

IT 운영자들이 반드시 알아야 하는 클라우드 네이티브 기술,  
운영 방법은 어떻게 변화할까요?

## 운영자가 알아야 할 것들 ????

기존 운영환경  
구성과의 차이점  
은 무엇인가요?

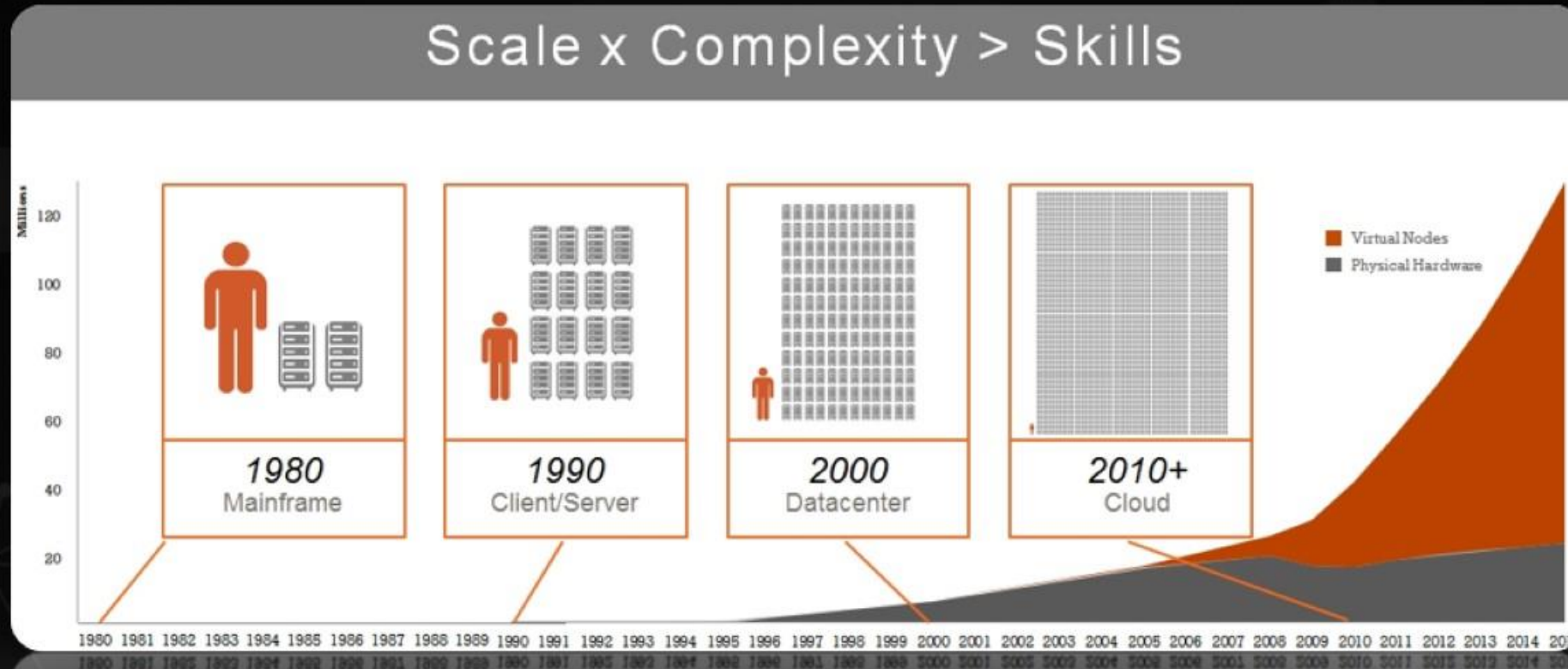
불변의 인프라스트럭처는  
어떤 개념인가요?

기존 환경 대비 컨테이너  
기술의 장점이 무엇이죠?

PaaS 환경에서 로그는  
어떻게 관리하나요?



# Increasing scale and complexity means we need admin automation



Opscode gets more venture dough for its Chef

From - <http://goo.gl/dLcjS>

# 시스템 운영의 현실

## 웹시스템의 요구사항과 특징

- Scale-out 형 인프라로 계층 형 아키텍처
- 가상화와 클라우드환경에 적합하며 인스턴스 개수가 많으며 노드 간 연결성이 높음
- 장애 민감하며 신속한 장애 복구와 재발 방지가 중요
- 설치되는 관련 소프트웨어가 많으며 환경 검증이 필요



## 시스템 운영 이슈

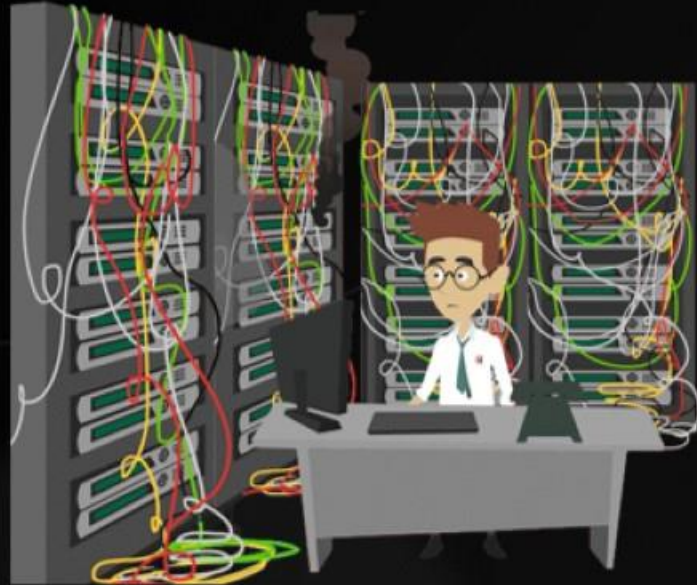
- 시스템 환경의 불일치(Dev/Stage/Prod, 서버별)
- 긴 배포 시간
- 수 작업으로 인한 Human Errors
- IT Agility 부족으로 인한 운영팀 축소
  - 개발팀에서 직접 IT 인프라 운영
- 문제점의 발견과 조치에 많은 시간 소요

서버 에러?

사용자 에러!



# 시스템 비대화로 작업 폭증과 인력 부족 어떻게 할까요?



컨테이너 기술의 차이점

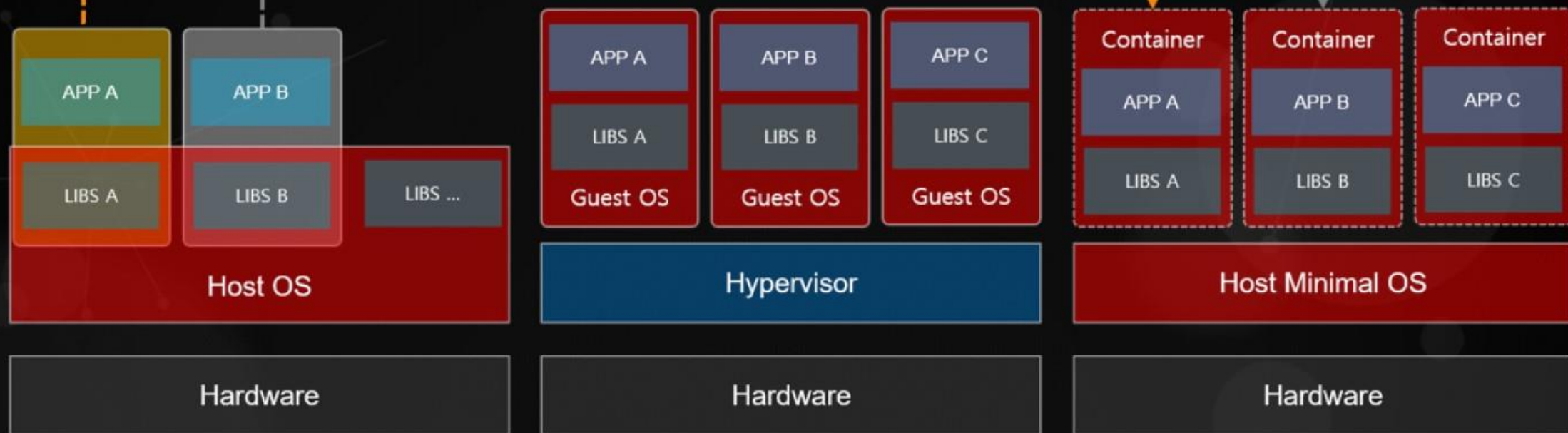
컨테이너 환경에서 인프라 운영의 변화

# Evolution

## Traditional *shared*

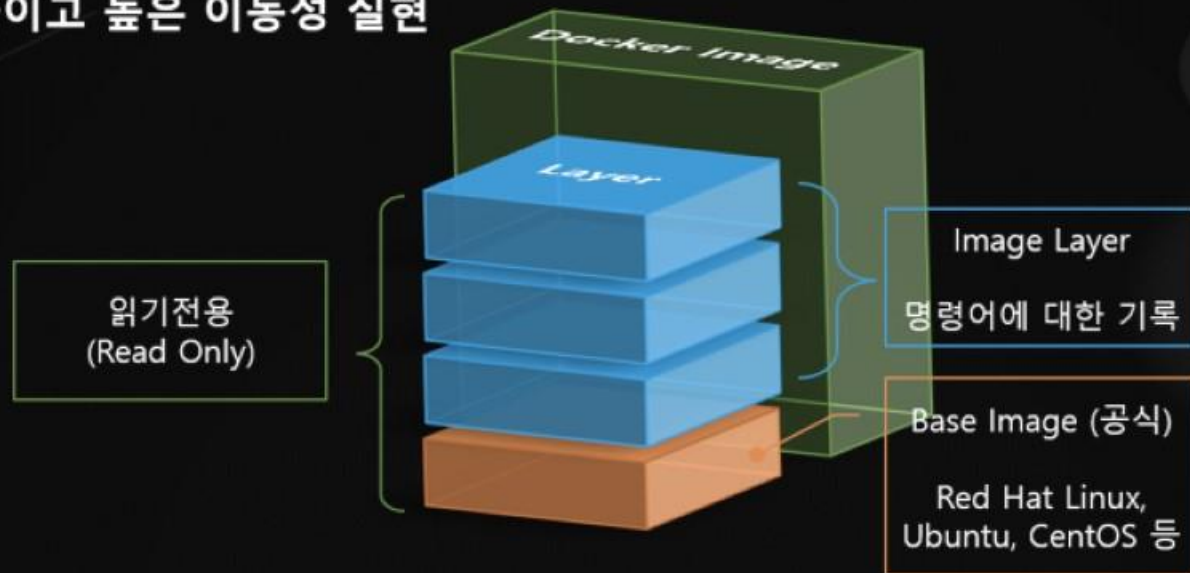
## Virtual *system isolation*

## Container *process isolation*



## 컨테이너 이미지란..?

- 컨테이너를 실행할 때 필요한 파일시스템
  - 이미지 레이어의 집합체
    - 파일 내용과 메타 정보를 포함
  - 레이어는 부모와 자식 관계
  - 변경분만 기록
  - Read Only (읽기 전용) 으로 쓸 수 없음
- 공통 레이어를 이미지 간에 공유
  - 디스크 용량을 줄이고 높은 이동성 실현







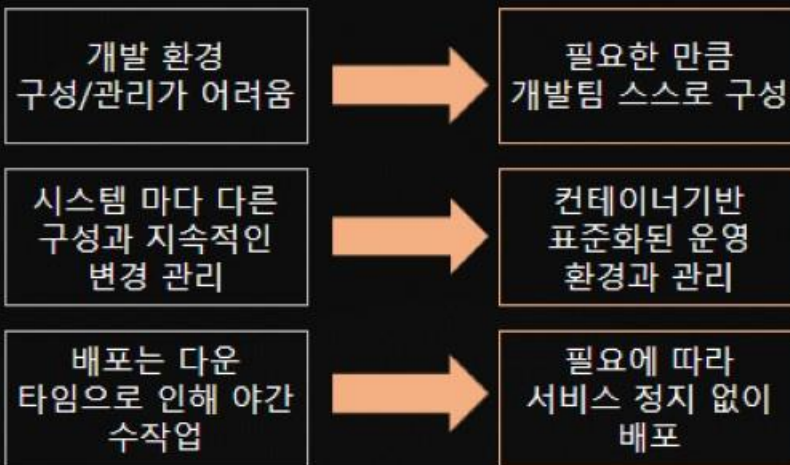
# 개발 환경 제공 - 기존 시스템별 vs. Kubernetes

- 시스템 마다 절차와 관리도구가 다르고 수작업에 의한 운영
- PaaS를 통한 개발팀 스스로 필요한 개발 환경 구축

## 업무 별로 분산된 개발 환경



## 컨테이너를 통한 표준화



## 개발 환경 자동화

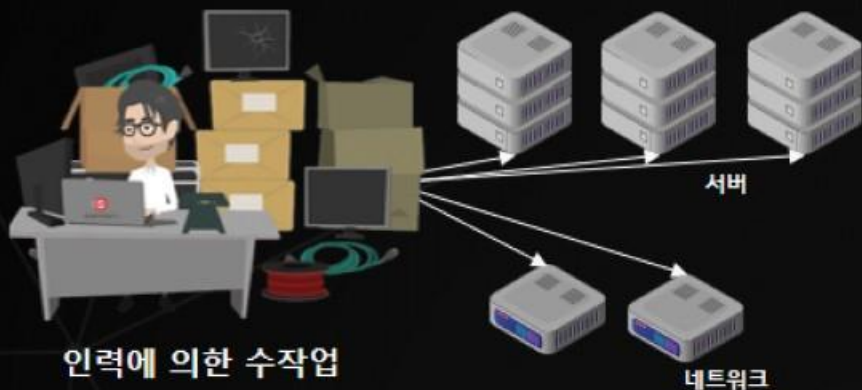
## 표준화와 자동화된 개발 환경



# 인프라 운영 - 기존 수작업 vs. Kubernetes 자동화 비교

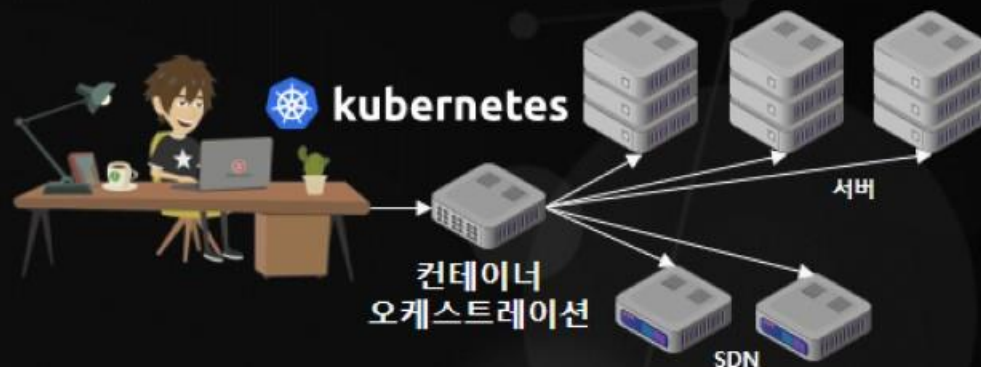
- IT 인프라의 대규모화, 고도화에 따라 IT 장비에 대한 환경설정 및 정보 취합이 복잡하고 어려움
- 작업 계획시간과 현장 작업 시간의 증가와 휴먼 에러의 증가

## 기존 시스템별 수작업



- ✓ 시스템 운영을 위한 관리 작업 증가
- ✓ 현장 작업 시간 증가
- ✓ Human Error 와 품질의 차이

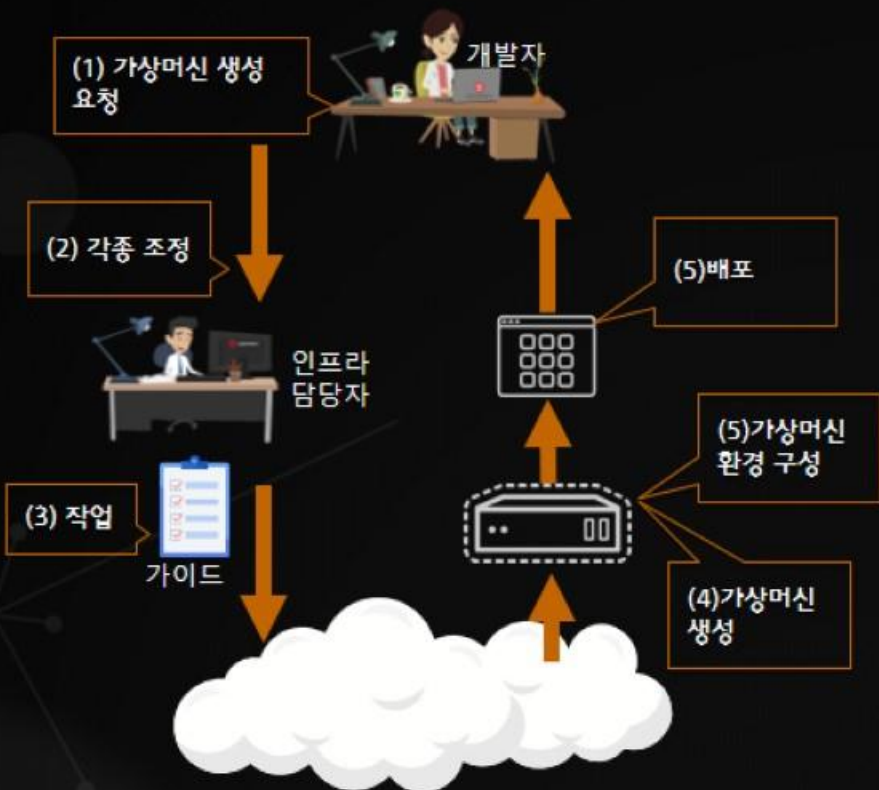
## 무정지 서비스/ 완전 자율 운영



- ✓ 운영 기술 표준화를 통한 준비 시간 및 작업 시간 감소
- ✓ 시스템 일괄 설정 작업 시간 단축
- ✓ 시스템을 통한 작업으로 Human Error 감소

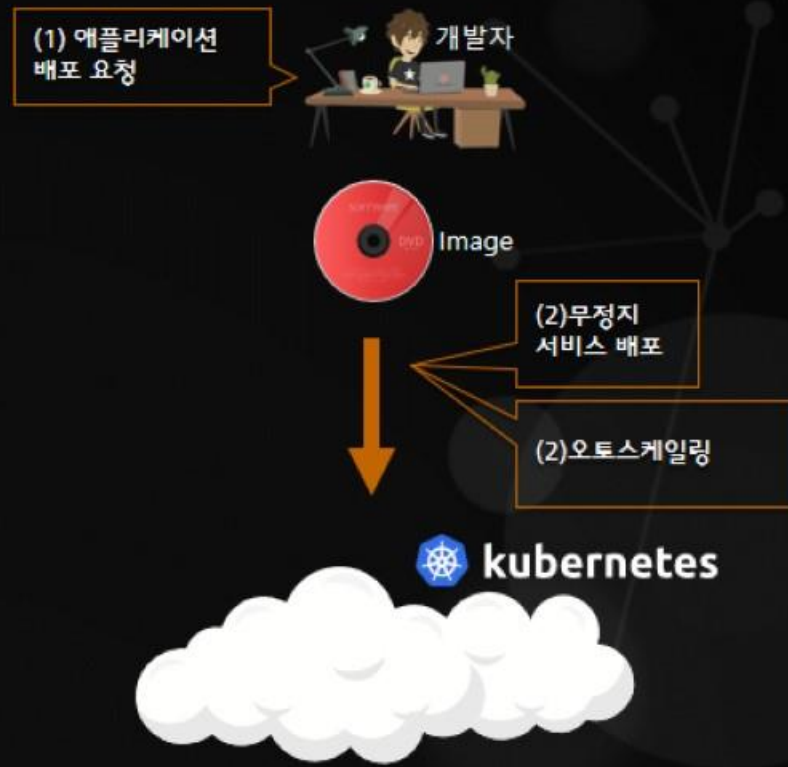
# 서버 추가 및 배포 - 기존 수작업 vs. Kubernetes 자동화 비교

## 수작업에 의한 서버 구성 및 배포



- ✓ 사람에 의한 조정과 작업
- ✓ Human Error 와 품질의 차이

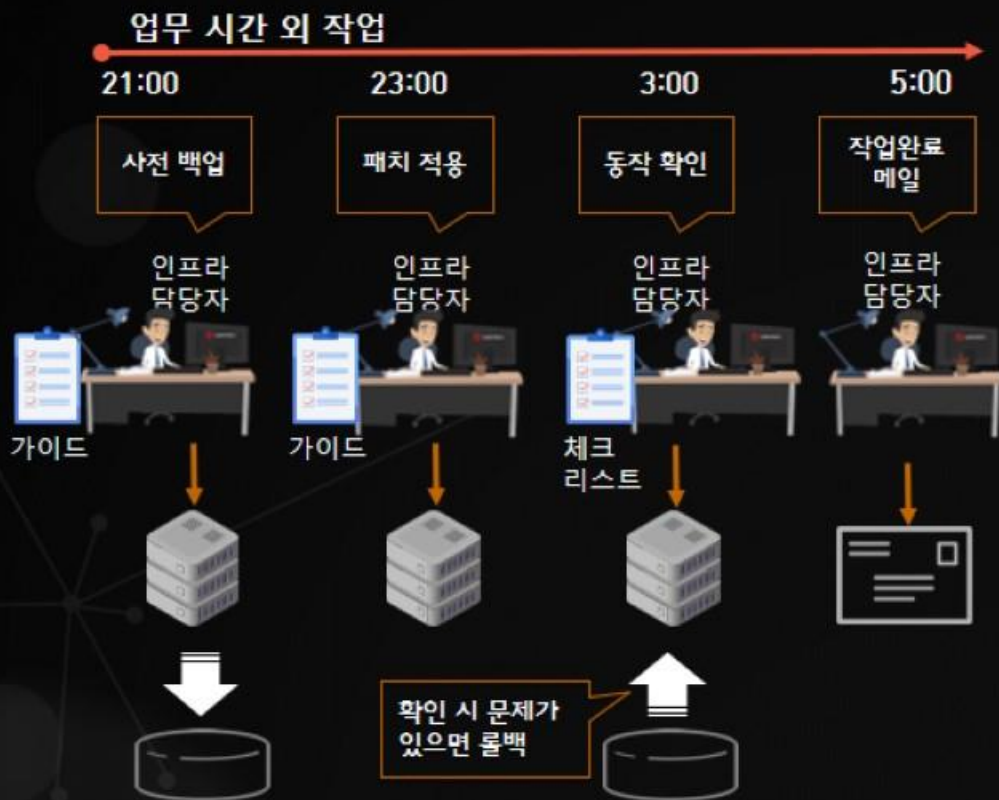
## 컨테이너 이미지 기반 무정지 서비스/자동 배포



- ✓ 테스트까지도 자동화
- ✓ 일정한 품질 유지

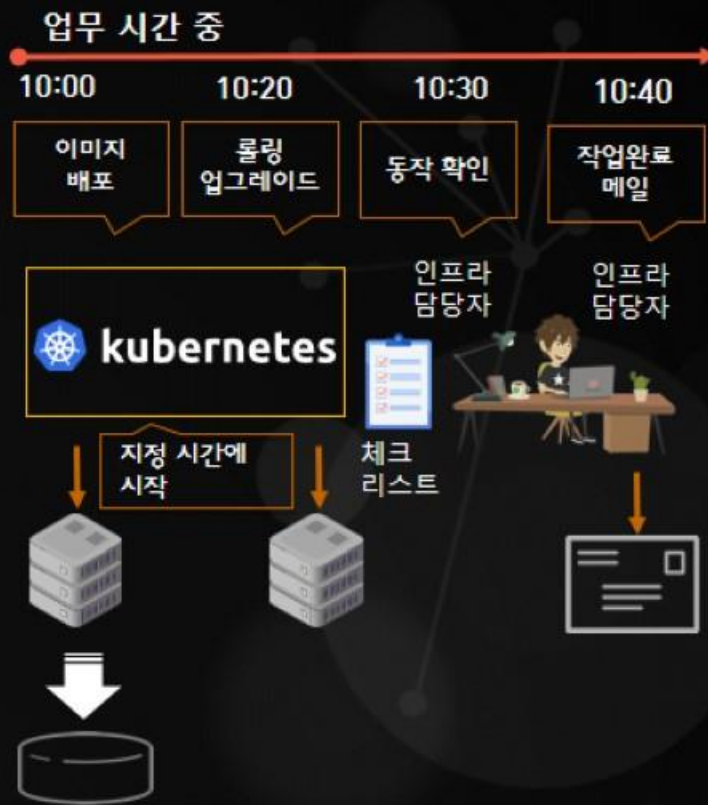
# 운영환경 패치 – 기존 수작업 vs. Kubernetes 자동화 비교

## 수작업으로 인한 야간 작업



- ✓ 작업하는 동안 상주 지원하며 시간 소요
- ✓ 오류 확인 및 롤백에 대한 위험요소
- ✓ 품질의 차이

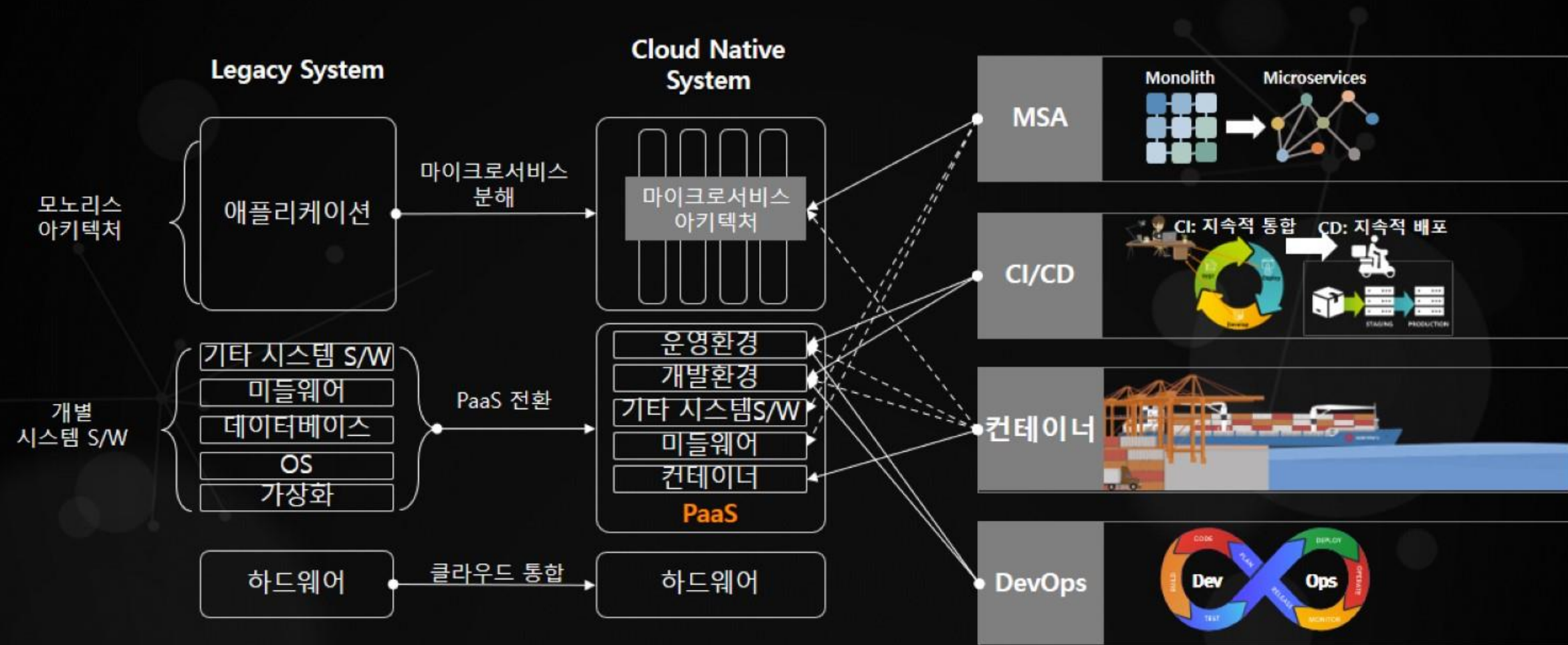
## 무정지 서비스로 업무 시간 중 패치



- ✓ 비상주 지원과 유비보수 시간 단축
- ✓ Human Error 차단
- ✓ 일정한 품질 유지

# IT 운영환경도 클라우드 네이티브 기반 PaaS로 전환

- 클라우드 네이티브는 "클라우드가 '클라우드 다룰 수 있도록' 애플리케이션을 구축, 실행하는 방식"

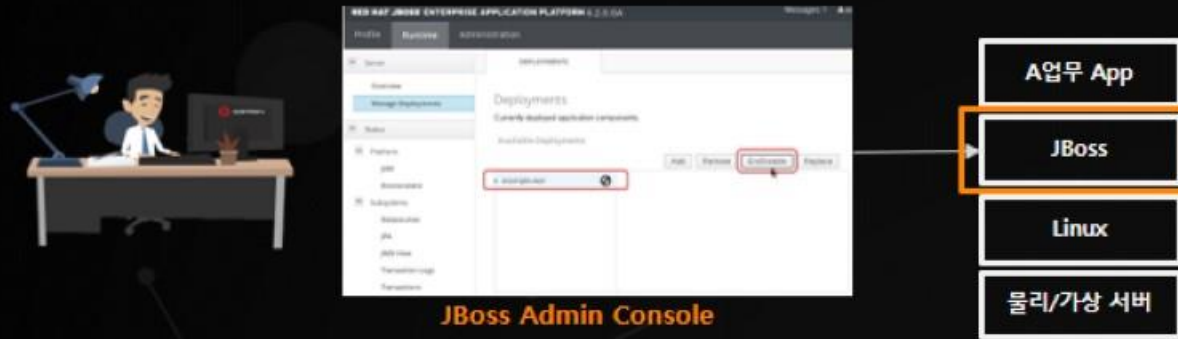


Container Orchestration

시스템 S/W 운영의 변화

# 인프라 운영 - WAS 관리 방법의 변화

- WAS 컨테이너 이미지는 관리 콘솔을 생략하고 배포
- 더 이상 개별 WAS 인스턴스에 대한 관리가 무의미 함



- WAS 수동 시작/정지
- WAS 수동 장애 복구

JBoss 관리 콘솔 - JBoss 관리

OpenShift 관리 콘솔 - 컨테이너(JBoss포함)

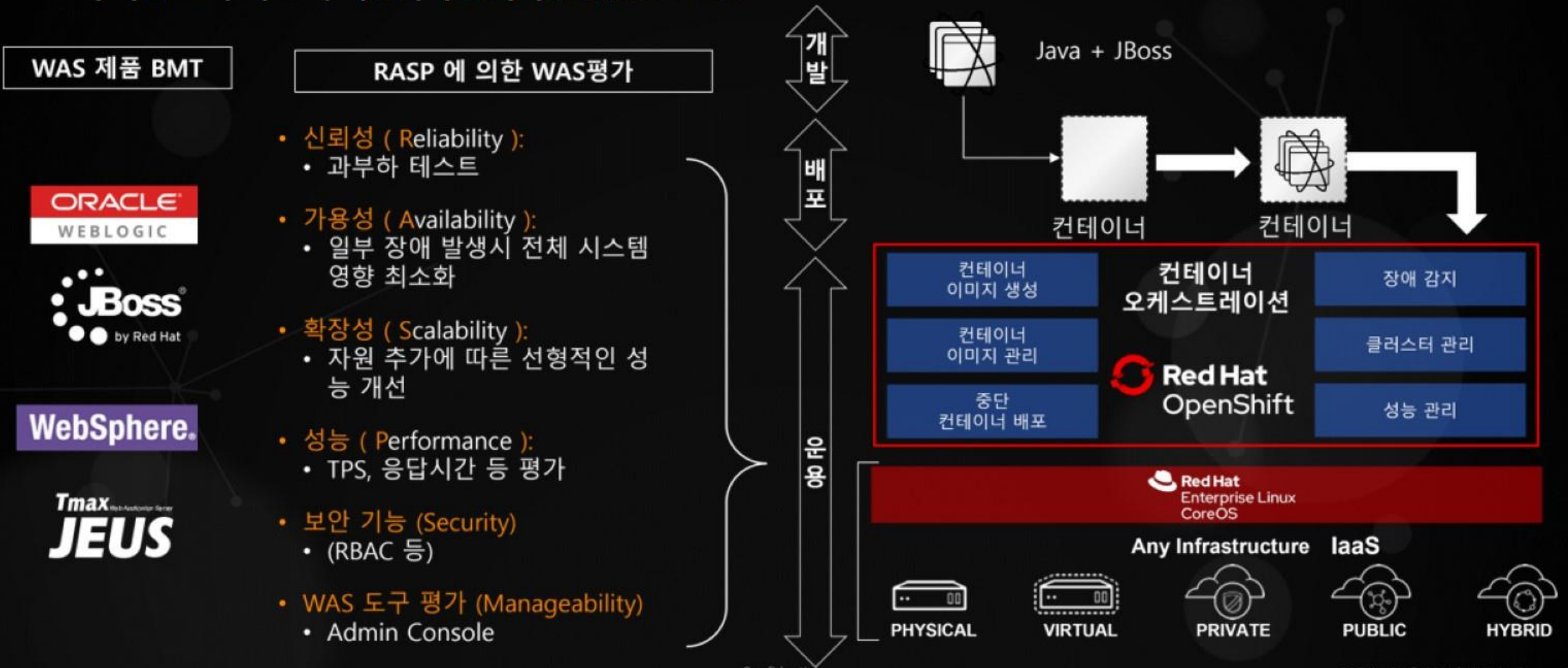


- 컨테이너 자동확장/축소
- 컨테이너 자동장애 복구
- 컨테이너 서비스체크
- 컨테이너 모니터링

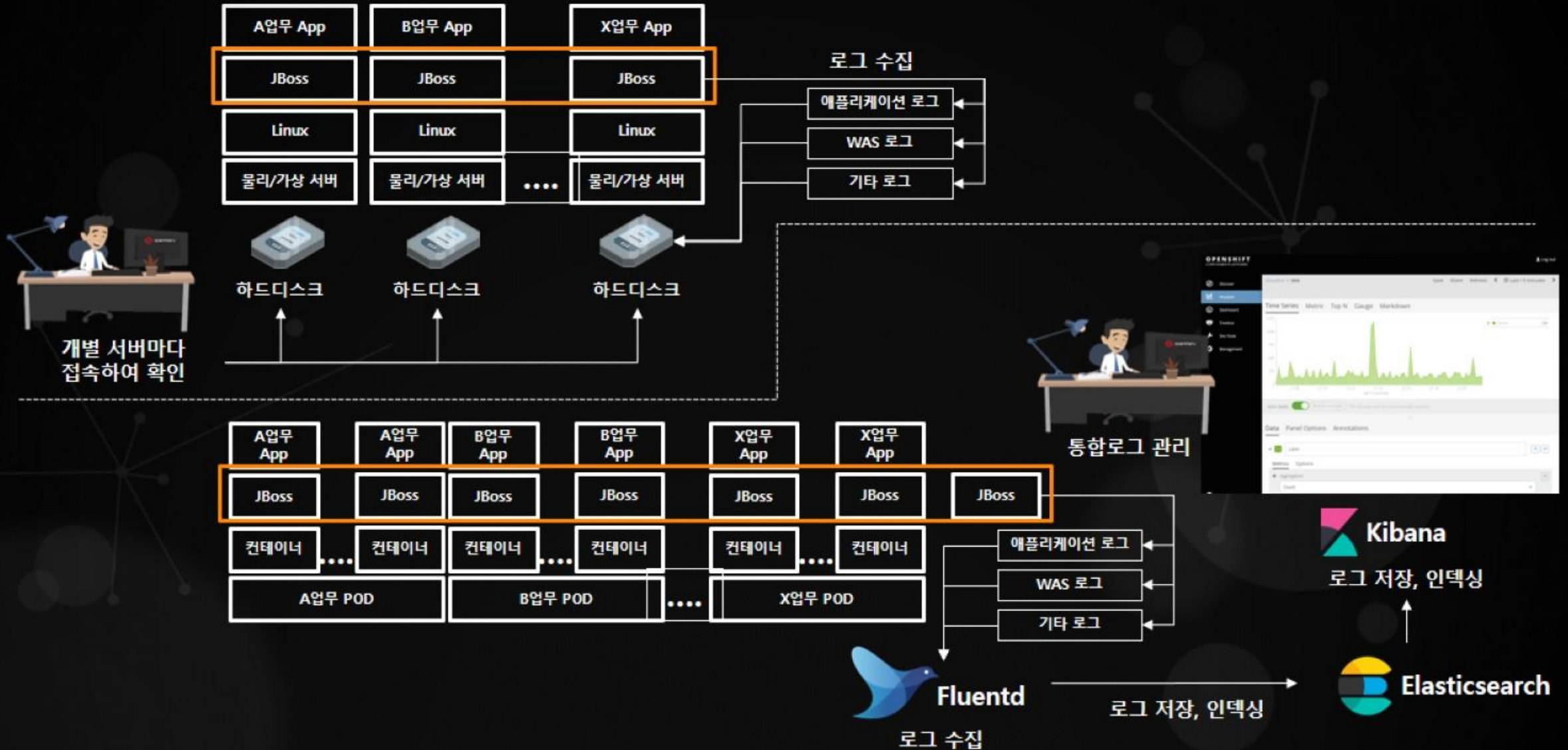


# 아직도 WAS BMT 나 POC를 하시나요?

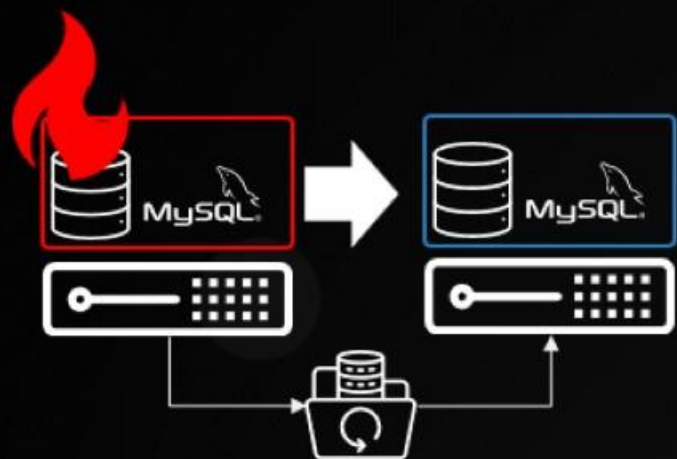
- WAS 의 가용성/확장성/성능/신뢰성 등의 기능은 플랫폼으로 이전
- 더 가볍고 더 빠르고, 자동화에 친화적인 WAS 로 전환



# 인프라 운영 - 로그 관리 방법의 변화

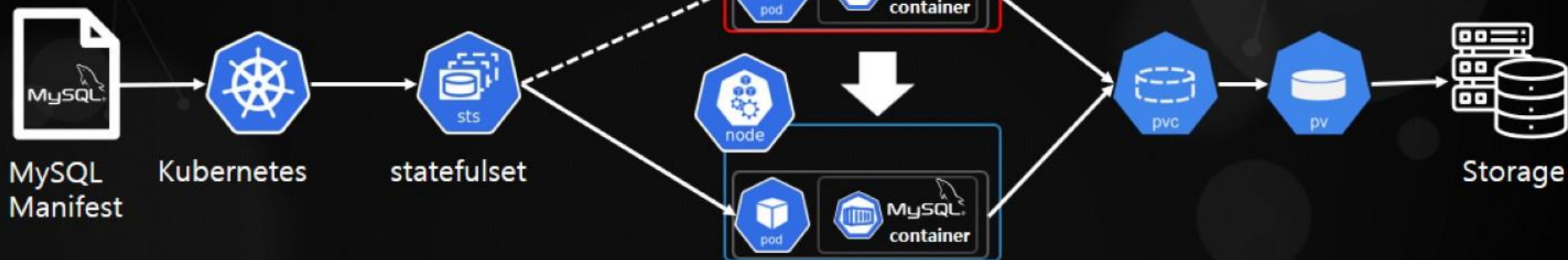


# Kubernetes 을 이용한 MySQL 이중화
























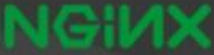

- 수작업에 의한 장시간 조치 및 오류
- HA 솔루션을 통한 이중화
- DB 장애 시 전환 작업

- 유연한 POD 스케줄링
- StatefulSet 으로 안정적인 POD 업데이트
- 서비스를 통한 연결 대상 자동 전환 및 부하 분산



# OpenShift Template 카탈로그 리스트

- OpenShift 에서는 각 서비스들을 프로비저닝 할 수 있는 템플릿을 제공
- CI/CD, Databases, Languages, Middleware, 기타 항목의 총 79 개의 템플릿을 제공

| CI/CD  | Databases   | Languages  | Middleware   | Etc..  |
|--|---|--|--|--|
| <br>jenkins | <br>PostgreSQL<br><br>MariaDB<br><br>MySQL | <br>Java<br><br>PHP<br><br>Node.js<br><br>Ruby<br><br>Python<br><br>Dancer | <br>JWS<br><br>HTTPD<br><br>A-MQ<br><br>3-scale<br><br>Wildfly<br><br>BPM<br><br>BRMS<br><br>Keycloak<br><br>JDG<br><br>Fuse<br><br>Data Virtualization | <br>Nginx<br><br>Redis |

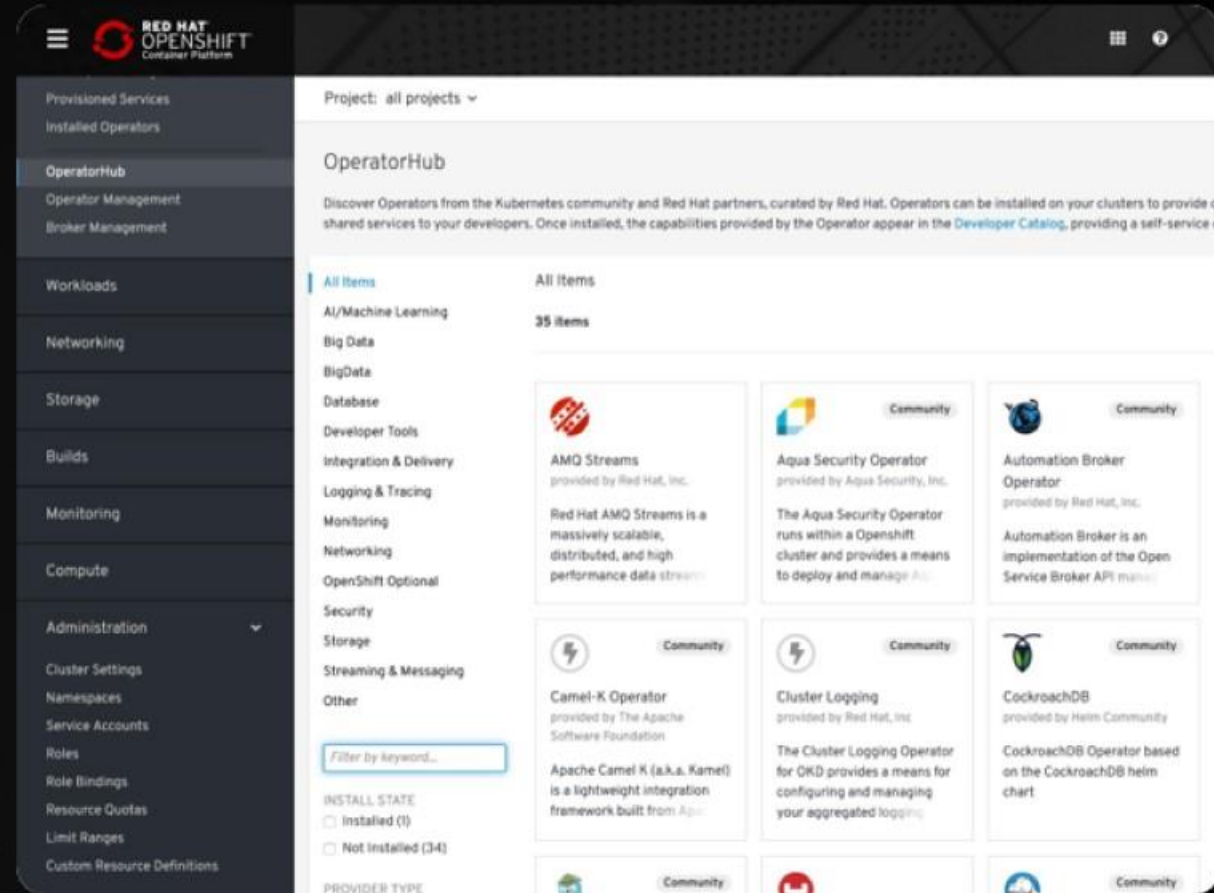
# OPERATOR HUB



- 시스템S/W 를 OperatorHub 를 통해 손쉽게 설치와 구성
- AI/Machine Learning, Integration & Delivery, Logging & Tracing, Networking 등의 18 가지 항목의 426 가지 Operator 를 제공



...and many more



Container Orchestration

# Summary

- OpenShift 을 적용하여 민첩성 높은 서비스를 제공





openmaru





openmaru

제품 / 서비스에 관한 문의

- 콜 센터 : 02-469-5426 ( 휴대폰 : 010-2243-3394 )
- 전자 메일 : [sales@openmaru.com](mailto:sales@openmaru.com)