에 대해 page table entry사 존재
  - 대응하는 page가 메모리에 있든 아니든 간에 page table에는 entry로 존재
- Inverted page Table
  - Page frame 하나당 page table에 하나의 entry를 둔 것 (sysyem-wide)
  - 각 page table entry는 각각의 물리적 메모리의 page frame에 담고 있는 내용 표시(process-id), process의 logical address)
  - 단점
    - 테이블 전체를 탐색해야 함
  - 조치
    - associative register 사용 (expensive)
      병렬적 탐색을 하는 하드웨어

Inverted Page Table Architecture

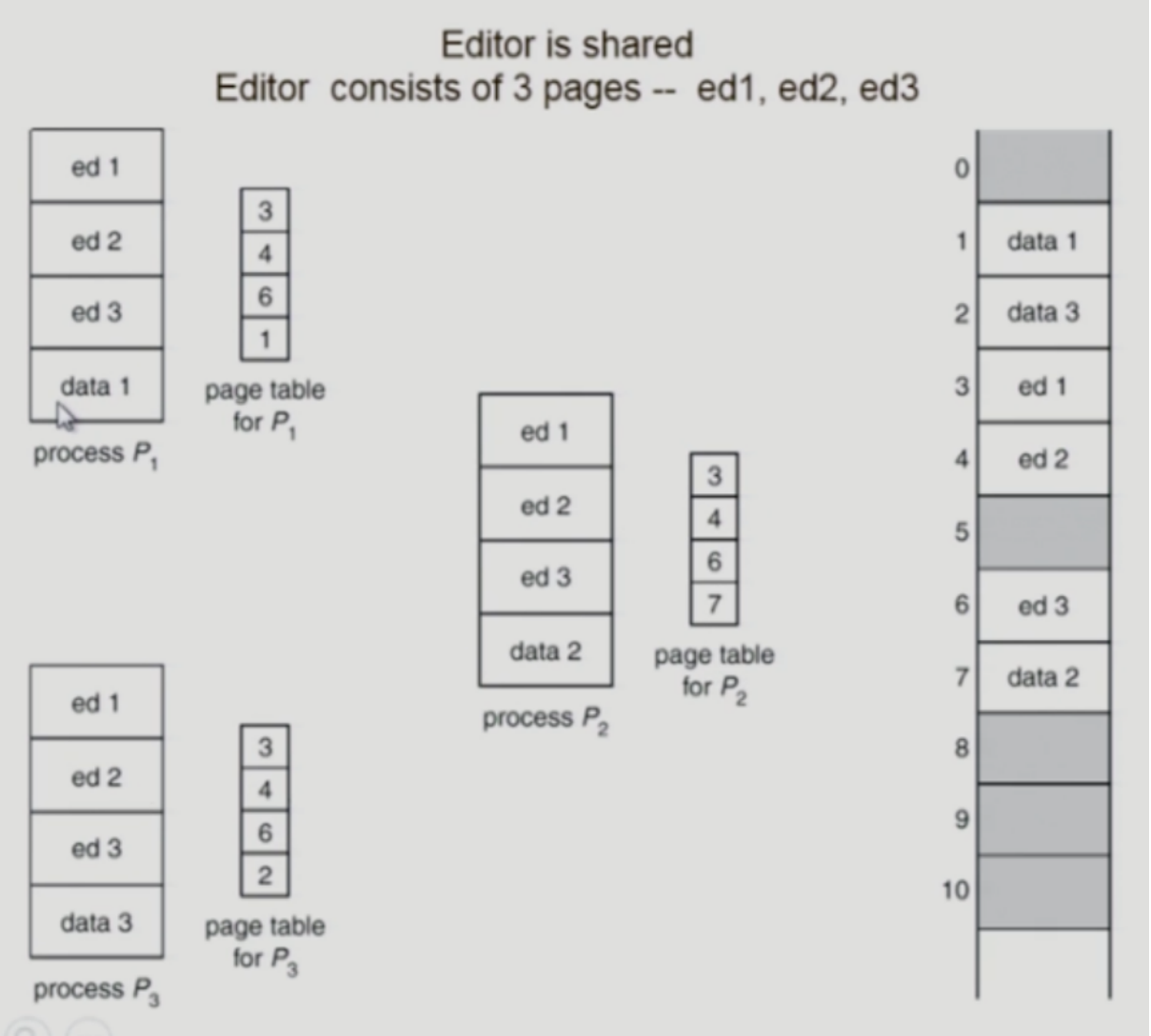


주소 변환에는 도움이 되지 않아

Shared Page

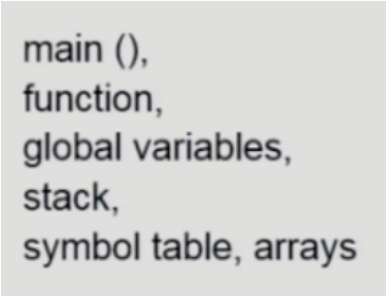
- Shared Code
  - Re-entrant Code(= Pure Code)
  - read-only로 하여 프로세스 간에 하나의 code만 메모리에 올림
    (e.g. text editors, compilers, window systems)
  - Shred Code는 모든 프로세스의 logical address space에서 동일한 위치에 있어야함
    페이지가 같아야한다.
- Private code and data
  - 각 프로세스들은 독자적으로 메모리에 올림
  - Private data는 logical address space의 아무 곳에 와도 무방

Shared Page Example



Segmentation

- 프로그램은 의미 단위인 여러개의 segment로 구성
  - 작게는 프로그램을 구성하는 함 수 하나하나를 세그먼트로 정의
  - 크게는 프로그램 전체를 하나의 세그먼트로 정의 가능
  - 일반적으로는 code, data, stack 부분이 하나씩의 세그먼트로 정의됨
- Segment는 다음과 같은 logical unit들임



Segment Architecture

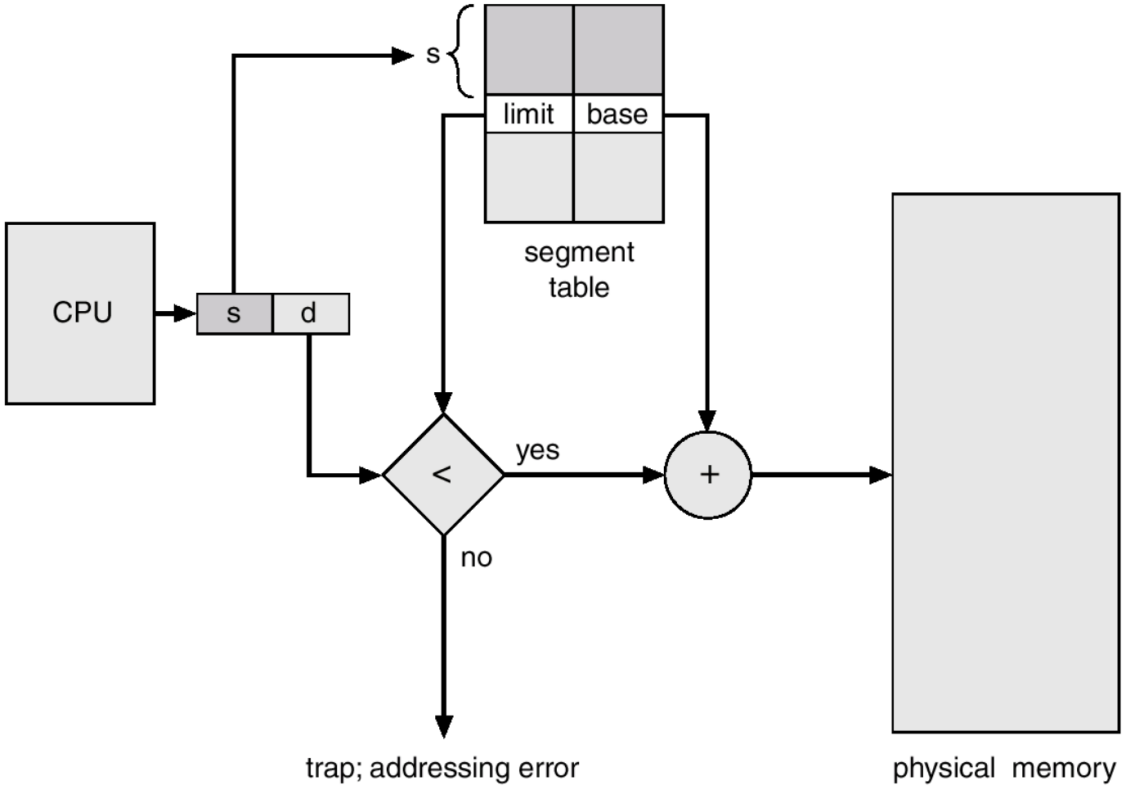
- Logical address structure는 다음의 두 가지로 구성

<Segment-number, offset>

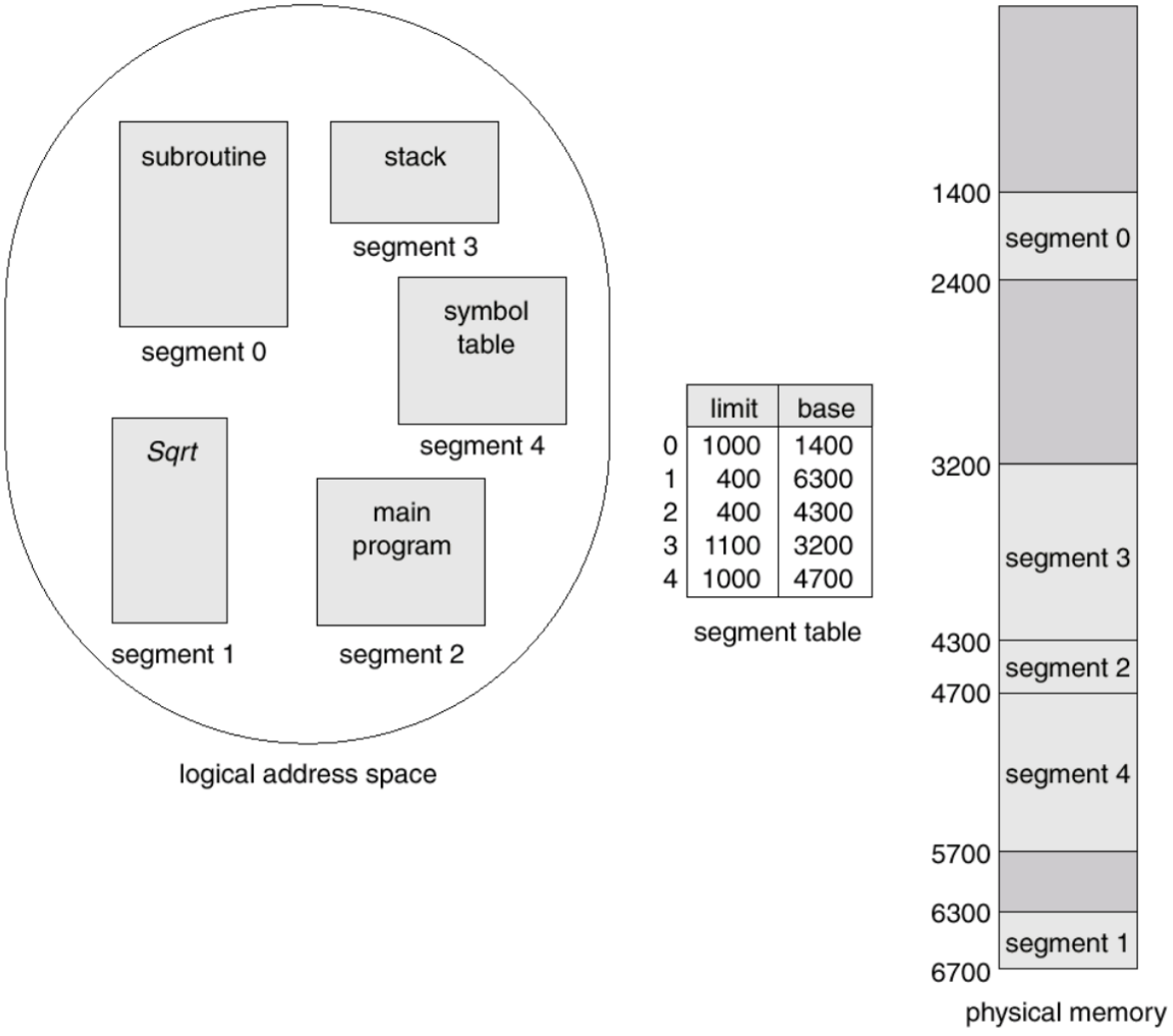
- Segment table
  - each table entry has:
    - base - starting physical address of the segment
    - limit - length of the segment
- Segment-table base register(STBR)
  - 물리적 메모리에서의 segment table의 위치
- Segment-table length register(STLR)
  - 프로그램이 사용하는 segment의 수

segment number is legal is s < STLR

Segment Hardware



Example of Segmentation



Segmentation Architecturen(Cont.)

- Protection
  - 각 세그먼트 별로 protection bit가 있음
  - Each entry:
    - Valid bit = 0 ⇒ illegal segment
    - Read/Write/Execution 권한 bits
- Sharing
  - shared segmetnt
  - same segment number

*** segment는 의미 단위이기 때문에 공유(sharing)와 보안(protection)에 있어 paging보다 훨씬 료과적이다.

- Allocation
  - first-fit / best-fit
  - external fragmentation 발생

*** segment의 길이가 동일하지 않으므로 가변분할 방식에서와 동일한 문제점들이 발생

Segmentation with Paging

- pure segmentation과의 차이점
  - segment-table entry가 segment의 base address를 가지고 있는 것이 아니라 segment를 구성하는 page table의 base address를 가지고 있음