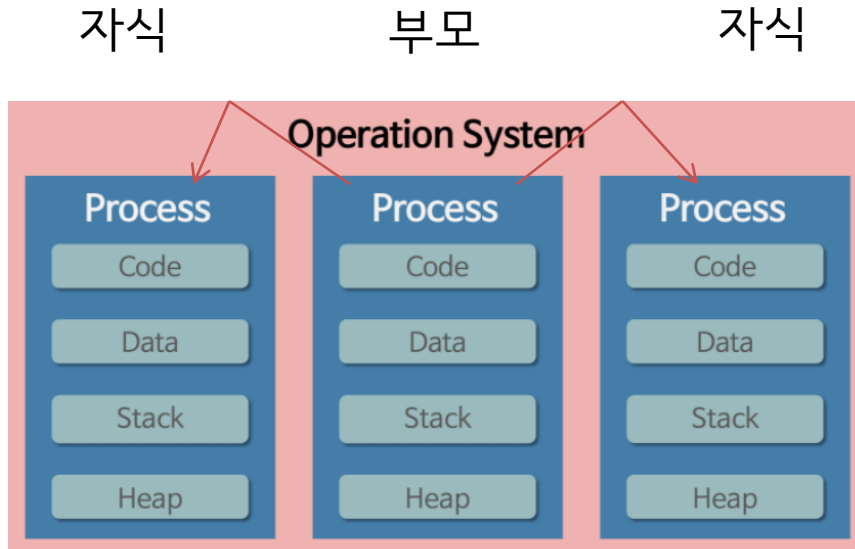


Multiprocess Vs MultiThread

Multiprocess

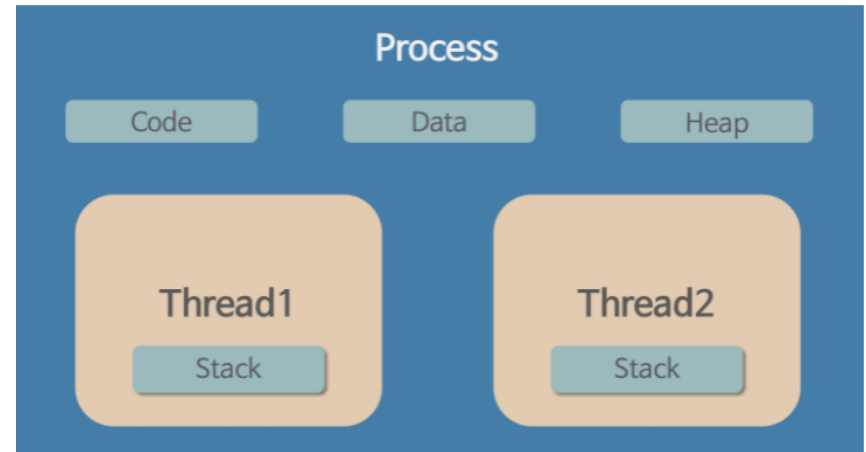
ex)어플의 처리
프로세스는 사용자가 로그인 시 한번에 한 승인만 가능 다중처리x
따라서 부모프로세스가 자식 프로세스를 fork해 처리
이에 따른 자식 프로세스의 메모리가 추가적으로 할당필요



MultiThread

ex)채팅어플로 사진을 전송함과 동시에 채팅이 가능하다.

한 어플에서 실행단위가 나누어져있기때문이다.
즉 한 프로세스 내에서 구분된 실행단위 프로세스가 다수 스레드로 구분되어있지않다면 단일 스레드로 실행되는 것이다.



특징

M-프로세스

각 프로세스는 독립적, IPC로 통신
자원 소모적 개별 메모리 차지
Context Switching 비용이 큼
동기 작업의 필요가 없다.

M-스레드

Thread끼리 긴밀하게 연결
공유된 자원을 비용절감
Context Switching 비용이 적다.
공유자원관리 이해가 필요하다.

M-프로세스가 좋지않아보이지만 장단점이 있는것이다.

예를 들면 M-스레드를 사용하는 인터넷익스플로러를 쓰다가 오류가 로딩바가 나오면 전체 창이 사라진다. 즉 상호 밀접관계에 있어 한 부분에서 오류가 발생하면 전체에 오류가 발생한다.

하지만 크롬의 경우 M-프로세스로 메모리 소비는 크지만 탭간의 관계가 적어 fragmentation등의 문제가 없다.

정리

1. 프로세스는 프로그램이 실행되는것 (cpu작업의 추상화)
2. 스레드는 한 프로세스 내에서 나뉜진 하나 이상의 실행단위
3. 한어플리케이션에 대한 작업을 동시에 하기 위해서는 2가지 처리방식이 필요하다.
(MT,MP)
4. 동시 처리로 보이 실행단위는 시분할 cpu점유(Context Switching)
5. 멀티프로세스는 독립적인 각각의 메모리 MT는 자원을 공유

추가 +

멀티 코어는 하드웨어 측면에서 실행단위를 병렬적으로 처리할 수 있도록 여러 프로세스가 있는것. (Parallelism)