

클라우드 네이티브 무상 컨설팅  
찾아가는 클라우드 네이티브 세미나



# 클라우드 네이티브 무상 컨설팅 찾아가는 클라우드 네이티브 세미나

---

## 공공부문 클라우드 네이티브 개념

2023. 09. 01.



**4차 산업혁명 시대**를 맞아 국민의 일상에 **디지털 전환**이 가속화되고 있고,

이에 적합한 대국민 서비스 제공을 위해

**공공서비스의 클라우드 전환이 추진**되고 있습니다.

클라우드 전환 및 도입 효과를 높이기 위해

**단순한 기술 인프라 위주의 클라우드** 도입보다 클라우드

환경에 최적화된 새로운 형태의 **클라우드 네이티브 정보시스템 구축**이

필요합니다. 즉, 기존의 크고,

**단일한 서비스 구조를 마이크로서비스 아키텍처로 구현**하

여 개발, 배포, 운영함으로써 빠르고 안정적인 할 수 있습니다.

*클라우드 네이티브 정보시스템을 구축을 위한 발주자 안내서-  
행정안전부, 한국지능정보사회진흥원*





# CNCF Cloud Native Definition v1.0

클라우드 네이티브 기술을 사용하는 조직은 현대적인 퍼블릭, 프라이빗, 그리고 하이브리드 클라우드와 같이 동적인 환경에서 확장성 있는 애플리케이션을 만들고 운영할 수 있다.

컨테이너, 서비스 메시, 마이크로서비스, 불변의 인프라스트럭처, 그리고 선언적 API가 전형적인 접근 방식에 해당한다.

이 기술은 회복성이 있고, 관리 편의성을 제공하며, 가시성을 갖는 느슨하게 결합된 시스템을 가능하게 한다.

견고한 자동화와 함께 사용하면, 엔지니어는 영향이 큰 변경을 최소한의 노력으로 자주, 예측 가능하게 수행할 수 있다.

Cloud Native Computing Foundation은 **벤더 중립적인 오픈소스 프로젝트 생태계**를 육성하고 유지함으로써 해당 패러다임 채택을 촉진한다.

우리 재단은 최신 기술 수준의 패턴을 대중화하여 이런 혁신을 누구나 접근 가능하도록 한다.



# Cloud Native Computing Foundation

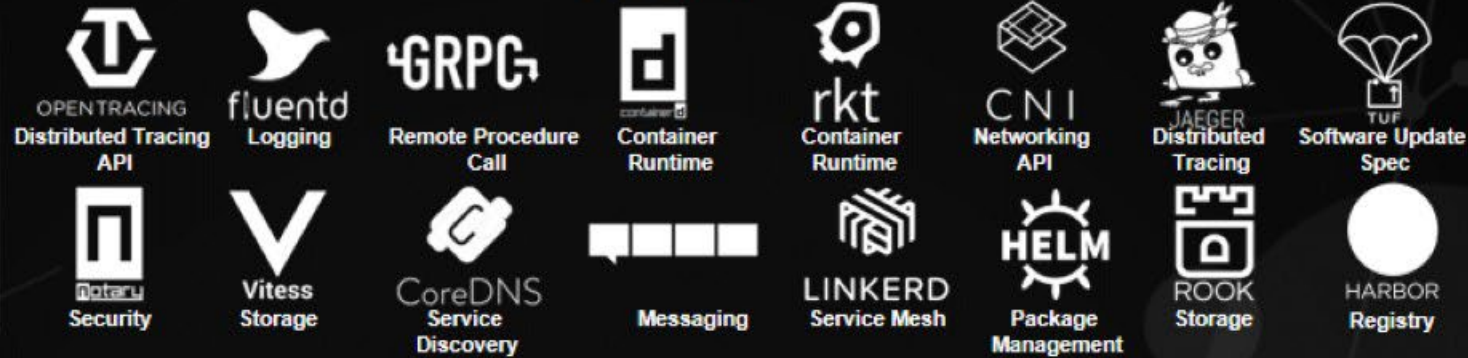


- Non-profit, part of the Linux Foundation; founded Dec 2015

## Graduated



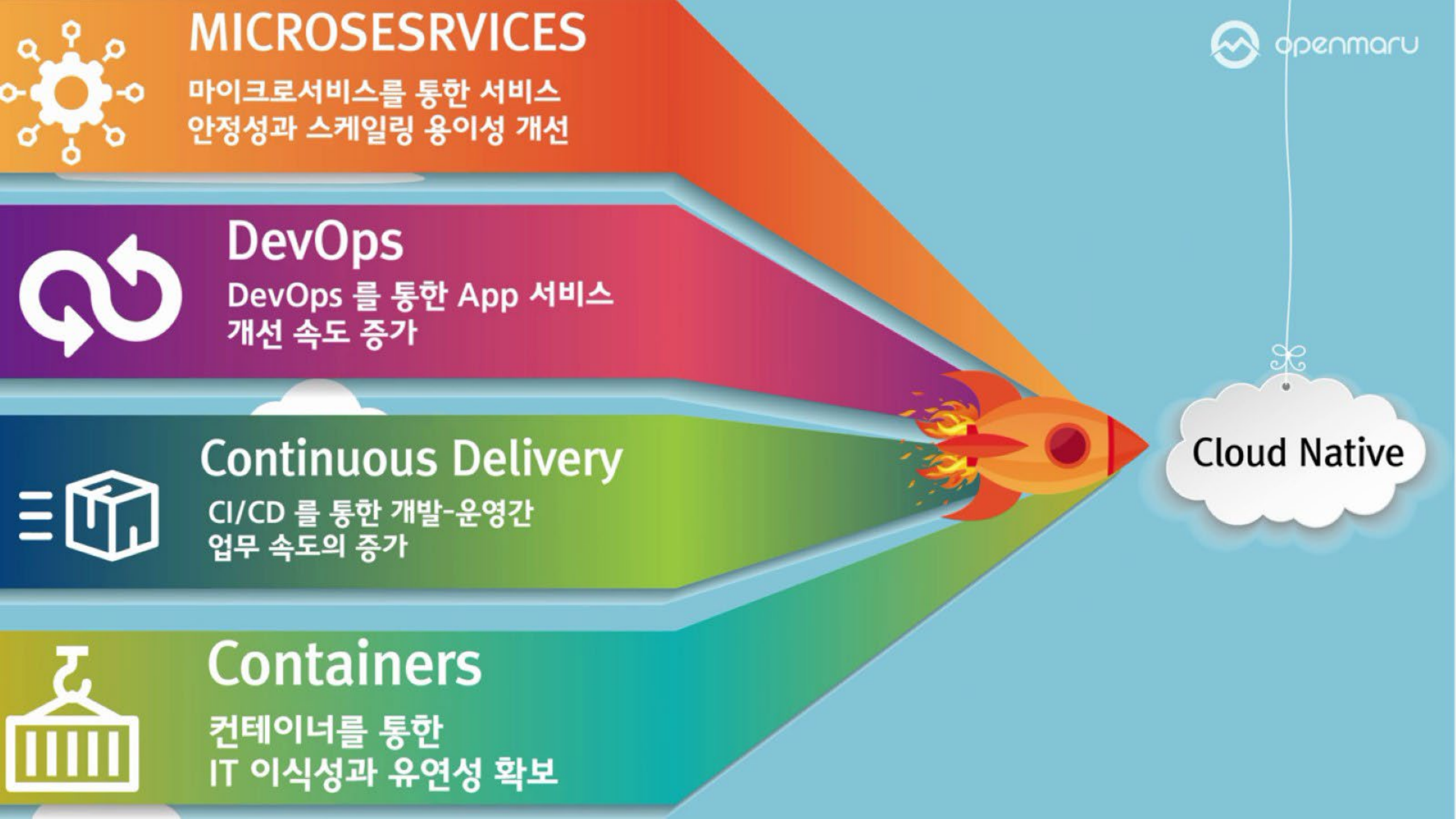
## Incubating



- Platinum members:







Cloud Native



컨테이너 기술



컨테이너가 도입되기  
전에는  
제품별로 배에 올리고  
내려야만 했습니다.



**Jasmine Kim**

Head of Container Department

# After Containerization

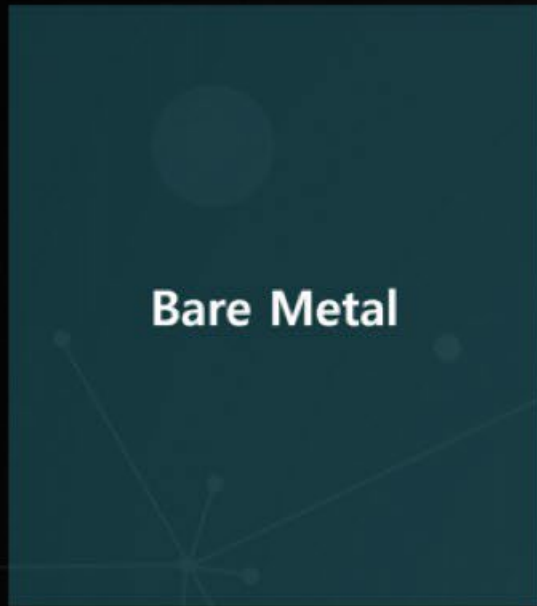
배에 컨테이너를 싣고

containerization

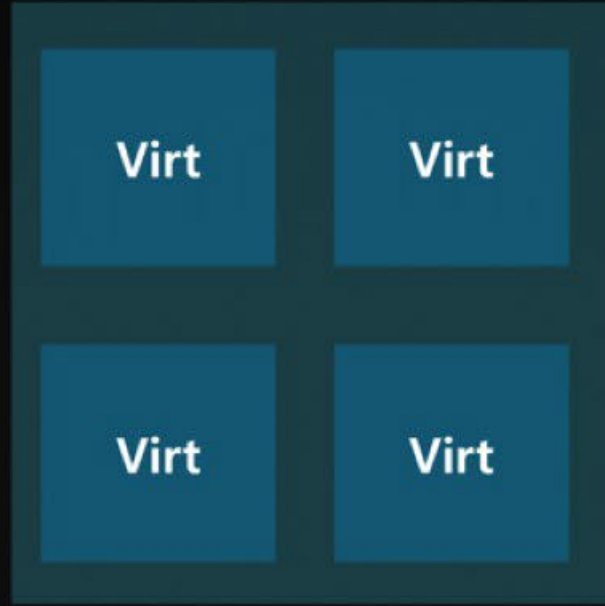




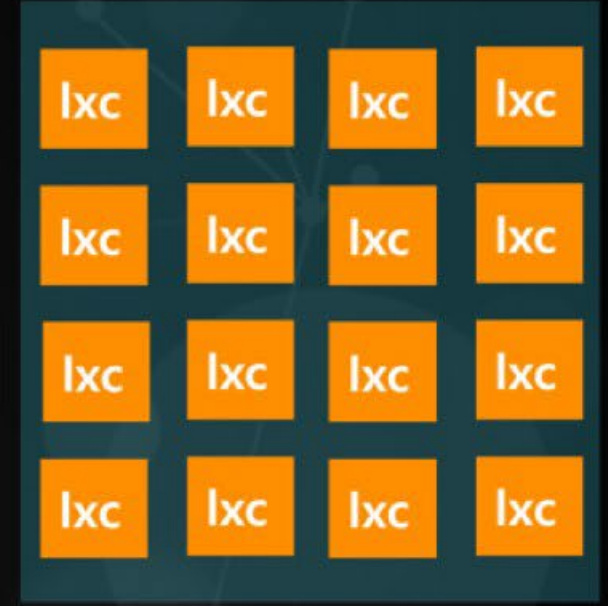
# Evolution of Infrastructure Architectures



Bare Metal



Virtualized

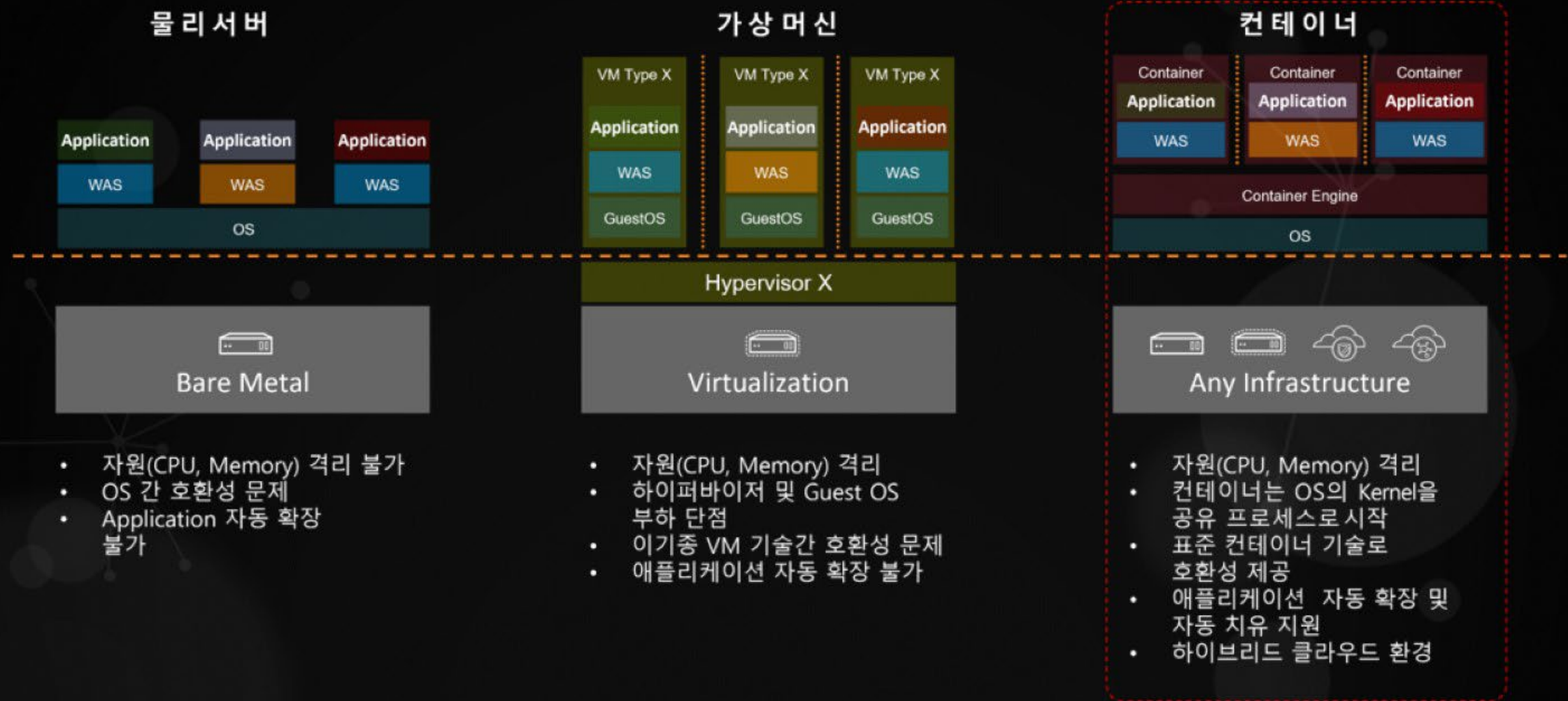


Containerized



# WHY CONTAINER ?

- 자원 효율성, 자원 격리, 호환성, Auto Scaling, DevOps, MSA, 관리 편의성



- 자원(CPU, Memory) 격리 불가
- OS 간 호환성 문제
- Application 자동 확장 불가

- 자원(CPU, Memory) 격리 하이퍼바이저 및 Guest OS 부하 단점
- 이기종 VM 기술간 호환성 문제
- 애플리케이션 자동 확장 불가

- 자원(CPU, Memory) 격리
- 컨테이너는 OS의 Kernel을 공유 프로세스로 시작
- 표준 컨테이너 기술로 호환성 제공
- 애플리케이션 자동 확장 및 자동 치유 지원
- 하이브리드 클라우드 환경



# 시작 시간 - Containers vs. VMs

- 하드웨어 가상화는 CPU, 메모리, 하드 디스크 등의 하드웨어를 가상화하고 있기 때문에 하드웨어 나 OS 부팅해야 부팅에 **분 단위 시간 소요**
- 컨테이너 형 가상화에서는 컨테이너 부팅 시 OS는 이미 시작하고 프로세스의 시작 만 할 **초 단위 시간**



# 오버헤드 - Containers vs. VMs

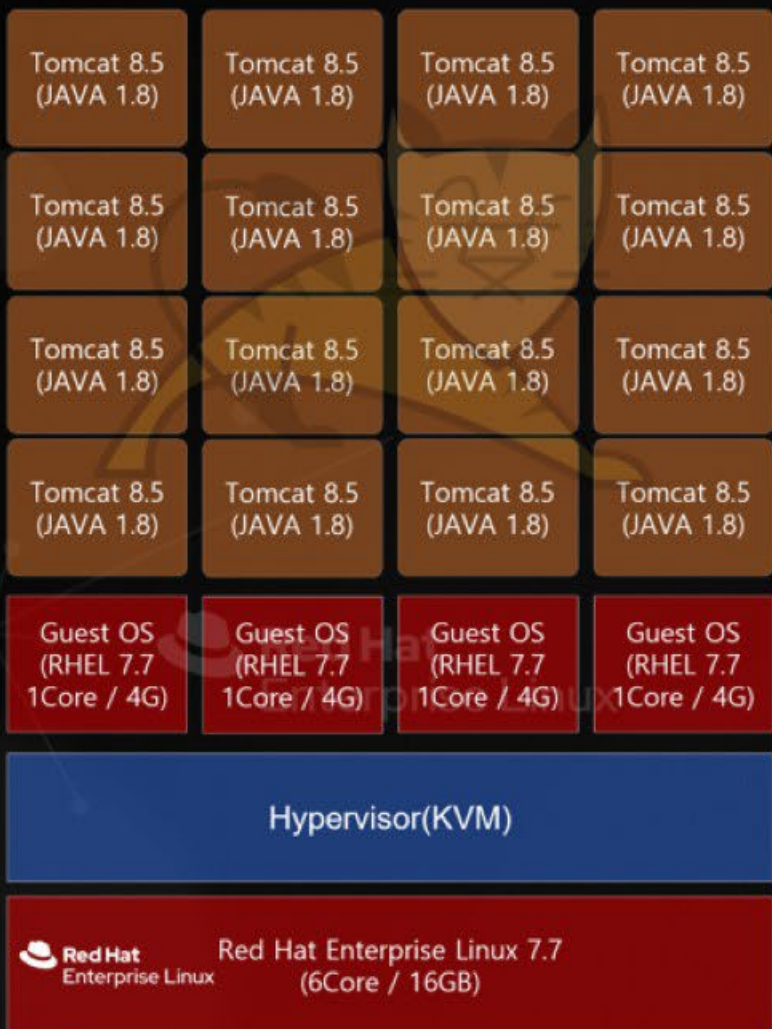
- OS에서 응용 프로그램을 작동하는 경우, 하드웨어 가상화에서는 **가상화 된 하드웨어 및 하이퍼바이저를 통해 처리하기 때문에 물리적 시스템보다 처리에 추가적인 시간 (오버 헤드)가 필요**
- 컨테이너 형 가상화 커널을 공유하고 **개별 프로세스가 작업을 하는 것과 같은 정도의 시간 밖에 걸리지 않기 때문에 대부분 오버 헤드가 없음**



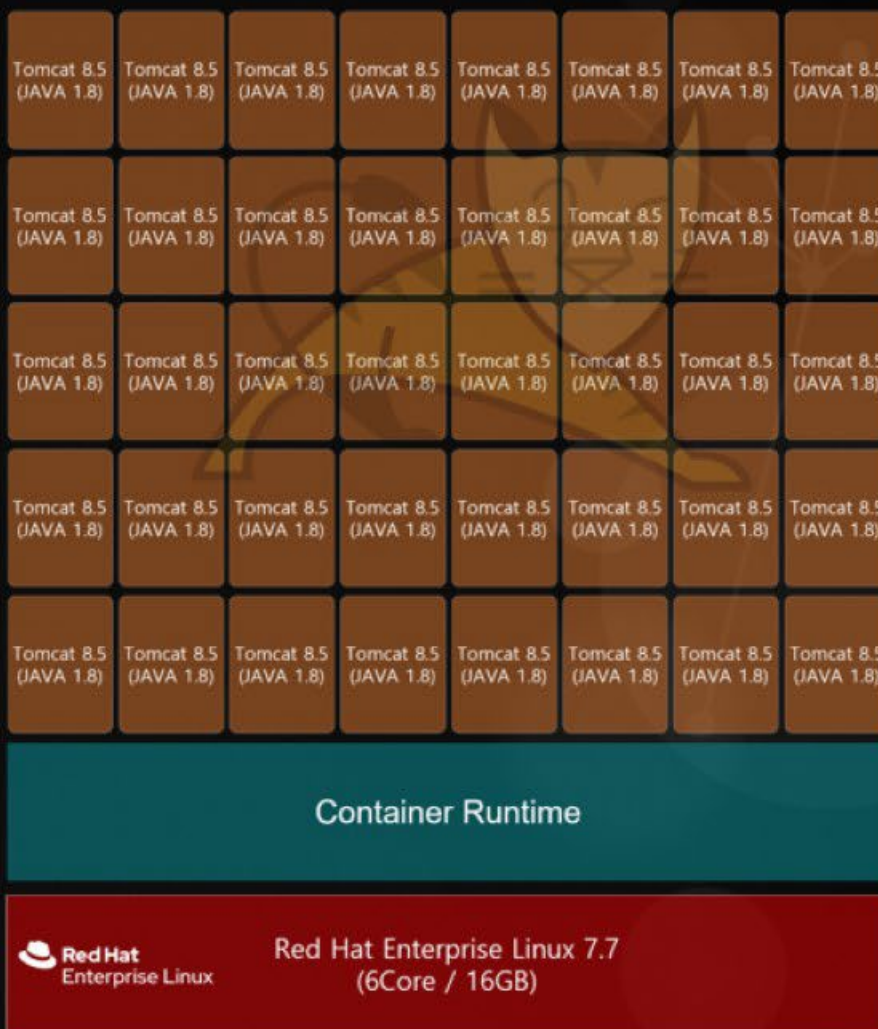


# 가상화와 컨테이너 비교 결과

## 가상화 - 16개 Tomcat 인스턴스



## 컨테이너 - 40개 Tomcat 인스턴스



# Pets vs Cattle

## Pets



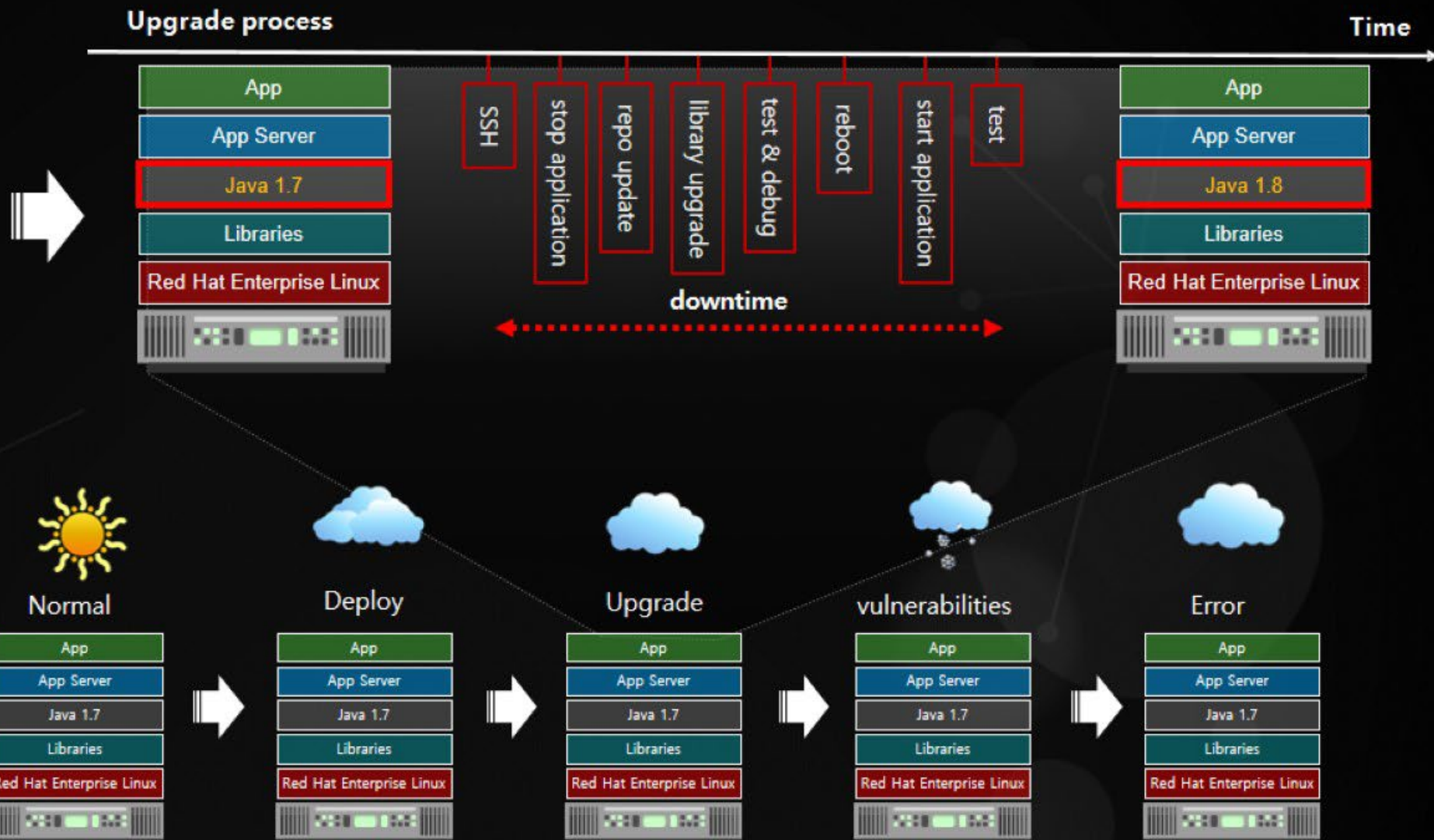
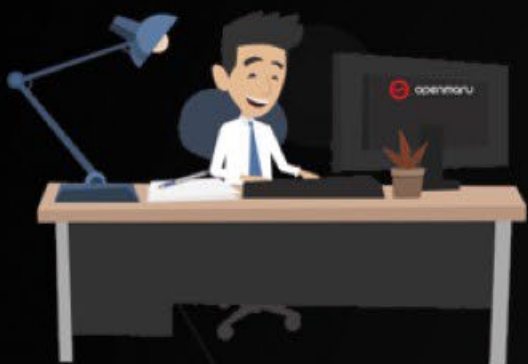
## Cattle



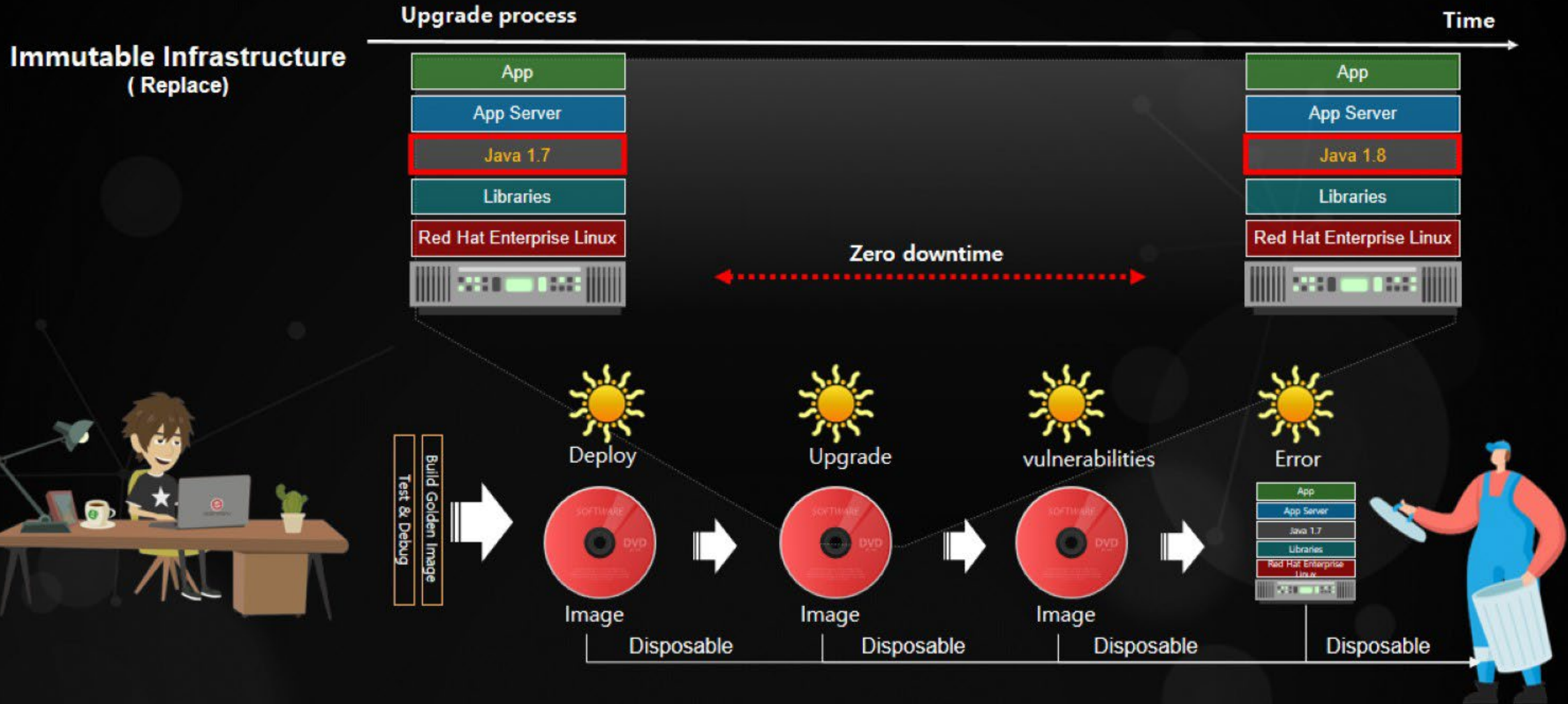


# In-place Server Upgrade

## Mutable Infrastructure ( In-Place )



# Replace Server Upgrade



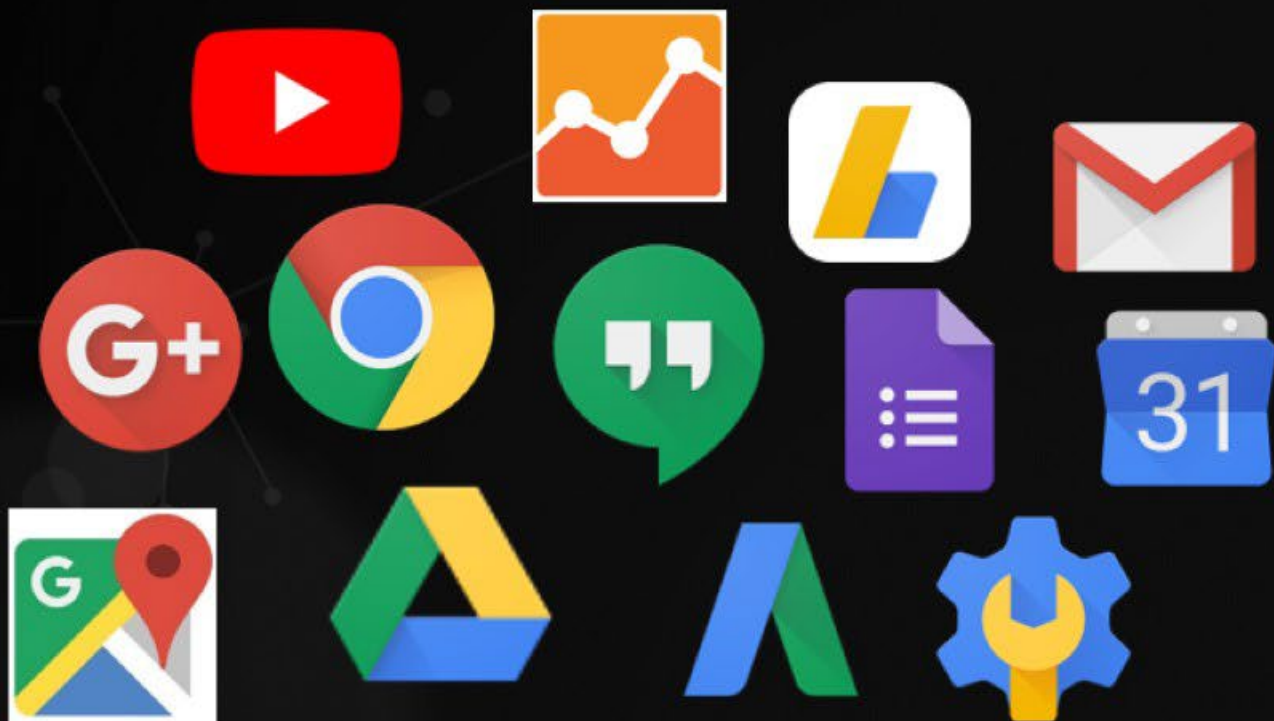


가상화 기술과 컨테이너 기술의 차이점

# 구글과 불변의 인프라스트럭처 & Kubernetes

# Google 는 모두 컨테이너에서 실행

- Gmail , 검색, 지도 ...
- MapReduce , GFS , Colossus ...
- Google Compute Engine 가상 머신도 컨테이너에서 실행!
- 매주 20 억개 이상의 컨테이너를 실행 중





# GOOGLE 과 컨테이너

- Google의 업무 방식

Gmail에서 YouTube, 검색에 이르기까지 Google의 모든 제품은 컨테이너에서 실행됩니다.

개발팀은 컨테이너화를 통해 더욱 신속하게 움직이고, 효율적으로 소프트웨어를 배포하며 전례 없는 수준의 확장성을 확보할 수 있게 되었습니다. Google은 매주 수십억 개가 넘는 컨테이너를 생성합니다. 지난 10여 년간 프로덕션 환경에서 컨테이너화된 워크로드를 실행하는 방법에 관해 많은 경험을 쌓으면서 Google은 커뮤니티에 계속 이 지식을 공유해 왔습니다.

초창기에 cgroup 기능을 Linux 커널에 제공한 것부터 내부 도구의 설계 소스를 Kubernetes 프로젝트로 공개한 것까지 공유의 사례는 다양합니다. 그리고 이 전문 지식을 Google Cloud Platform으로 구현하여 개발자와 크고 작은 규모의 회사가 최신의 컨테이너 혁신 기술을 쉽게 활용할 수 있도록 하였습니다.



# Kubernetes는

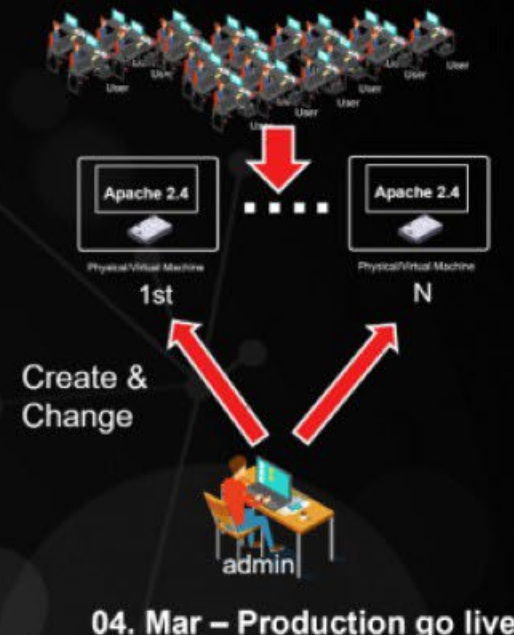
- **Kubnetes는 Open Source Software**
  - [github.com/kubernetes/kubernetes](https://github.com/kubernetes/kubernetes) (Apache License 2.0)
  - 오픈 디자인
    - [github.com/kubernetes/community](https://github.com/kubernetes/community)
  - 오픈 커뮤니티
    - Cloud Native Computing Foundation
    - Special Interest Groups (SIGs)
- **Kubernetes 는 Portability**
  - **cloud-controller-manager** (클라우드 작업)
    - L4 LB 블록 스토리지 등의 작업을 추상화
    - GCP, AWS, Azure, OpenStack 등
  - **CNI (Container Network Interface)**
    - 컨테이너 네트워크의 표준화
    - Calico, Flannel, Weave Net 등
  - **CRI (Container Runtime Interface)**
    - 컨테이너 런타임 표준
    - containerd , cri-o, rkt



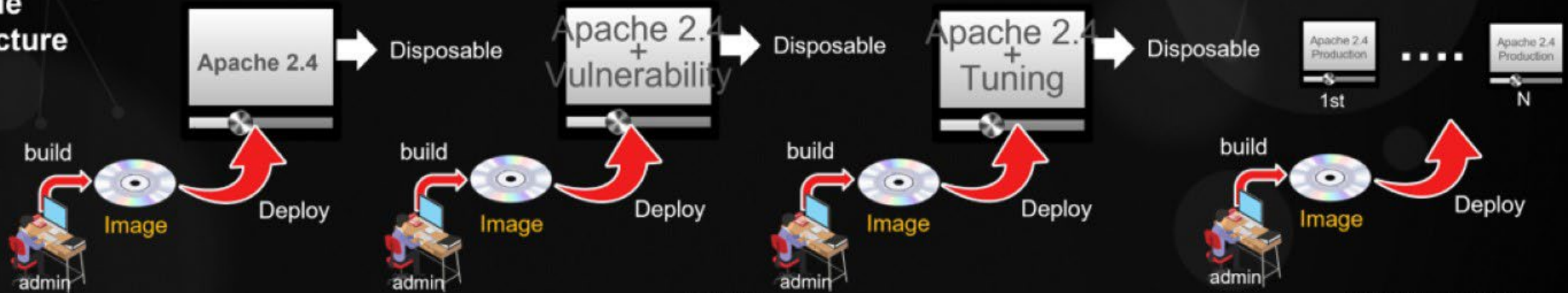


# 가변 vs. 불변 인프라스트럭처

## Mutable Infrastructure (In-Place)



## Immutable Infrastructure (Replace)



운영자님. 아직도 서버를 애지중지 키우시고 계신가요?





Application Performance Management

감사합니다.



openmaru  
APM





openmaru