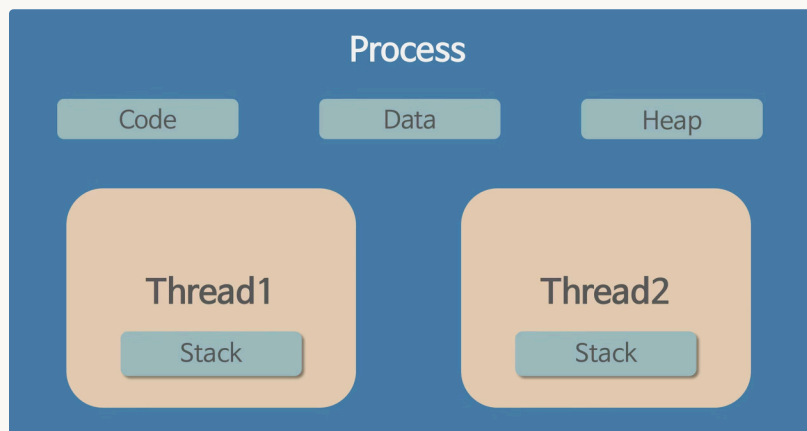


프로세스(Process) vs 스레드(Thread)

By commentLee

프로세스 vs? 스레드



면접에서 굉장히 자주 나오는 질문. 그런데..

- 프로세스 안에 스레드가 존재. (현재) 대립하는 개념아님
- 서로 다른 개념처럼 생각하지 않기!
- 프로세스 = 하나의 실행 흐름(단일스레드)일 시절에는 이런 비교가 맞지만 지금은 아니다.

프로그램

OS의 시선

- 하드웨어를 논리적인 자원으로 추상화
- 자원을 잘 정의된 인터페이스로 추상화

코딩을 한 뒤

- 소스코드가 일련의 과정을 거쳐 프로그램이 된다.

프로그램은? sequences of instructions

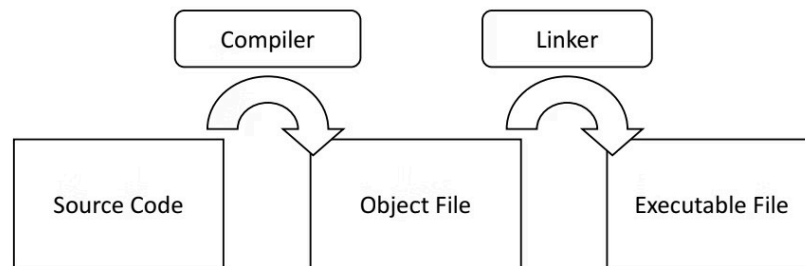
- OS 에겐 그저 명령어의 모음
- CPU를 주면 알아서 처리하다가 다시 반환된다

Abstraction

- The OS **abstracts** hardware into **logical resources** and well-defined **interfaces** to those resources

- Programs (sequences of instructions)
- Processes (CPU, memory)
- Files (storage)
- Sockets (network)

논리적인 자원과 인터페이스
물리적인 자원



프로세스(Process)란?

정의: 실행 중인 프로그램 (Program in Execution)

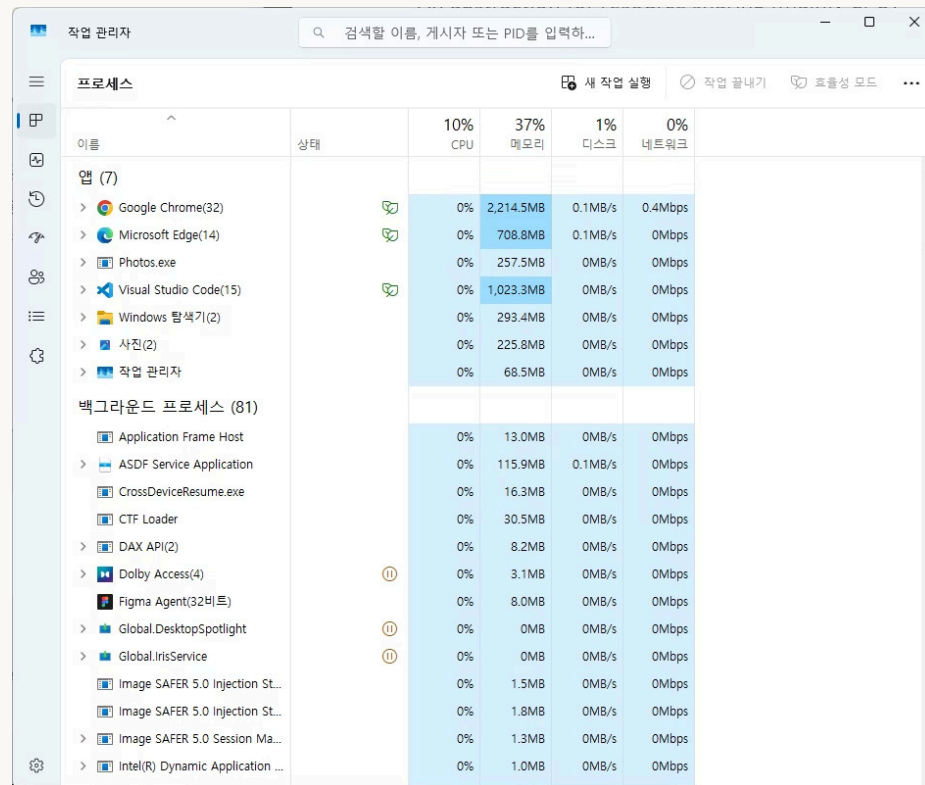
- CPU, 메모리 등 **자원 할당의 단위**

구성요소: Context(s)와 자원들

- 메모리(virtual): code, data, heap, stack
- storage
- network

Context = 수행 흐름 = 스레드

- 프로그램의 시작부터 끝까지 수행되는 흐름
- 프로세스는 최소 1개의 수행 흐름을 포함



The screenshot shows the Windows Task Manager window with the 'Processes' tab selected. It displays a list of running applications and background processes, including Google Chrome, Microsoft Edge, Photos, Visual Studio Code, Windows Explorer, and various system services. The columns show the process name, status, CPU usage, memory usage, disk usage, and network usage.

이름	상태	10% CPU	37% 메모리	1% 디스크	0% 네트워크
앱 (7)					
Google Chrome(32)		0%	2,214.5MB	0.1MB/s	0.4Mbps
Microsoft Edge(14)		0%	708.8MB	0.1MB/s	0Mbps
Photos.exe		0%	257.5MB	0MB/s	0Mbps
Visual Studio Code(15)		0%	1,023.3MB	0MB/s	0Mbps
Windows 탐색기(2)		0%	293.4MB	0MB/s	0Mbps
사진(2)		0%	225.8MB	0MB/s	0Mbps
작업 관리자		0%	68.5MB	0MB/s	0Mbps
백그라운드 프로세스 (81)					
Application Frame Host		0%	13.0MB	0MB/s	0Mbps
ASDF Service Application		0%	115.9MB	0.1MB/s	0Mbps
CrossDeviceResume.exe		0%	16.3MB	0MB/s	0Mbps
CTF Loader		0%	30.5MB	0MB/s	0Mbps
DAX API(2)		0%	8.2MB	0MB/s	0Mbps
Dolby Access(4)		0%	3.1MB	0MB/s	0Mbps
Figma Agent(32비트)		0%	8.0MB	0MB/s	0Mbps
Global.DesktopSpotlight		0%	0MB	0MB/s	0Mbps
Global.IrisService		0%	0MB	0MB/s	0Mbps
Image SAFER 5.0 Injection St...		0%	1.5MB	0MB/s	0Mbps
Image SAFER 5.0 Injection St...		0%	1.8MB	0MB/s	0Mbps
Image SAFER 5.0 Session Ma...		0%	1.3MB	0MB/s	0Mbps
Intel(R) Dynamic Application ...		0%	1.0MB	0MB/s	0Mbps

스레드(Thread)란?

- 정의

프로세스 내에서 실행되는

실행 흐름의 단위

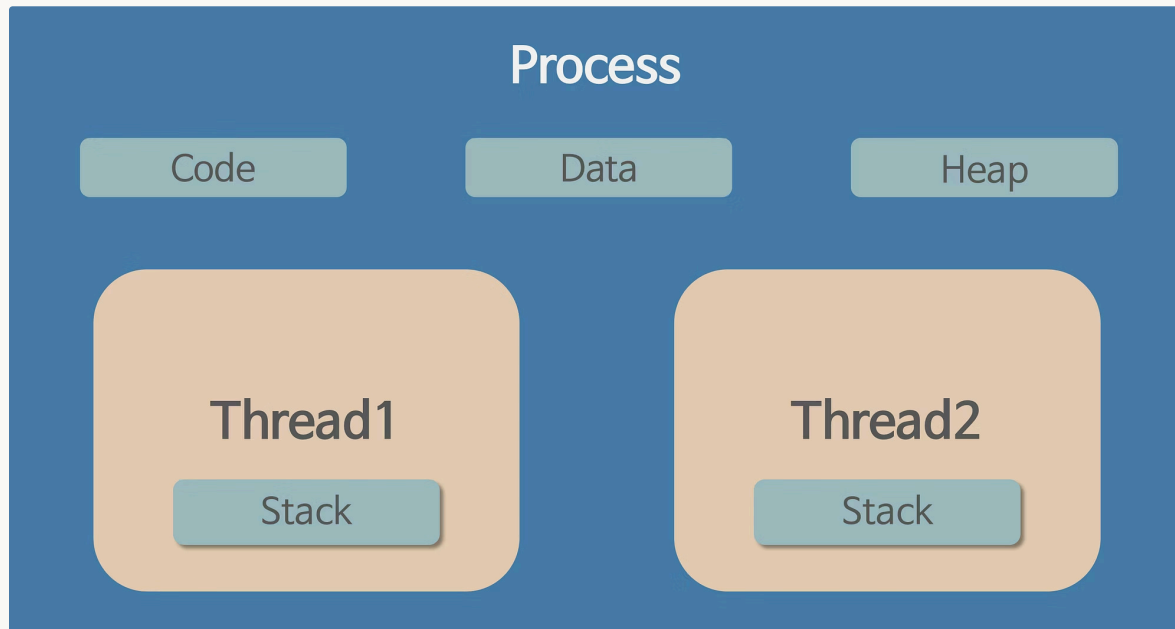
- 구성요소: Set of Register and Stack

현재 실행중인 상태의 레지스터와 스택으로 구성

- 자원 공유

프로세스의 자원(메모리)을 공유

Code, Data, Heap 영역을 함께 사용



Context Switch

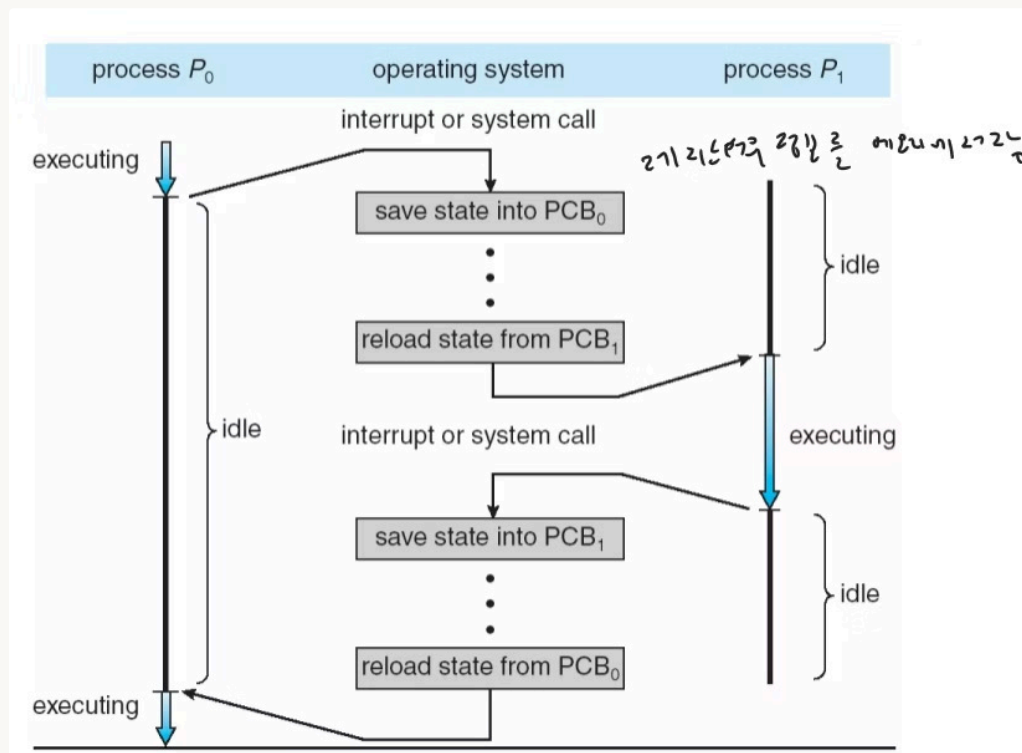
CPU는 한정되어있고 해야 할 일은 하나가 아니다.

프로세스간 전환이 필요한 순간 → context switch가 필요하다

- 일반적으로는 프로세스간 전환을 의미한다.
- 지금까지의 설명을 들었다면 Thread switching이라 불러야하지 않을까?

Context Switch의 경우의 수

- 다른 process 간의 전환
- 같은 process 내 전환



Context Switch(2)

컨텍스트 스위치 할 때..

1. 레지스터 저장 및 복원 (PCB)

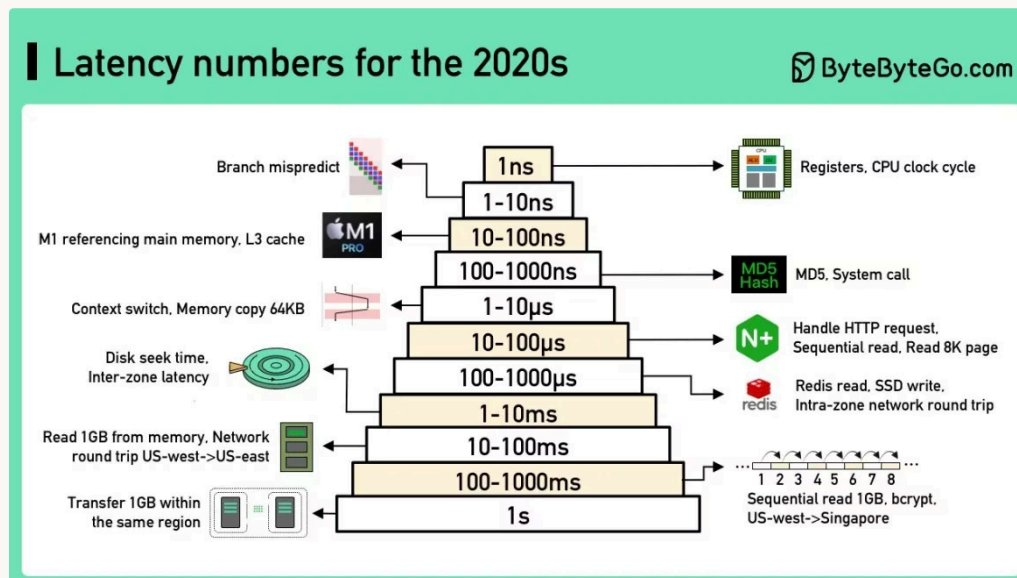
2. (프로세스 schedule)

3. (캐시, TLB 무효화)

= 오버헤드

다른 프로세스간 전환(Address Space의 변경)에 따라 오버헤드가 더 커진다

특히 캐시 미스가 매우 큼 (원래 90%이상 HIT)



멀티 프로세스 VS 멀티스레드 ?

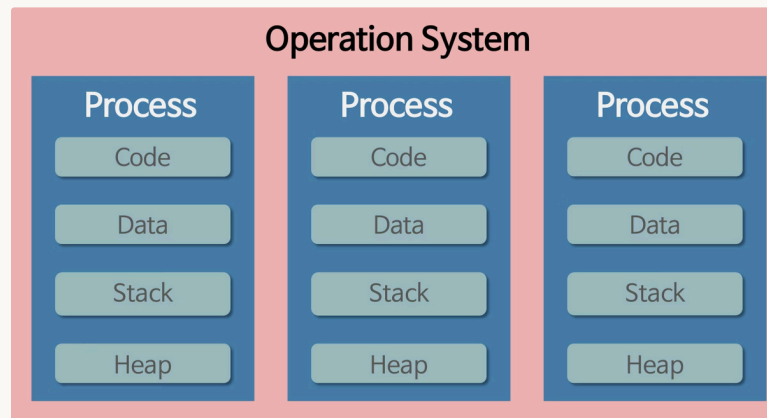
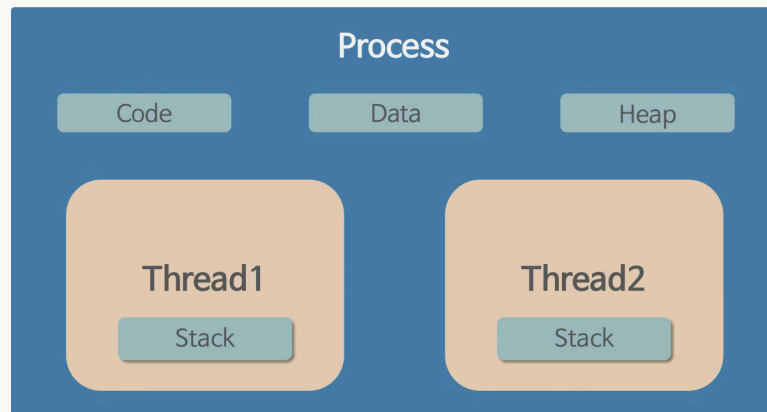
→(단일 스레드의) 프로세스 여러 개 VS (프로세스 하나에) 스레드 여러 개

프로세스 (Process)

- 메모리: 독립적 (완전히 분리된 메모리 공간)
- 자원 공유: 기본적으로 공유하지 않음
- 통신 방식: IPC(Inter-Process Communication) 필요

스레드 (Thread)

- 메모리: 공유 (Code, Data, Heap 영역)
- 자원 공유: 대부분의 자원 공유 (Stack 제외)
- 통신 방식: 메모리를 직접 공유하여 통신



멀티 프로세스 VS 멀티스레드 (2)

Context Switch 관점에서

프로세스: 높은 비용

- 스케줄링
- 캐시 비용

스레드: 낮은 비용

공유 자원을 제외한 상태(레지스터)
만 변경

안정성 측면에서

프로세스: 높은 안정성

한 프로세스에서 발생한 오류가 다른
프로세스에 영향을 주지 않는다.

스레드: 낮은 안정성

한 스레드에서 발생한 오류(예:
Segmentation Fault)가 프로세스
전체를 종료시킬 수 있다.

메모리 관점

프로세스: 높은 메모리

스레드: 낮은 메모리

멀티스레드가 성능이 무조건 좋을까?

Q&A
