

## 1장. 시스템과 인프라 기초 지식

✓ Docker를 도입하기 앞서 알아둬야 할 시스템 기반의 개요와 시스템 기반을 다루기 위해 필요한 인프라 기술 학습

→ 개발한 app을 release하여 최종 사용자가 이용할 수 있도록 하려면 시스템 기반을 구축하고 그 위에 application의 실행 환경을 마련해야 함.

· 시스템 기반? application을 가동시키기 위해 필요한 Hardware, OS/middleware 등과 같은 infra.

· 미들웨어 (middleware)? 공통 서비스 및 기능을 app에 제공하는 software.

= 양쪽을 연결하여 data를 주고 받을 수 있도록 중간 (middle)에서 매개 역할을 하는 software.

· Docker? application 실행 환경을 작성·관리하기 위한 platform

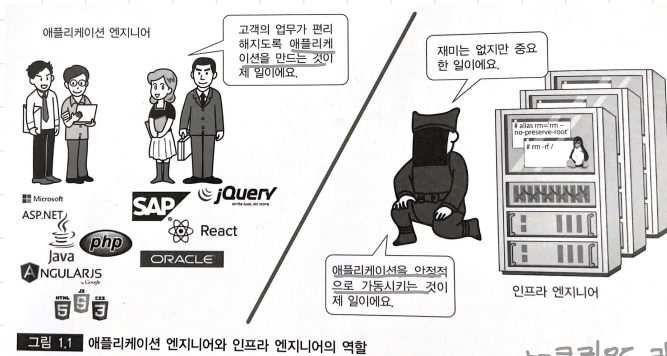
### 1.1. 시스템 기반의 기초 지식

\* 폭포형 모델을 사용한 시스템 개발

[일반적인 역할 분담]

[application 실행 환경의 구축] → ("네트워크나 하드웨어에 능통한") 인프라 엔지니어 담당.

[application 개발] → ("업무 지식 및 프로그래밍", "테스트 방법과 같은 기술을 잘 아는") application 엔지니어 담당



[장점] 기업의 비즈니스 정보를 보안성 높게 관리.

[단점] 시스템 구축하는데 있어서 시간상 비용상

→ 클라우드 컴퓨팅 기술이 나오기 이전의 기업 인프라 구축의 일반적인 방식.

[현재] 클라우드 등장으로 흐름 변화

· 자사에서 데이터센터/기계를 보유

→ 온프레미스 환경에서 가동시켰던 서버들을, 클라우드 상의 가상 인스턴스로 옮김.

· 데이터베이스나 네트워크와 같은 클라우드 서비스를 이용

→ 실행 환경의 구축 범위가 극도로 감소.

→ 짧은 사이클로 리소스를 확보하는 스타일로 변화

온프레미스 (on-premise) ← 반대 → 오프프레미스 (off-premise)

↓  
클라우드 방식

· 그래서 기업에서 <보안성 높은 데이터 → 온프레미스 환경  
" 낮은 " → 클라우드 환경

Cloud? 네트워크 전반에서

확장 가능한 리소스를 추상화, 풀링, 공유하는 IT환경

→ application이 실행되는 환경.

[서버/스토리지 등의 자원을 미리 확보하고 이를 사용자 요청에 따라 제공한다는 개념.]

(1대의 물리 host 상에서 움직이는 시스템과는 달리)

· 클라우드를 구성하는 대부분의 기술: 분산환경에서 가동시키는 것이 기본.

· 분산환경 → 인프라 엔지니어가 수동으로 operation(운영)을 하지 않음.

→ 자동화된 툴을 사용하여 orchestration(오케스트레이션) 함.

(여러 대의 서버에 각 컨테이너를 적절히 배치하고 관리하는 것.)

① Pod(포드)? 여러 컨테이너를 묶는 것.

· 인프라엔지니어 → 인프라 기술 + (application 엔지니어 업무) 코드 작성 스킬 요구.

✓ 여기서는 application 엔지니어가 Docker를 사용하기 위해 필요한 인프라 기술의 기초에 대해 설명

· application 엔지니어 → (인프라 엔지니어 업무) 제품 환경에 대한 배포 테스트 스킬 요구,  
↳ OS(커널)나 네트워크 등과 같은 인프라 기술의 기초지식 필요.

시스템 기반의 구성 요소.

· 시스템 기반? application을 가동시키기 위해 필요한 인프라 (하드웨어나 OS, 미들웨어)

<앞으로 책에서는> 시스템 기반을 다루는 기술 = 인프라 기술이라고 칭함.

· 시스템에 요구되는 사항 (요구사항)

\* 기능 요구사항 (functional requirement)

: 시스템의 기능으로서 요구되는 사항.

→ 시스템이나 소프트웨어에서 무엇을 할 수 있는지를 모아놓은 것.

\* 비기능 요구사항 (non-functional requirement)

: 시스템의 성능이나 신뢰성, 확장성, 운용성, 보안 등과 같은 요구사항.

: 기능 요구사항이외의 모든 요구사항.

→ 비기능 요구사항을 충족시키려면 프로그래밍 지식 + 시스템 기반에 관한 지식 필요.

# 시스템 기반의 관점에서 시스템 전체 구성

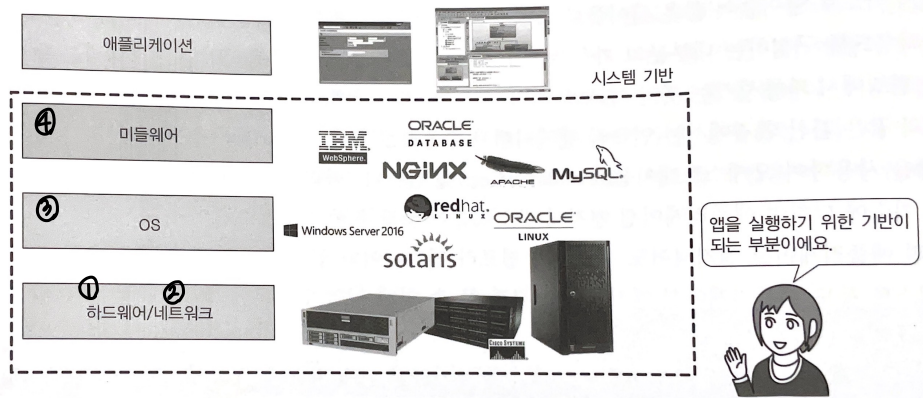


그림 1.2 시스템 기반의 구성 요소

## ① 하드웨어 : 시스템 기반을 구성하는 물리적인 요소.

→ 서버 장비 본체, 데이터 저장하기 위한 스토리지, 전원 장치, ...  
 (넓은 의미) 하드웨어들을 설치하는 데이터센터의 설비도 포함.  
 ~ 건물, 공조 보안 설비, ...

## ② 네트워크 : 시스템 이용자가 원격지에서 액세스할 수 있도록 서버들을 연결하기 위한 요구사항.

→ 네트워크 장비 (루터, 스위치, 방화벽, ...), 네트워크 장비들을 연결하기 위한 케이블 배선도 관리, ...

· 사용자가 이용하는 클라이언트 단말기에서 무선 LAN으로 연결하는 경우에는 액세스 포인트 등도 필요.

## ③ OS (운영체제) : 하드웨어·네트워크 장비를 제어하기 위한 기본 소프트웨어

→ 하드웨어의 리소스, 프로세스 관리.

· 클라이언트 OS : Windows / macOS / ...

→ 클라이언트 OS에는 이용자가 사용하기 쉽도록 GUI 기능이나 멀티미디어 기능이 파생되어 있음.

· 서버 OS : Windows Server, Unix, Linux, ...

→ 서버 OS는 시스템을 고독 및 안정적으로 가동시키기 위해 필요한 기능으로 특화되어 있음.

→ 하드웨어의 성능을 최대한 끌어낼 수 있도록 만들어져 있음.

→ 장시간 가동해도 안정적으로 작동할 수 있도록 만들어져 있음.

→ 대량의 데이터를 효율적으로 수행하는 장치도 갖추고 있음.

## ④ 미들웨어 : 서버 OS 상에서 서버가 특정 역할을 다하기 위한 기능을 갖고 있는 소프트웨어.

- 상용된 것
- 오픈소스로 된 것
- 오래 역사가 있어서 안정적인 것.
- 현재 활발히 개발되고 있는 것.
- ...

→ 오픈소스 미들웨어는 누구든지 개발에 참여 가능.

→ 시스템 기반을 설계할 때는 어떤 미들웨어를 사용할 지 선택하는 것이 중요.