

Chapter.8 OS Support

OS

- Convenience: 사용하는데 편리
- Efficiency: computer resource의 활용도를 극대화

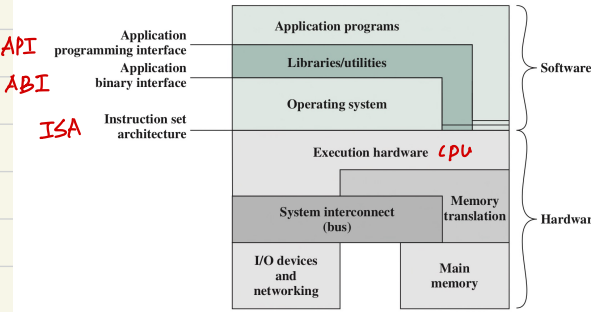


Figure 8.1 Computer Hardware and Software Structure

- OS는 일반 software라 함
- OS는 빈번히 제어를 양호함
- OS는 Main Memory에 위치

<Types of OS>

- Interactive system

: User/programmer는 키보드/터미널을 이용한 컴퓨터와의 직접적인 대화를 통하여 각급실행을 요청

- Batch system (일괄 처리 시스템)

Multi programming / multiprocessing

Multiprogrammed Batch System - 데이터 전송하는데 95%

Read one record from file	15 μ s
Execute 100 instructions	1 μ s
Write one record to file	15 μ s
TOTAL	31 μ s

$$\text{Percent CPU utilization} = \frac{1}{31} = 0.032 = 3.2\%$$

Multiprogramming
(multitasking)

Services

- Program creation
- Program execution
- Access to I/O devices
- Controlled access to files
- System access
- Error detection & response
- Accounting

ISA: machine language
ABI
API

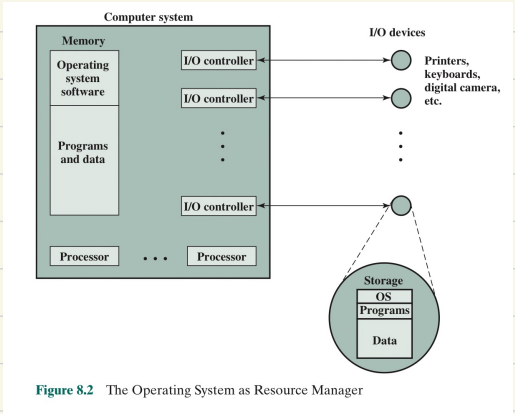


Figure 8.2 The Operating System as Resource Manager

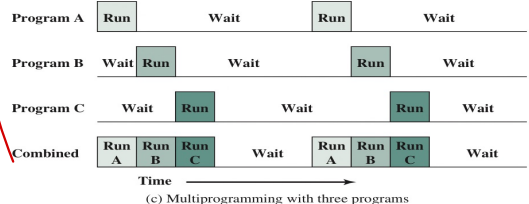
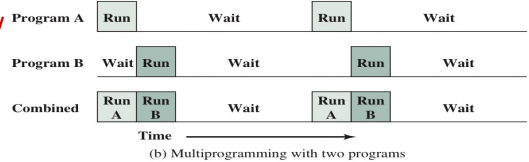
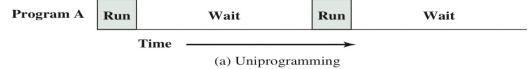


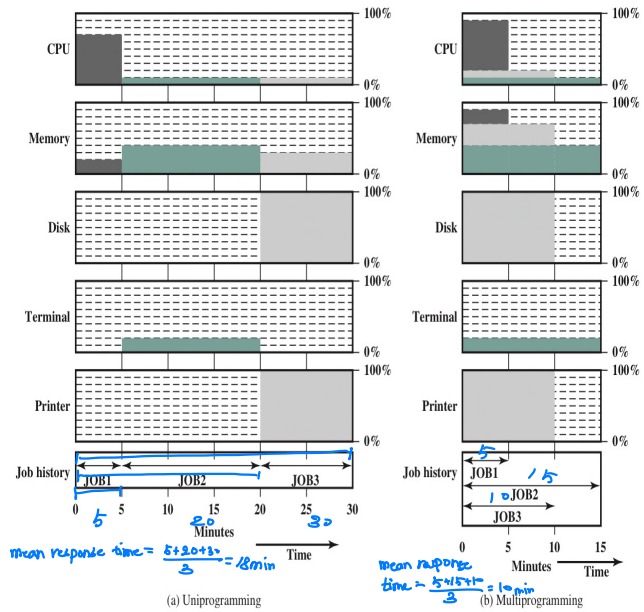
Figure 8.5 Multiprogramming Example

Table 8.1 Sample Program Execution Attributes

	JOB1	JOB2	JOB3
Type of job	Heavy compute	Heavy I/O	Heavy I/O
Duration (min)	5	15	10
Memory required (M)	50	100	80
Need disk?	No	No	Yes
Need terminal?	No	Yes	No
Need printer?	No	No	Yes

Table 8.2 Effects of Multiprogramming on Resource Utilization

	Uniprogramming	Multiprogramming
Processor use (%)	20	40
Memory use (%)	33	67
Disk use (%)	33	67
Printer use (%)	33	67
Elapsed time (min)	30	15
Throughput rate (jobs/hr) 도달률	6	12
Mean response time (min) 평균 응답 시간	18	10



CPU Utilization (CPU 활용도)

a) $(0.1 \times 5 + 0.1 \times 15 + 0.1 \times 10) / 30 = 6 / 30 = 0.2$

(20%)

b) $(0.1 \times 5 + 0.1 \times 10 + 0.1 \times 15) / 15 = 6 / 15 = 0.4$

(40%)

scheduling Time sharing

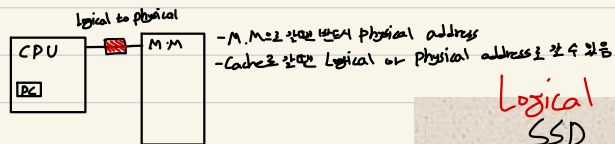
오직 3개

21 개

process

- 하나의 프로그램은 여러개의 process로 나뉨
- 실행중인 프로그램

Logical address (SSD) - virtual mem-ry
 Physical address (main memory)
 8GB (유저가 쓸 수 있는 data: 6GB)



1 page = 512B
 2GB = 2^{30} B
 1 process = 2^{22} pages
 page fault (similar to cache miss)

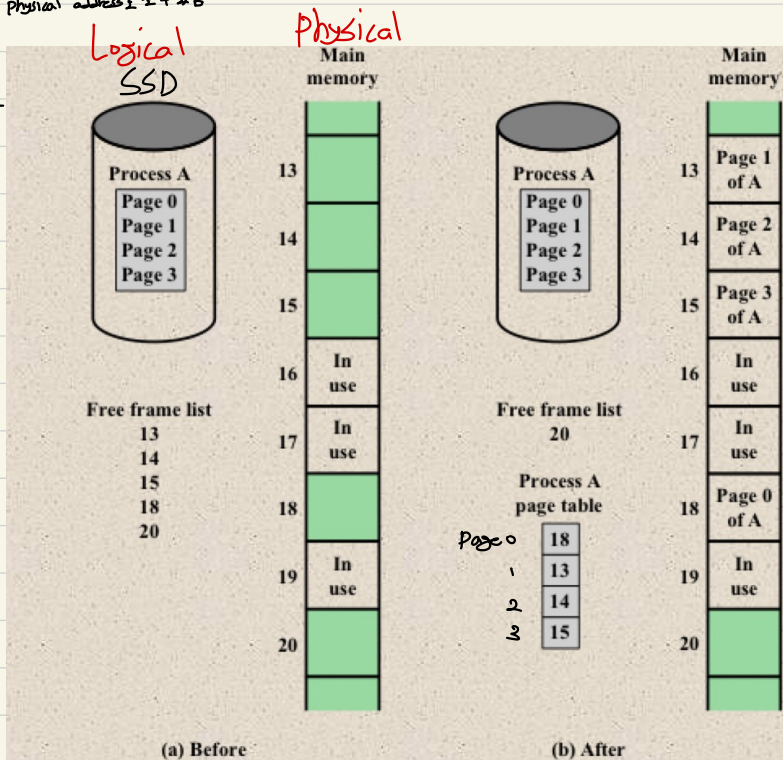


Figure 8.15 Allocation of Free Frames

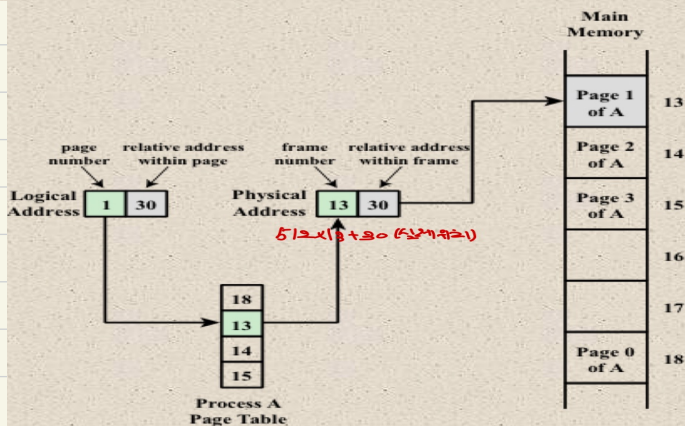


Figure 8.16 Logical and Physical Addresses

Virtual Memory (외장하드) Demand Paging

- Process의 Page는 필요할 때만 불러옴
- Principle of Locality (지역성의 원리): 자주 사용되는 Page 참조

- 장점
- 시간 절약
 - 공간 절약
 - 각 프로세스의 메모리 페이지를 적재하여 외부 프로세스 적재 가능
- 단점
- Page replacement
 - Thrashing: 빈번히 Page fault가 일어남

2GB Process / Page당 0.5k / main memory
 2^9 / 8MB = 2^{23}
 2^{31} 총 Page < 2^{23} ← 1GB 2^{30} 총 2^{14} frames

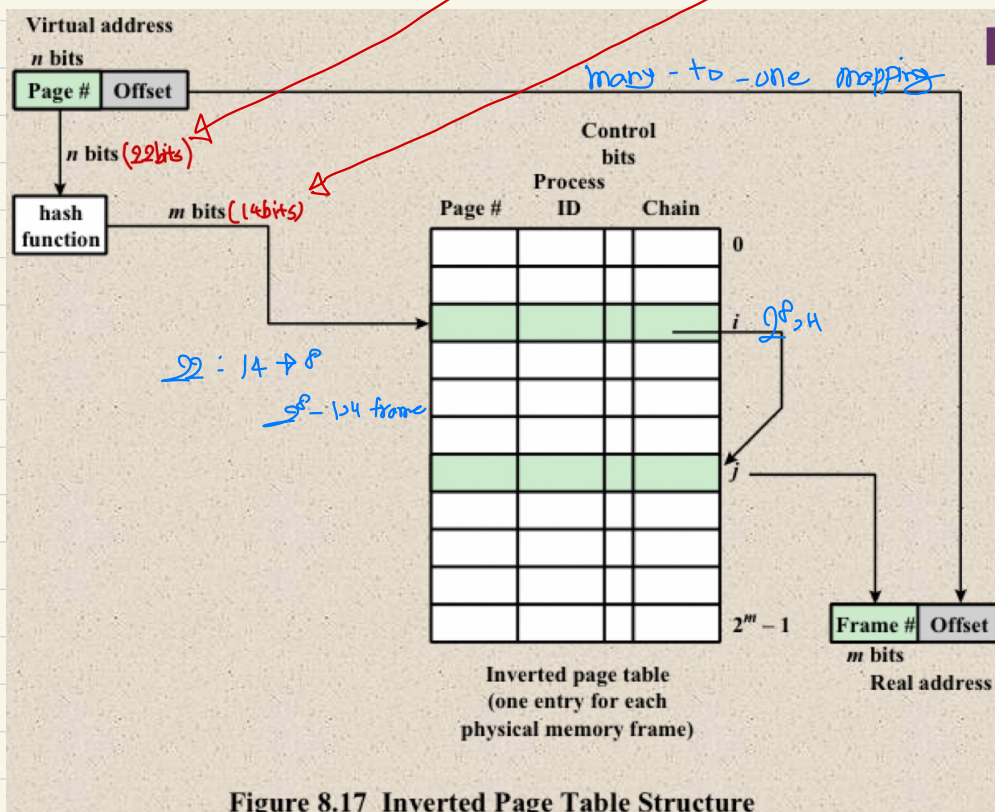


Figure 8.17 Inverted Page Table Structure

TLB (Translation Lookaside Buffer)

: virtual to m.m

logical to physical address