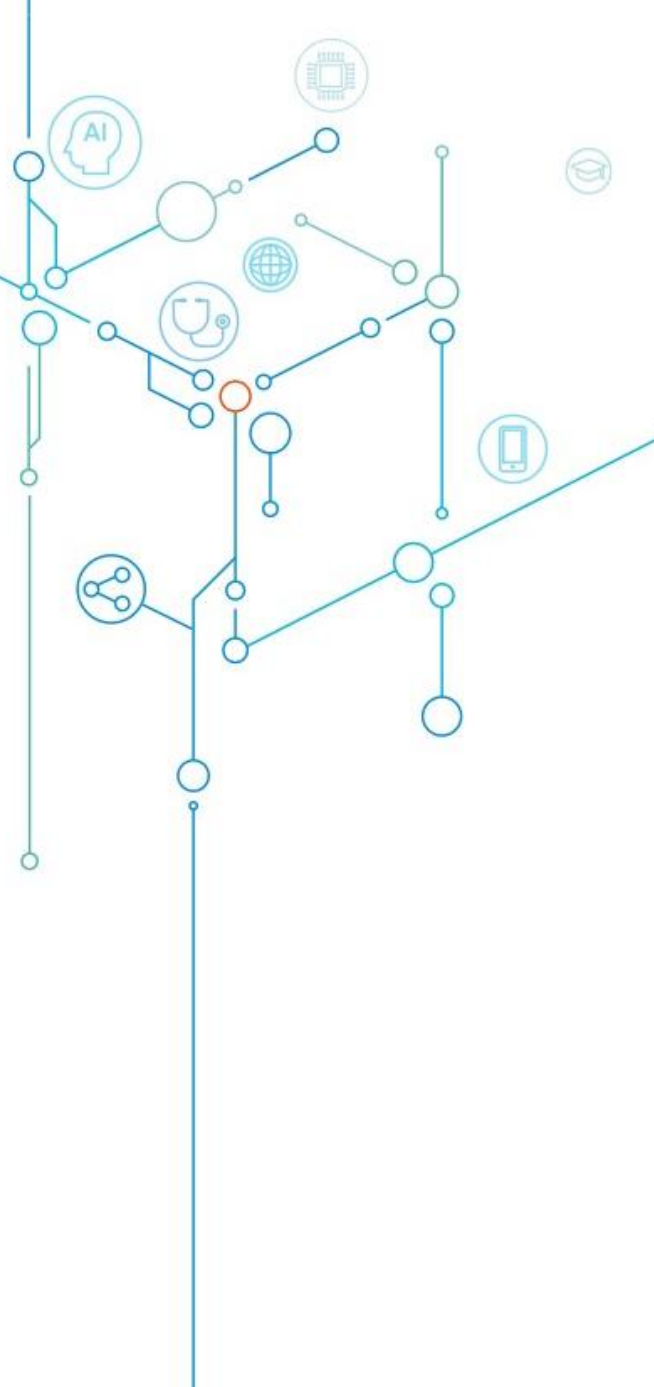


컨테이너와 하이브리드 클라우드 동행

조형근
한국레드햇 솔루션 아키텍트

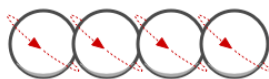


IT 기술 변화

시장의 변화에 맞게 최대한 짧은 시간에 상품을 시장에 출시하냐에 따라 기업의 생존이 결정되기 때문에 이를 지원하는 IT 기술도 개발방법론, 애플리케이션 아키텍처, 배포방법, 인프라 기술에 변화가 생겼음

Development Process

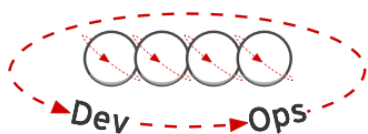
Waterfall



Agile

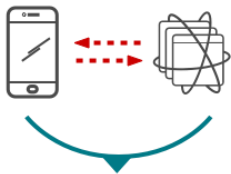


DevOps



Application Architecture

Monolithic



N-Tier

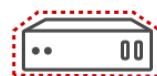
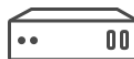


Microservices



Deployment & Packaging

Physical Servers



Virtual Servers



Containers



Application Infrastructure

Datacenter



Hosted

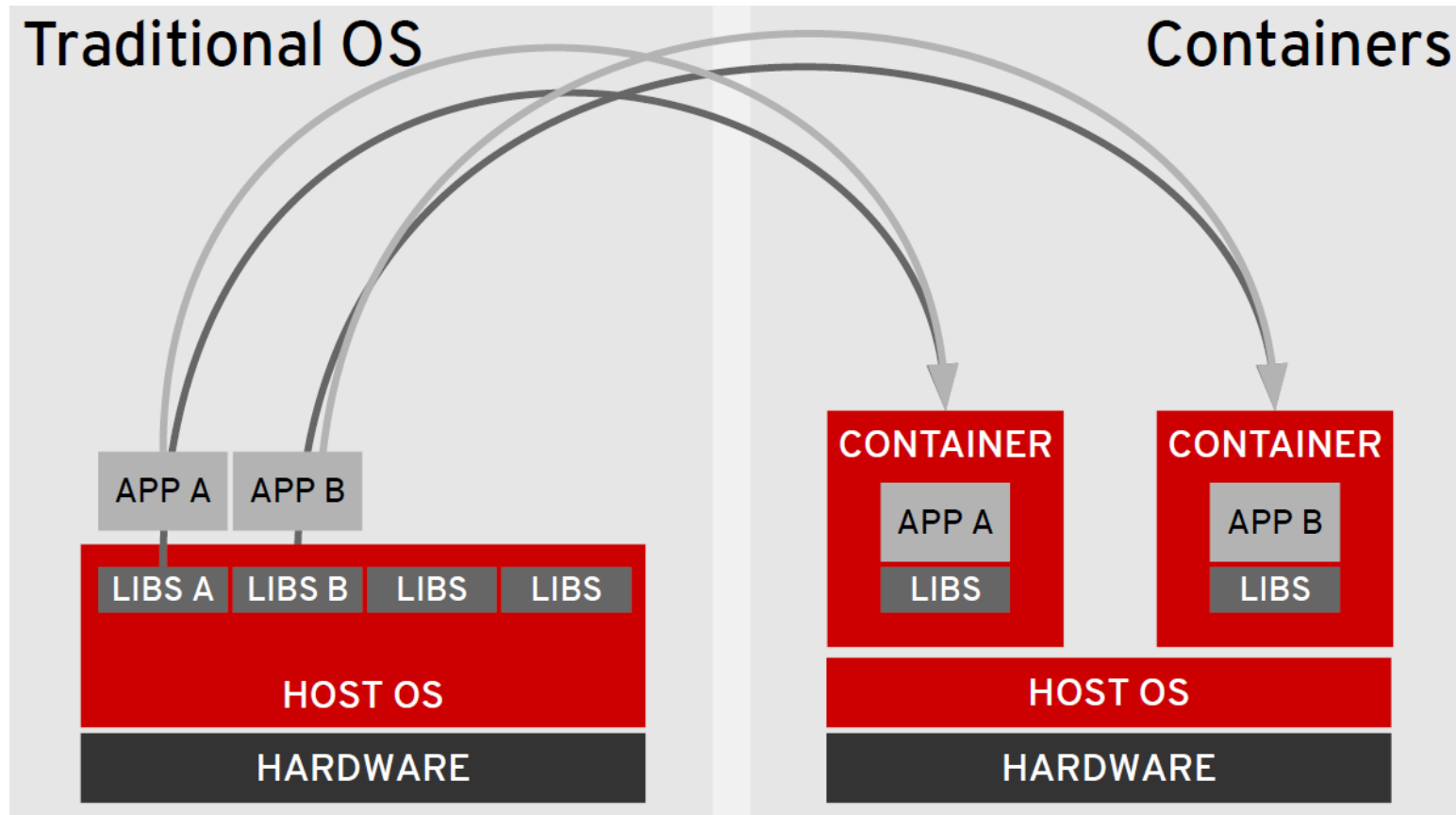


Cloud



컨테이너(Container)

컨테이너는 실행에 필요한 모든 파일을 포함하여 전체 런타임 환경에서 애플리케이션을 패키지와 분리할 수 있도록 하는 기술입니다. 이를 통해 전체 기능을 유지하면서 컨테이너화된 애플리케이션을 환경(개발, 테스트, 생산 등) 간에 쉽게 이동할 수 있음



컨테이너 기술 요소

컨테이너에 사용되는 기술은 호스트 운영체제에서 리소스를 분리하고 관리하기 위해 사용되며, 리눅스 운영체제가 갖고 있는 Namespace, control groups, SELinux, iptables 기술을 사용

Namespace

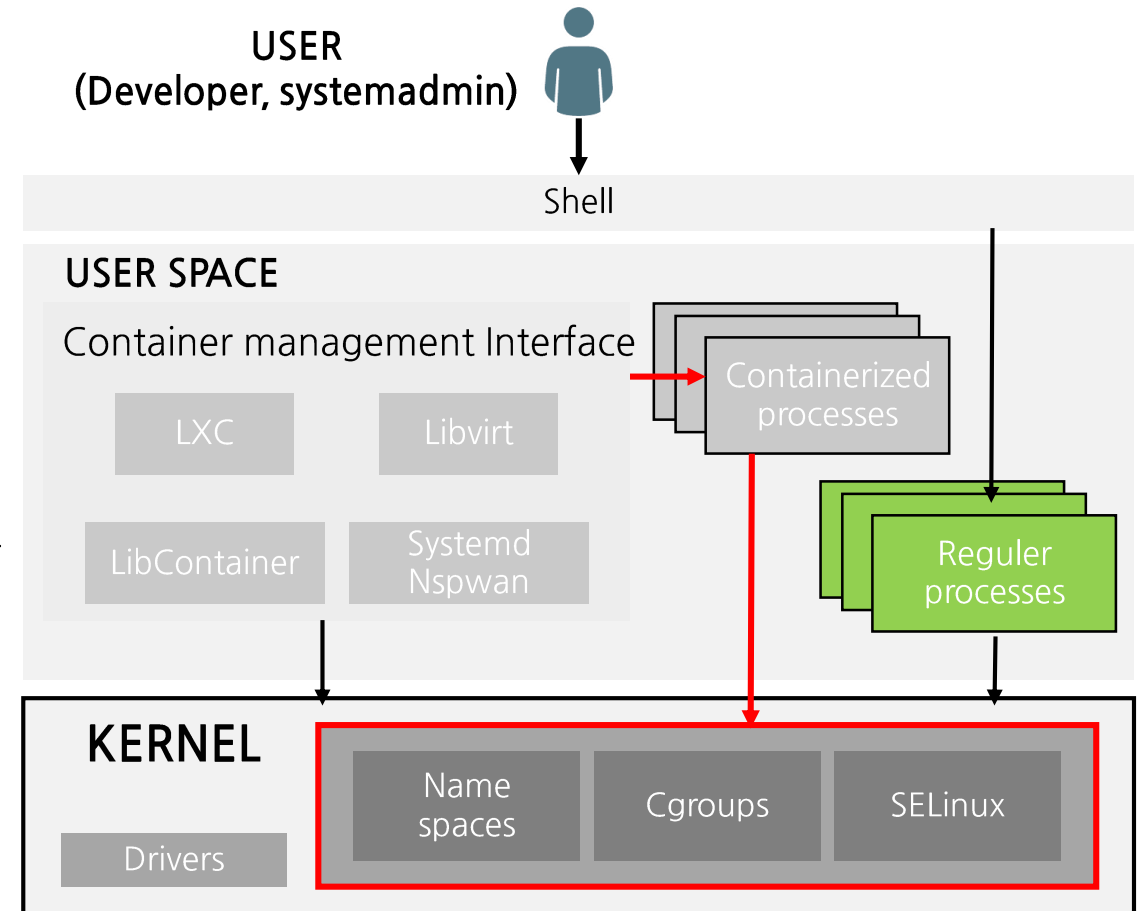
커널은 리소스를 네임 스페이스 내에 배치하여 일반적으로 모든 프로세스에 표시되는 특정 시스템 리소스를 격리 할 수 있습니다. 네임 스페이스 내에서 해당 네임 스페이스의 멤버인 프로세스만 해당 리소스를 볼 수 있습니다. 네임 스페이스에는 네트워크 인터페이스, 프로세스 ID 목록, 마운트 지점, IPC 리소스 및 시스템의 호스트 이름 정보와 같은 리소스가 포함될 수 있습니다.

Control groups (cgroups)

cgroups는 프로세스와 해당 하위 세트를 그룹으로 분할하여 그들이 사용하는 자원을 관리하고 제한합니다. 제어 그룹은 프로세스가 사용할 수 있는 시스템 자원의 양을 제한합니다. 이러한 제한은 하나의 프로세스가 호스트에서 너무 많은 리소스를 사용하지 못하게합니다.

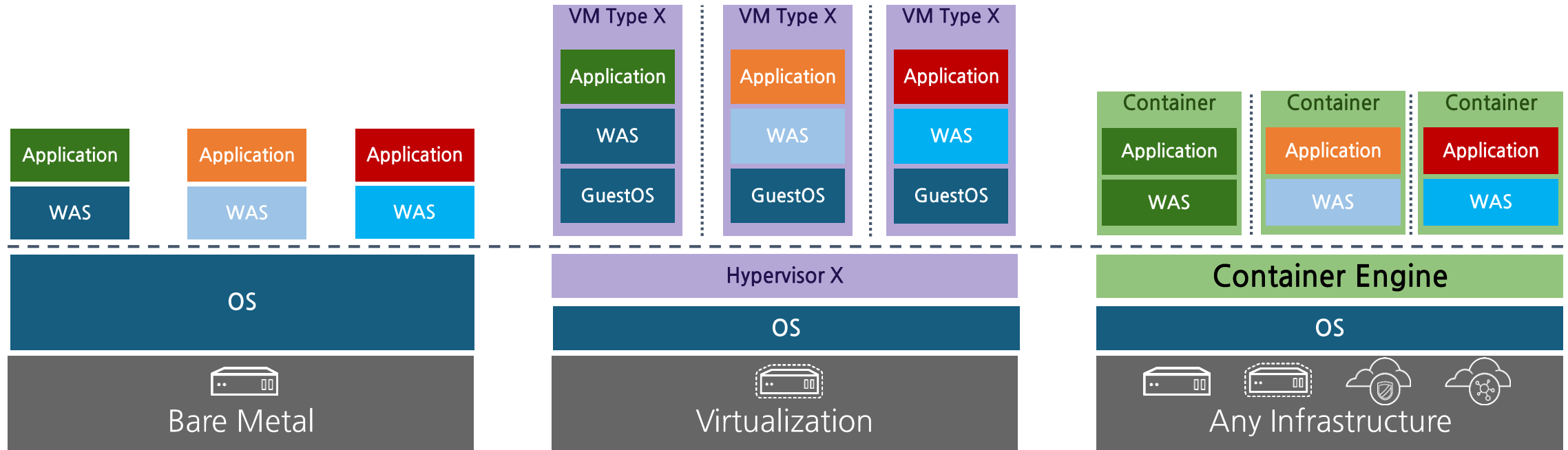
SELinux

SELinux (Security-Enhanced Linux)는 프로세스를 위한 필수 액세스 제어 시스템입니다. Linux 커널은 SELinux를 사용하여 프로세스를 서로 보호하고 실행중인 프로세스로부터 호스트 시스템을 보호합니다. 프로세스는 호스트 시스템 자원에 대한 액세스가 제한된 제한된 SELinux 유형으로 실행됩니다.



컨테이너 장점

컨테이너는 가상서버 대비 Guest OS가 없기 때문에 컴퓨팅 자원을 좀 더 효율적 사용하며, 이기종 VM 간 마이그레이션의 한계를 극복하여 자원 효율성, 자원 격리, 호환성, Auto Scaling, 마이크로서비스, 관리 편의성을 제공



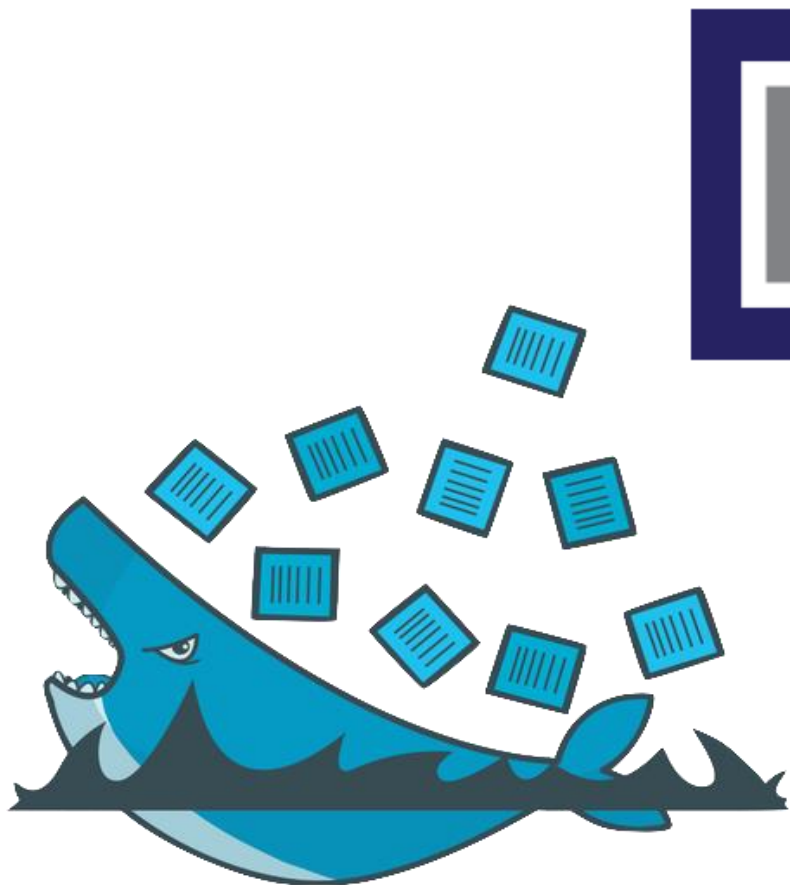
- 자원(CPU, Memory) 격리 불가
- OS 간 호환성 문제
- Application 자동 확장 불가

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

- 자원(CPU, Memory) 격리
- 표준 컨테이너 기술로 호환성 제공
- Application 자동 확장 가능
- 하이브리드 클라우드 환경

표준 컨테이너 사용

Docker 제단의 Enterprise 출시에 따른 Open Source 제약에 따른 컨테이너 표준 제정 필요성 대두



OPEN CONTAINER
INITIATIVE

Established in June 2015

Create Open Industry Standards
around Container Formats and Runtime

<https://www.opencontainers.org/>

쿠버네티스(Kubernetes)

구글이 2014년 오픈소스 개방한 컨테이너 오케스트레이터 쿠버네티스는 사업계 사실상 표준으로 자리잡음

표준 CONTAINER ORCHESTRATION

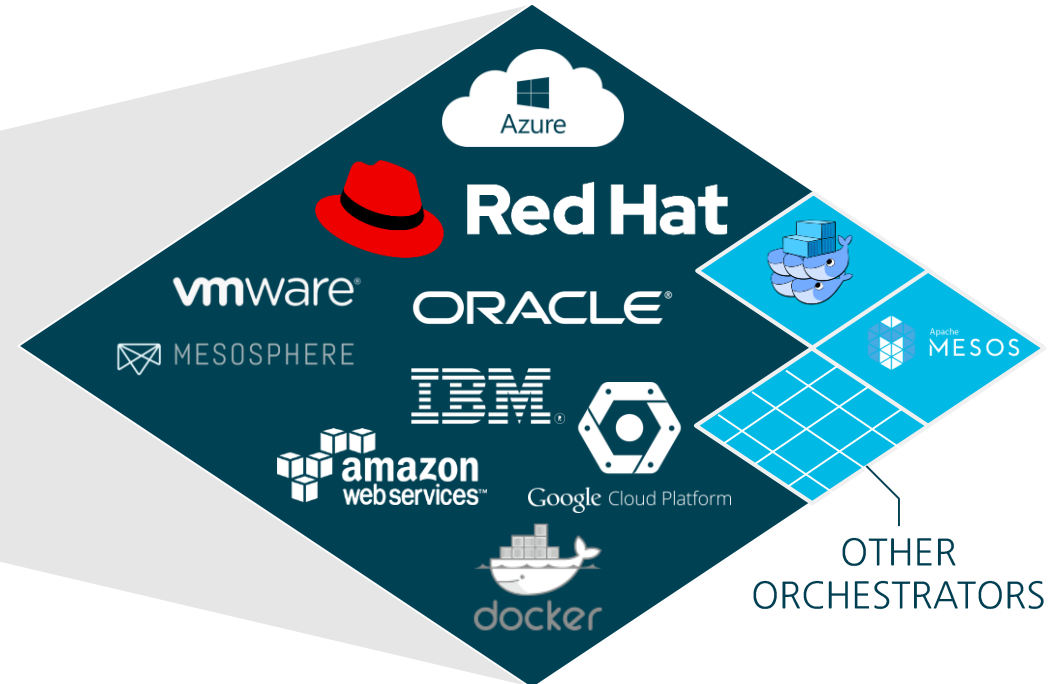
4-5 YEARS AGO

Fragmented landscape



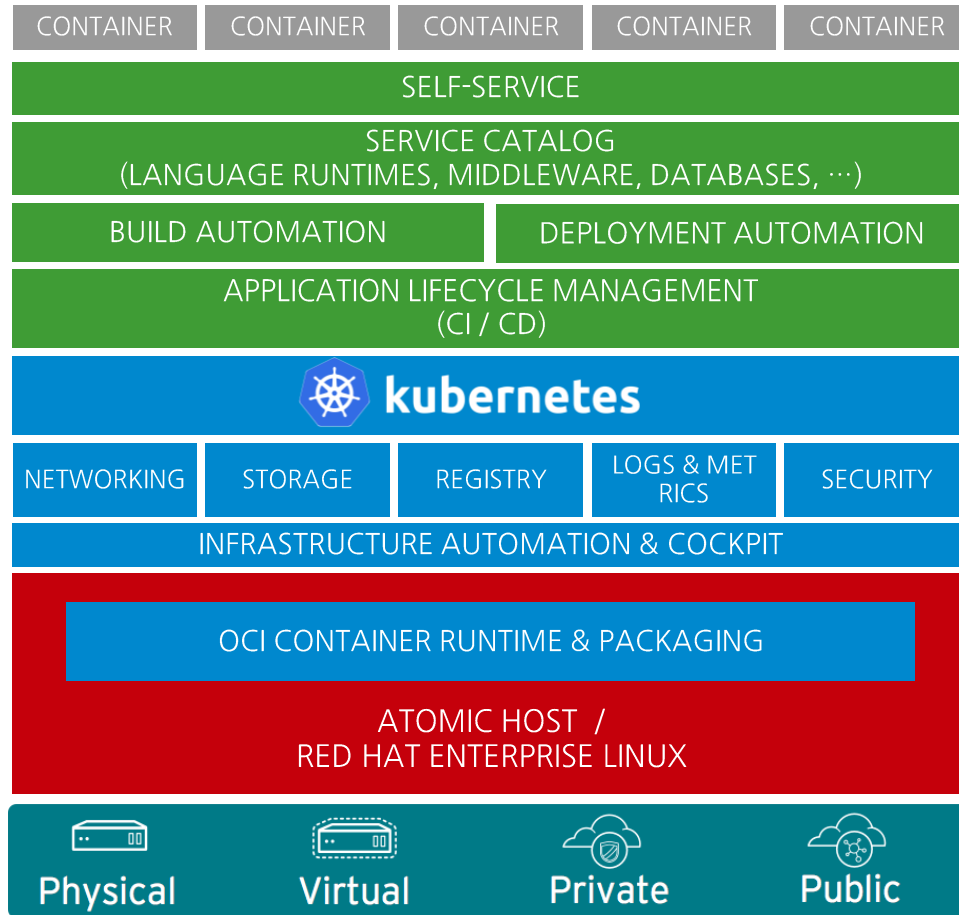
TODAY

Kubernetes consolidation



레드햇 오픈시프트(Red Hat OpenShift)

레드햇 오픈시프트는 쿠버네티스 위에 구성된 PaaS(Platform as a Service) 이다. PaaS 플랫폼은 멀티 테넌시, 보안 강화, 모니터링 및 감사, 응용 프로그램 수명주기 관리 및 개발자를 위한 셀프 서비스를 제공한다.



- 개발 및 운영을 위한 다양한 툴 제공(IDE, UX 등)
- 전통적인 WEB/WAS, 최신의 MSA 및 Serverless 아키텍처를 위한 다양한 application runtimes & services 제공
- Auto-Healing으로 Application HA 기능 제공
- Auto-Scaling으로 과부하 대비 기능 제공
- Kubernetes 기반의 컨테이너 오케스트레이션
- Docker/OCI 표준 컨테이너
- 컨테이너 최적화 OS 기반 - Red Hat Enterprise Linux/Atomic Host
- 하이브리드 클라우드 환경에 설치(Bare-Metal, 다양한 VM, Public/Private IaaS 등)

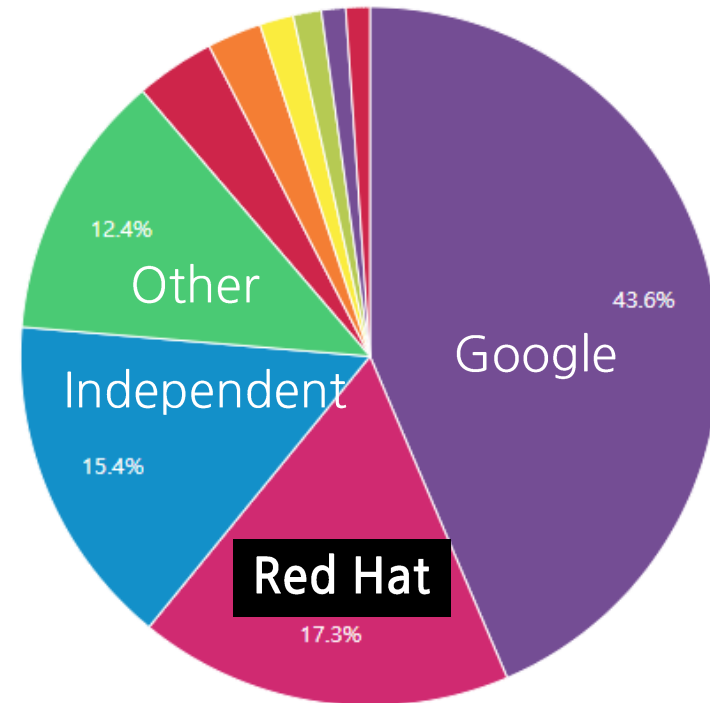
쿠버네티스 기여도

Commits by Company

Show entries

#	Company	Commits
1	Google	19648
2	Red Hat	7811
	*independent	6961
3	Huawei	1658
4	ZTE Corporation	1133
5	Microsoft	707
6	FathomDB	587
7	CoreOS	505
8	VMware	500
9	Fujitsu	482

Showing 1 to 10 of 115 entries [Previous](#) [Next](#)



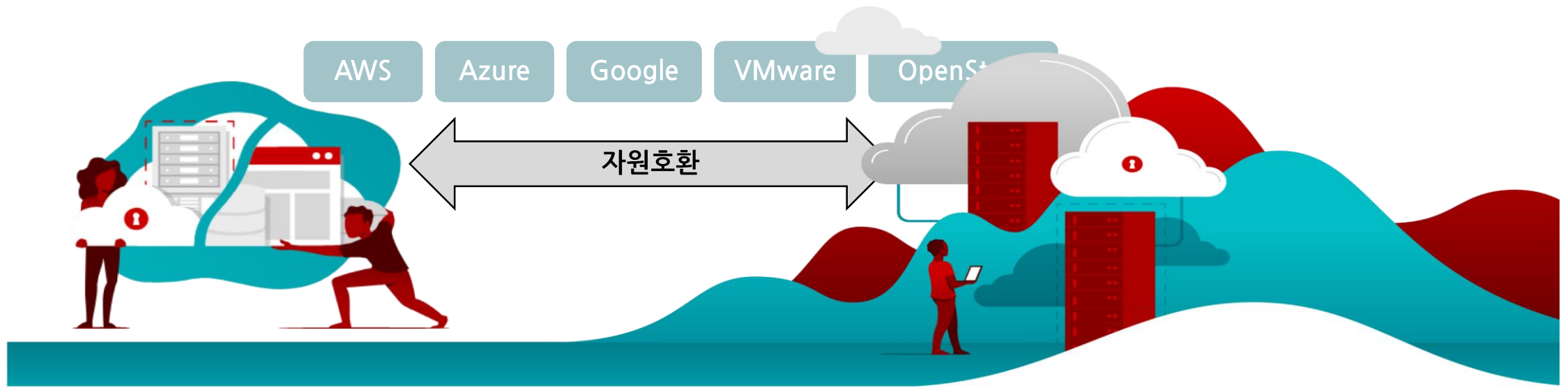
+ 레드햇의 한국내 OpenShift 지원 인력 50명 이상

Source: Stackalytics (May, 2019)

<https://www.stackalytics.com/cncf?module=kubernetes&date=all>

멀티 클라우드

많은 기업이 애플리케이션 운영환경으로 멀티클라우드 사용하여 벤더 락인 예방과 장애대응 기능의 장점을 확보한 반면 이기종 매니지먼트 환경 관리의 단점도 있음



긍정적 요인

- 벤더 락인의 예방
- 클라우드 벤더별 특색
- 퍼포먼스 향상(빠른 서비스를 위해 가장 가까운 곳에 위치한 호스팅 클라우드 선택 가능)
- Fail-Over(리전장애 / 자연재해)

부정적 요인

- 이기종 매니지먼트 환경-복잡성 및 새로운 학습 필요
- 각각의 보안 정책 및 워크플로우 필요

멀티 클라우드 관리 어려움

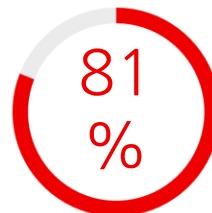
조직이 여러 클라우드에 걸쳐 더 많이 배포
할 수록, 새로운 과제가 발생합니다.

- ▶ 대규모 관리는 어렵고 오류가 발생하기 쉬움
- ▶ 검증 해야하는 구성요소, 설정,
규정이 너무 많음
- ▶ Configuration Drift 발생 가능성
: 조금씩 변경된 서버 설정이
원래 배포된 설정과 점차 달라져
확장이나 디버깅이 어려워짐
- ▶ 환경마다 일관되지 않은 보안 통제
- ▶ 개별 클러스터 수준에서 관리 시
클러스터 수에 따라 관리비용이 빠르게 증가

운영 애플리케이션에 2개 이상의 “인프라 클라우드”를 적극적으로 사
용하는 200대 10억 달러 기업에 대한 IDC 설문조사



여러 인프라 클라우드 사용*



여러 퍼블릭 클라우드 및 하나 이상의 프
라이빗 클라우드 사용*

Source:

DC Multicloud Management Survey, 2019: Special Study, Doc # US45020919, April 2019

*IDC Survey of 200 US-based \$1B companies actively using two or more “infrastructure clouds” for production applications

멀티 클라우드 vs. 하이브리드 클라우드



MULTI CLOUD

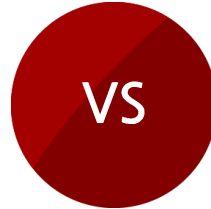
클라우드사이의 워크로드 이식성, 상호 연결,
오케스트레이션, 통합 관리 없이
퍼블릭과 프라이빗 클라우드를 개별적으로 사용

Google

amazon

openstack.

Microsoft
Azure



HYBRID CLOUD

클라우드사이의 워크로드 이식성, 상호 연결,
오케스트레이션, 통합 관리를 통해
퍼블릭과 프라이빗 클라우드를 상호 운용적으로 사용

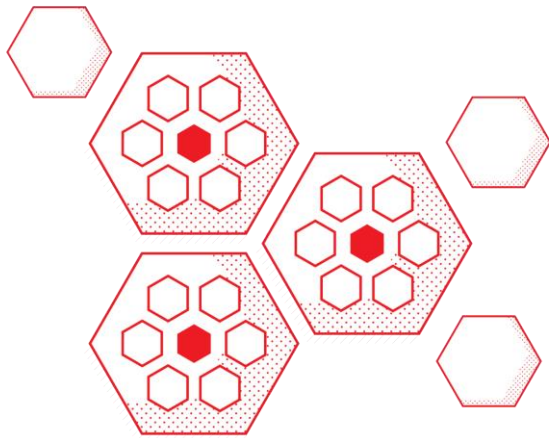
amazon

Microsoft
Azure

Google

openstack.

하이브리드 클라우드와 쿠버네티스



“멀티 테넌시(Multi-tenancy), 재해 복구(DR), 하이브리드, 멀티 클라우드 또는 에지 배포 등 다양한 이유로 컨테이너와 쿠버네티스가 업계 전반에서 채택됨에 따라, 인프라&운영팀이 온-프레미스나 클라우드, 또는 여러 지역에 걸쳐 다중 클러스터를 구축하고 관리해야 하는 경우가 발생하고 있습니다.”

Source:

Assessing Patterns for Deploying Distributed Kubernetes Clusters doc # G00465217, by Tony Iams

The Next Red Hat Hybrid Cloud Strategy = OpenShift



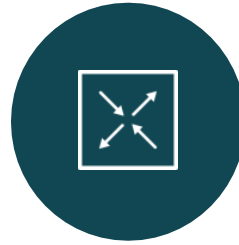
Automate
d operatio
ns



Multi-tenant



Secure by defau
lt



Network t
raffic control



Over-the-air
updates



Monitoring
& chargeback



Pluggable
architecture



RED HAT
ENTERPRISE LINUX



RED HAT
VIRTUALIZATION



RED HAT
OPENSTACK
PLATFORM



Google Cloud



aws

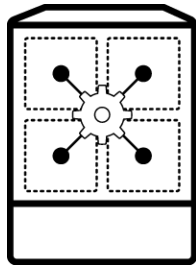


Azure



OTHER PUBLIC
CLOUD

왜 하이브리드 클라우드는 레드햇 오픈시프트인가?



Trusted enterprise Kubernetes



Cloud-like experience everywhere



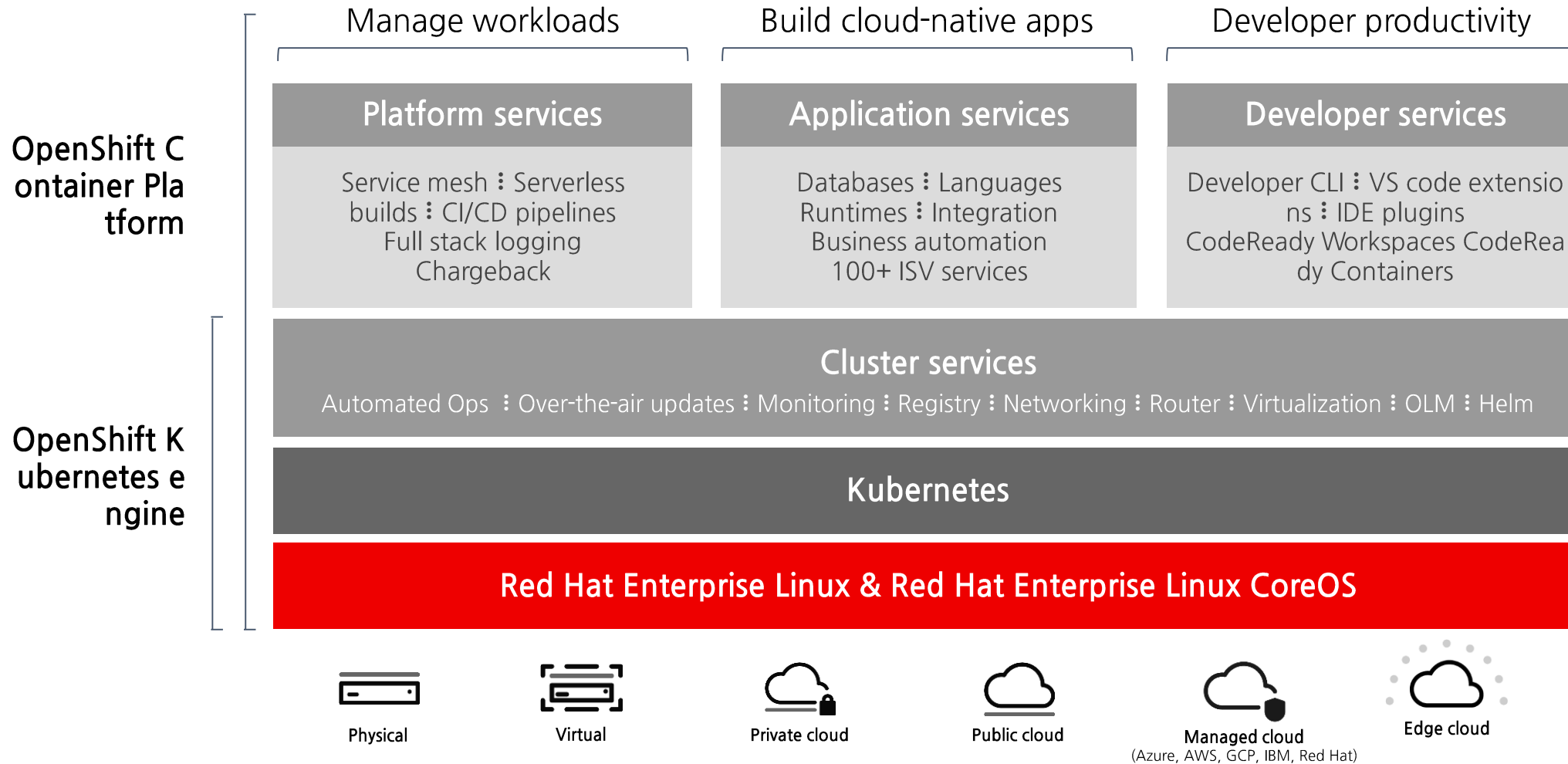
Empowering developers to innovate



Open source innovation

Red Hat OpenShift with Open Source Software

레드햇은 오픈소스 소프트웨어 성장과 함께합니다.



—



감사합니다

—●

