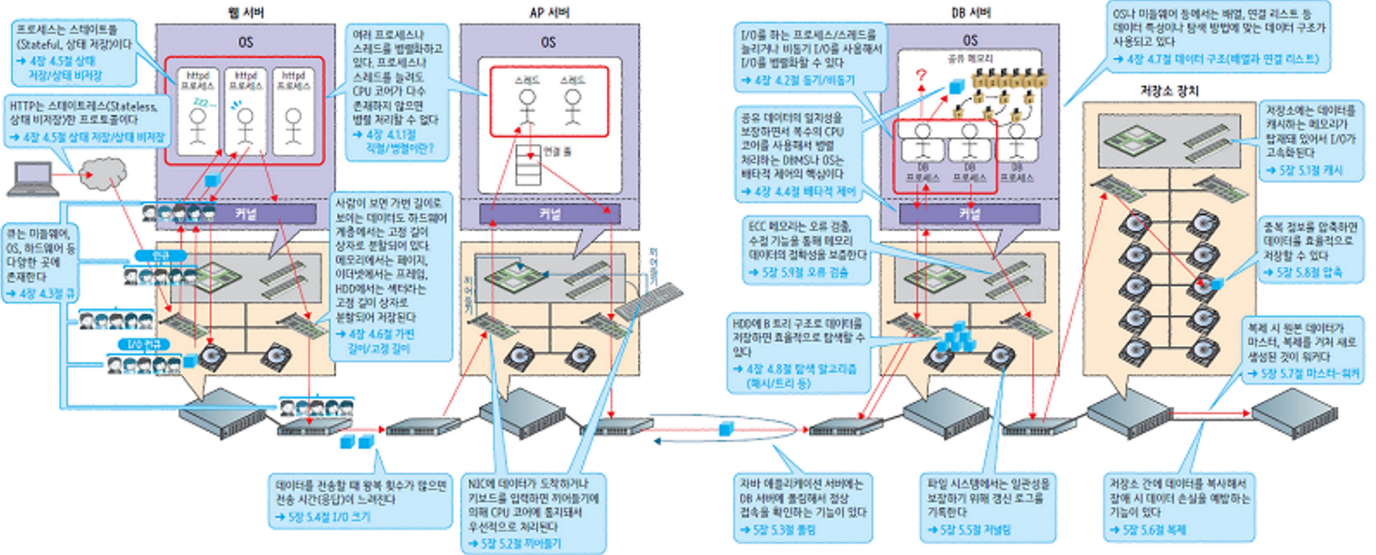


제3장 3계층형 시스템을 살펴보자

3계층형 시스템의 데이터 흐름과 각각을 지탱하는 기술

IT 시스템을 구성하는 컴포넌트들과 그 관계를 이해하기 쉽게 하나의 그림으로 정리했습니다. 3장에서 설명하는 3계층형 시스템의 데이터 흐름과 4~5장에서 설명하는 인프라를 지탱하는 개념이나 구조를 한눈에 볼 수 있습니다. 본문을 읽을 때도 필요에 따라 이 그림을 참고하기 바랍니다.



3.1 3계층형 시스템의 구성도

주요 구성요소

- 웹 서버
- AP 서버
- DB 서버

3.2 주요 개념 설명 (프로세스 / 스레드 / 커널)

*프로세스

프로세스 및 스레드 = OS상에서 실행돼서 어느정도 독립성을 가지고 동작하는 것

활동하려면 메모리 공간이 필요

이것을 커널에 의해 메모리상에 확보됨

스레드끼리는 메모리 공간을 공유

커널이 프로세스 및 스레드의 활동을 위한 메모리 공간을 확보

	프로세스	스레드
장점	개별 처리 독립성 높다	생성 시 부하가 낮다
단점	생성 시 CPU 부하 높다	메모리공간을 공유 (의도치 않는 데이터 읽기/쓰기 발생가능)

오라클DB에서는 여러 프로세스가 '공유 메모리 공간'을 이용할 수 있게 되어 있음

*커널

커널 = OS의 본질, OS의 인프라

뒤에서 무슨 일이 벌어지는지 은폐하면서, 편리한 인터페이스 제공하는 것

[커널의 역할 6가지]

1. 시스템 콜 인터페이스

애플리케이션이 OS를 통한 처리를 하고 싶을때 시스템 콜이라는 명령을 이용해 커널에 명령을 내림
프로세스는 물리적 장치에서 어떤 작업이 일어나는지 신경 쓸 필요 없음

2. 프로세스 관리

언제 어떤 프로세스가 어느 CPU코어를 이용할 것인지, 처리 우선순위를 어떻게 할 것인지를 결정
OS에서 가장 중요한 기능

3. 메모리 관리

물리메모리 공간의 최대치를 고려하여 관리
프로세스를 위한 공간 확보
메모리간의 독립성 관리

4. 네트워크 스택

5. 파일 시스템 관리

프로세스는 모든 것을 파일 단위로 생각
파일 시스템 = 물리 디스크에 제공된 데이터를 관리하는 기능
파일 시스템 덕분에 '파일'이라는 단위로 데이터를 작성하거나 삭제 가능
파일 시스템용 인터페이스를 제공

- 디렉터리 구조 제공, 액세스 관리, 고속화, 안정성 향상

6. 장치 드라이버

디스크, NIC 등의 물리장치용 인터페이스 제공
장치 드라이버를 통해 다양한 드라이버 인터페이스를 제공
각 장치사가 OS에 대응하는 장치드라이버를 제공해 해당 OS에 맞게 커널을 경유해 이용

3.3 웹 데이터 흐름

*데이터 흐름 공통점 정리

- 프로세스나 스레드가 요청을 받는다.
- 도착한 요청을 파악해서 필요에 따라 별도 서버로 요청을 보낸다.
- 도착한 요청에 대해 응답한다.

3.3.1. 클라이언트 PC부터 웹 서버까지

전체 흐름

1. 웹 브라우저 요청을 발행한다.
2. 이름 해석을 한다.
3. 웹 서버가 요청을 접수한다.
4. 웹 서버가 정적 콘텐츠인지 동적 콘텐츠인지 판단한다.
5. 필요한 경로로 데이터를 액세스 한다.

- 1) 웹 브라우저가 특정 인터넷 사이트로 요청을 보냄
- 2) 사이트의 이름을 통해, 해당 사이트가 어디있는지 해석 후, 이 결과를 해당하는 웹서버에 요청
- 3) 웹서버의 'httpd' 프로세스가 요청을 접수
- 4) httpd프로세스는 요청을 분석하여
 - A. 디스크에서 읽을지(정적) B. AP서버에 요청할지(동적) 판단
- 5-A) 디스크에서 읽을 경우 : 커널의 시스템콜로 실행. 커널을 통해 물리디스크의 내용을 취득
- 5-B) AP서버에 요청할 경우 : OS의 시스템콜로 실행. 커널을 통해 네트워크 통신이 요청됨

3.3.2. 웹 서버부터 AP 서버까지

1. 웹 서버로부터 요청을 도착한다.

2. 스레드가 요청을 받으면 자신이 계산할 수 있는지, 아니면 DB 접속이 필요한지를 판단한다.
3. DB 접속이 필요하면 연결 풀에 액세스한다.
4. DB 서버에 요청을 던진다.

3.3.3. AP 서버부터 DB 서버까지

1. AP 서버로부터 요청이 도착한다.
2. 프로세스가 요청을 접수하고 캐시가 존재하는지 확인한다.
3. 캐시에 없으면 디스크에 액세스 한다.
4. 디스크가 데이터를 반환한다.
5. 데이터를 캐시 형태로 저장한다.
6. 결과를 AP 서버에 반환한다.

3.3.4. AP 서버부터 웹 서버까지

1. DB 서버로부터 데이터가 도착한다.
2. 스레드가 데이터를 가지고 계산 등을 한 후에 파일 데이터를 생성한다.
3. 결과를 웹 서버로 반환한다.

3.3.5. 웹 서버부터 클라이언트 PC까지

1. AP 서버로부터 데이터가 도착한다.
2. 프로세스는 받은 데이터를 그대로 반환한다.
3. 결과가 웹 브라우저로 반환 되고 화면에 표시된다.

3.4 가상화

클라우드 환경은 가상화 기술 기반으로 구축

*가상화 = 컴퓨터 시스템에서 물리 리소스(서버, 네트워크, 저장소)를 추상화하는 것

OS도 가상화 기술의 하나

- OS의 커널에 의해 하드웨어가 추상화되면서, 하드웨어 의식X 컴퓨터에 연결된 기억장치나 네트워크 통한 데이터 교환 이루어짐
- 가상 메모리 사용해 프로세스 및 OS 커널의 메모리 공간을 분리 -> 하나의 프로그램이 실패해도 시스템 전체 영향X

*가상머신

1. 호스트OS형 : 소프트웨어를 에뮬레이터하는것. 성능면에서 제한 존재
2. 하이퍼바이저형 : 하드웨어상에서 직접 가상화 소프트웨어 실행하고, 그 위에 가상머신 동작. 서버 가상화 대표기술
 - a. 완전 가상화
 - b. 준 가상화 : 완전 가상화의 성능 저하 문제 해결

*컨테이너 = 리소스가 격리된 프로세스

- 하나의 OS상에서 여러 개 동시에 가동
- 각각 독립된 루트 파일시스템, CPU/메모리, 프로세스 공간 등 사용 가능 (VM과의 차이점)

*도커

- 애플리케이션 실행 환경을 자동 구축해주는 '도커 이미지'
- 도커 허브 = 도커 이미지를 공유할 수 있는 레지스트리

*클라우드와 가상화 기술

- AWS
- GCP
- Azure

참고

[그림으로 공부하는 IT 인프라 구조](#)