

AKS 활용법 101

Korea, 2020



Container 그리고 Docker

Korea, 2020



Container 란



- VM보다 가벼운 형식으로 애플리케이션을 pack, ship, run
- 컨테이너는 프로세스 격리를 기반으로 하는 애플리케이션 전달 메커니즘
- Linux Kernel 기술 사용 : cgroups, namespaces + overlay fs + tooling
- 컨테이너 이미지를 사용하면 응용 프로그램 코드, 런타임 및 모든 Dependency들을 Pre-Defined Format으로 제한 가능
- Container 는 새로운 것이 아닙니다 - 리눅스 LXC, Solaris Zones, BSD Jails 처럼 기존에 있던 개념!!!
 - ✓ Docker 는 기존의 있는 것을 사용하여 빌드하고 관리할 오픈소스 소프트웨어를 만들었음

Containers vs. VM's

가상 머신 (Virtual Machines)

개별 VM이 독립적으로, OS 전체와 함께 동작
완벽한 격리 (isolation)
앱 프레임워크 분리가 잘 이루어짐
라이브 마이그레이션 등 기능 지원
부팅이 다소 느림



Virtual Machine

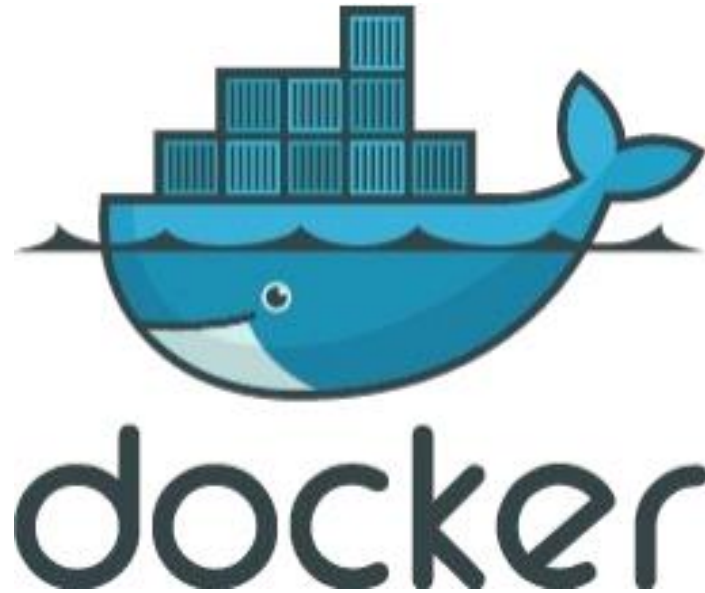
컨테이너 (Containers)

호스트 OS를 공유
거의 즉시 실행 (start-up)이 이루어짐
컨테이너 내 프로세스가 격리됨
의존성있는 앱 서비스 & 라이브러리가
컨테이너에 의존 (layer)
모든 컨테이너가 격리되어 제공되며, 개별 파일
시스템을 보유 – PID0 (루트 프로세스) 및 eth0
네트워크 인터페이스를 가짐



Container

Docker 란

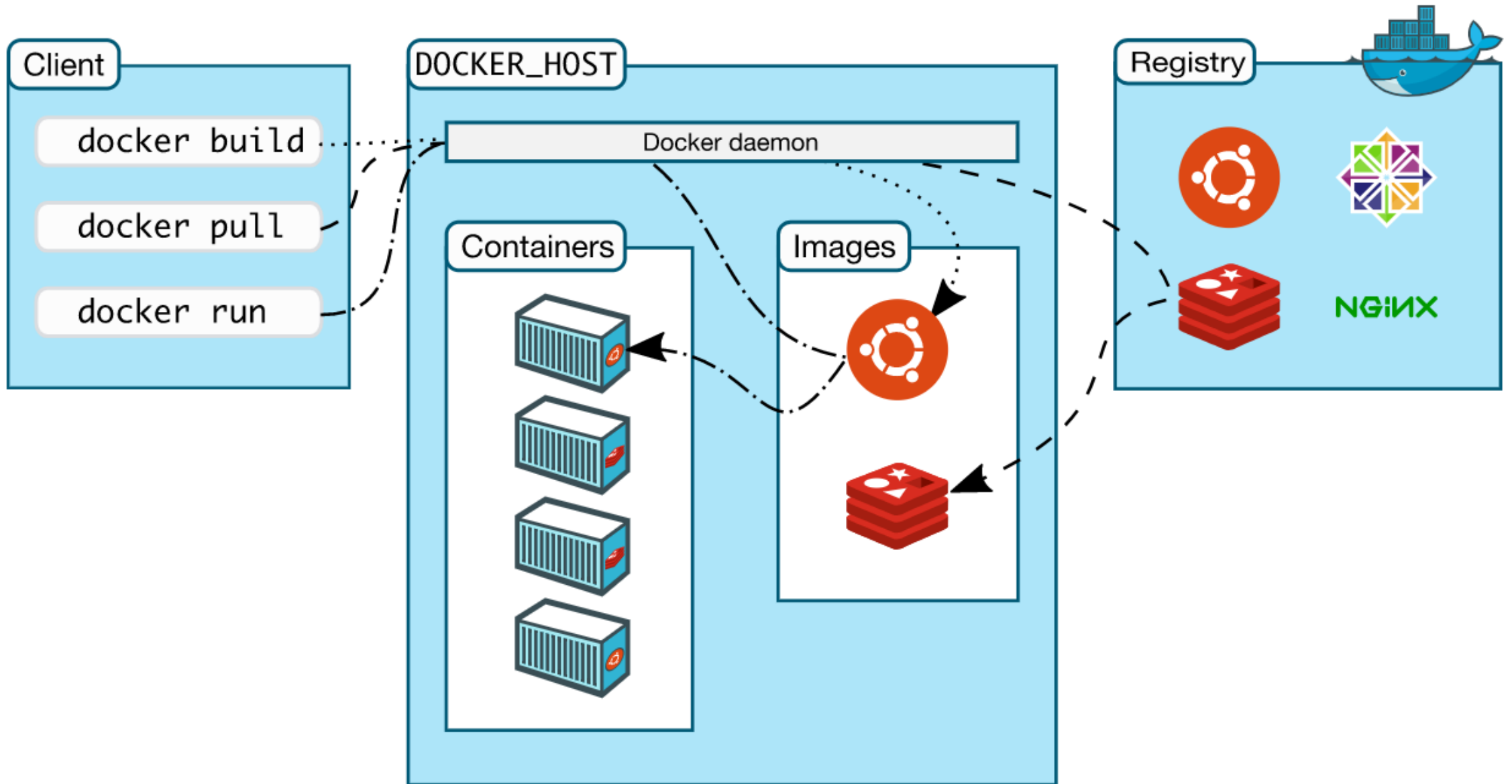


2013년 3월 PyCon Conference – Docker (Solomon Hykes)

컨테이너 기반의 오픈소스 가상화 플랫폼

Docker - Build, Ship, and Run Any App, Anywhere

Docker 아키텍처



레이어 within a Docker 이미지

- 도커파일은 텍스트 기반 문서로, 도커 이미지를 자동으로 빌드하기 위한 명령들을 기술함
- Container 는 Multiple Layer로 구성되어 있음 - Top layer는 read/write, 나머지는 read-only
- This is made possible by an overlay filesystem
- The lowest layer is referred to as the base image (ubuntu:1704)



Kubernetes

Korea, 2020



컨테이너, 그 기반에는 무엇이 있는가?

Kubernetes (쿠버네티스): 앱 개발 & 관리 전반에서 컨테이너를 오케스트레이션

다양하게 선택 & 사용

Kubernetes is in production for **global companies across industries**¹

Capital One	eBay	SAP
New York Times	Pokémon Go	Spotify

Vendor-neutral

A **variety of cloud providers** offer robust Kubernetes support

Azure	AWS
VMWare	Red Hat

커뮤니티 지원

There's a **huge community** of active contributors supporting Kubernetes³

24,000 contributors since 2016	1.1 million contributions since 2016
--------------------------------------	--

¹Kubernetes.io. "Kubernetes User Case Studies." ²CNCF. "Kubernetes Is First..." ³CNCF. Keynote address.



Kubernetes(k8s)란?

Kubernetes: “컨테이너화된 응용 프로그램에 대한 자동화된 배포, 확장, 그리고 관리를 위한 오픈 소스 소프트웨어”

그리스어로 κυβερνήτης 입니다 – 배에 있는 키잡이 (Helmsman)를 의미합니다.

Docker 컨테이너가 항구/해안을 테마로 했던 것과 비슷하게, **Kubernetes**는 컨테이너가 실어지는 배 운항을 담당하는 항해사를 테마로 합니다..

History (짧은 역사)

Google에서 Borg를 오픈 소스화 + 지속적인 기여 중

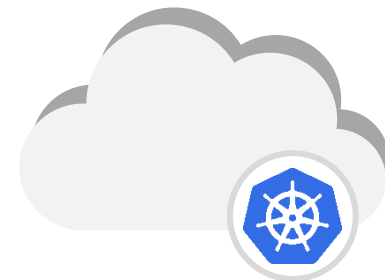
Kubernetes v1.0: 2015년 7월 21일 릴리즈 (Founder: Joe Beda, Brendan Burns
Craig McLuckie)

GitHub를 메인 저장 공간으로 사용 중. 기여자: >1,700;

매 3-6개월마다 릴리즈 중

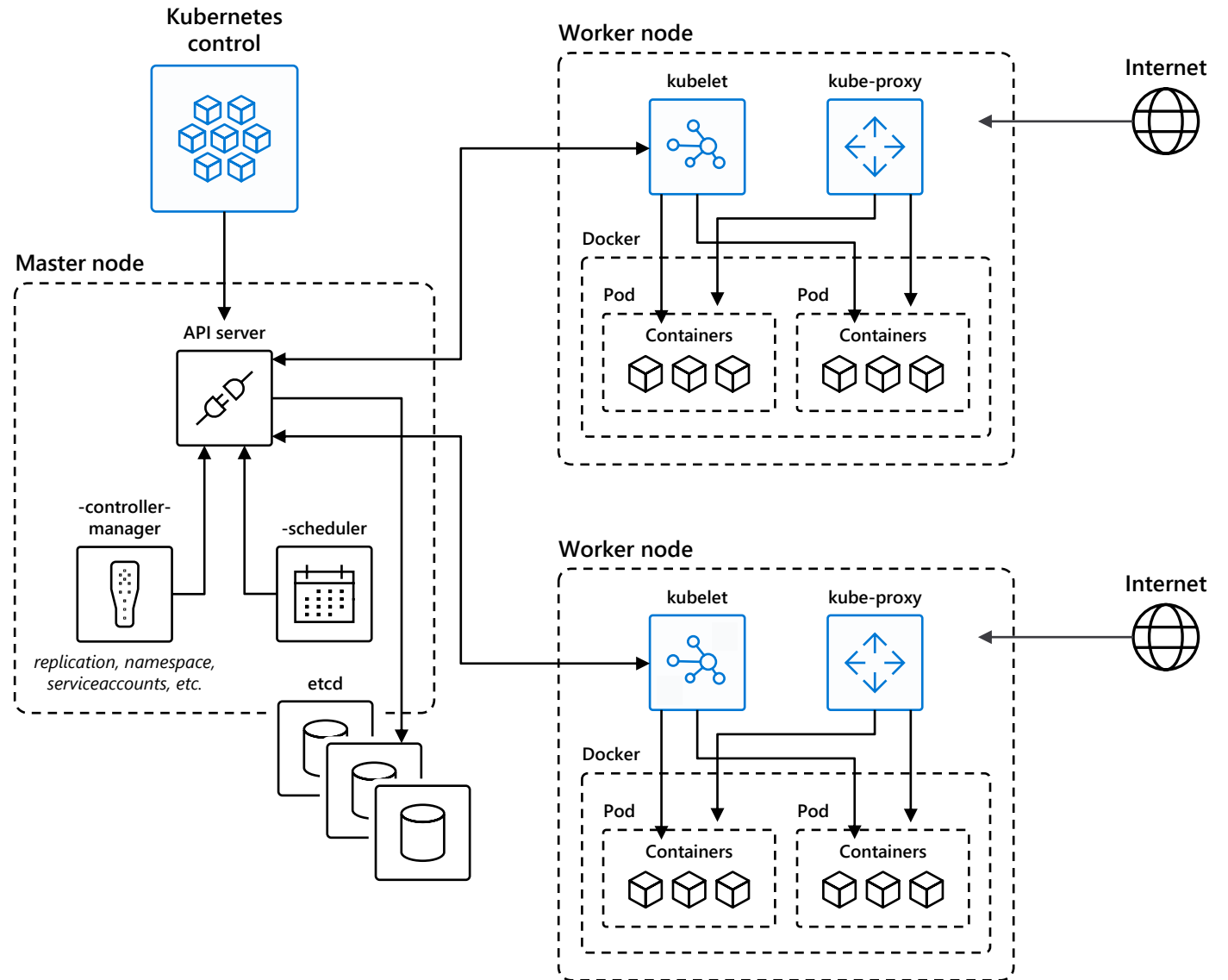
Kubernetes 관련 자세한 배경 & 아이디어:

[Large-scale cluster management at Google with Borg](#) 논문 참고



Kubernetes 101

1. Kubernetes 사용자: API 서버와 통신을 하여 "state"를 가짐
2. Master node: worker node들이 해당 "state"를 가지도록 설정 & 보장
3. Worker node: 컨테이너를 직접적으로 관리
4. 또한 인터넷 연결에 대한 관리가 Worker node에서 필요



Azure Kubernetes Service

Korea, 2020



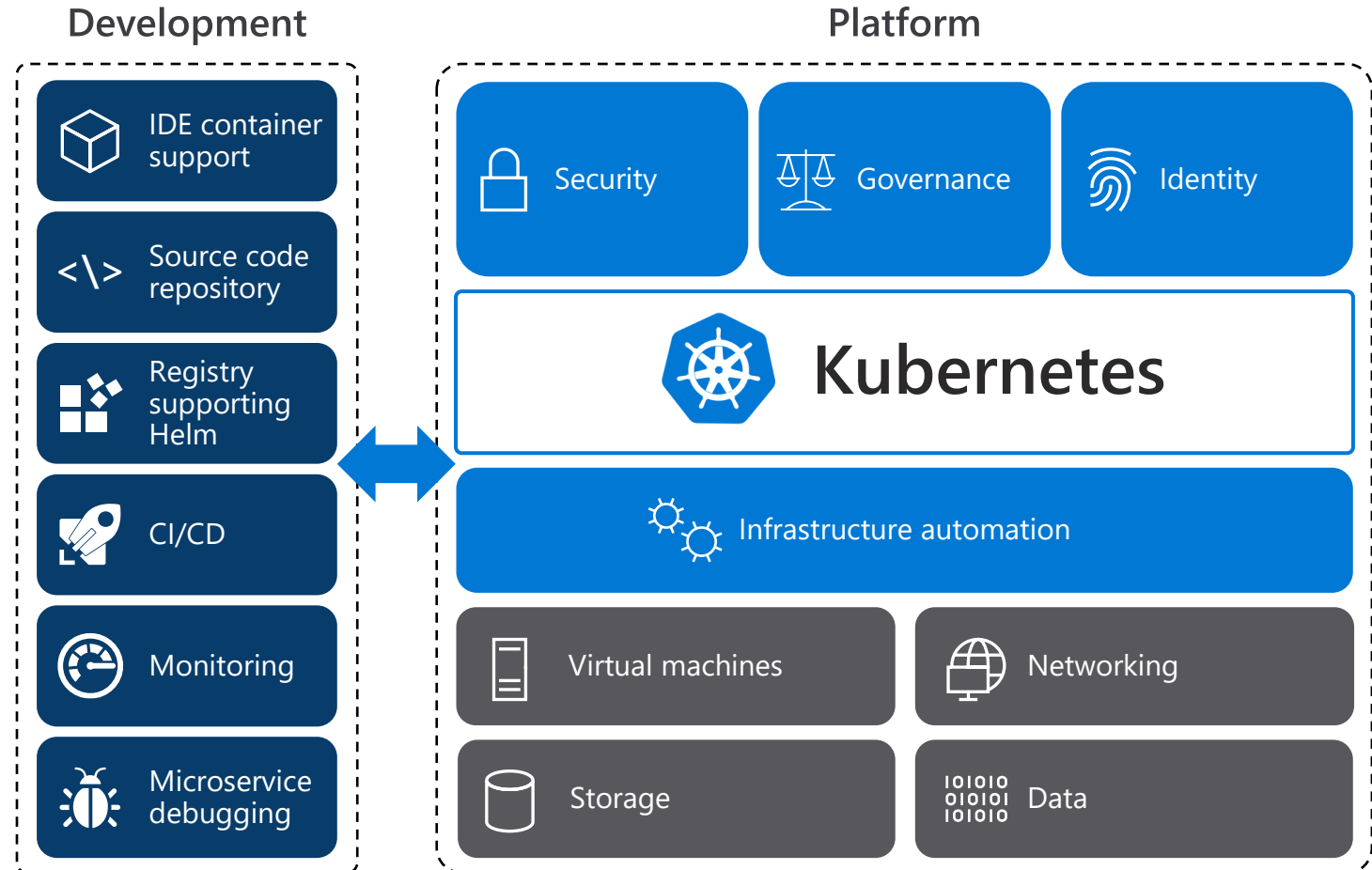
그런데 말입니다

컨테이너만으로 안되어 Kubernetes를 쓰는데, 그것만으로도 **충분하지 않다?!**

보안 안정성을 위해 인프라 관리,
지속적인 업데이트를 빠르게 해야
하는데.. 이 시간을 아껴야 한다!!

컨테이너화된 환경에 Agility를 추가해 봅시다:

- **인프라 자동화**: 프로비저닝, 패치, 업그레이드 단순화
- **컨테이너화된 앱 개발, 그리고 CI/CD 워크플로우에 적합한** 도구 및 툴
- **보안, 거버넌스, 인증, 액세스 관리**를 지원하는 서비스



마이크로소프트 & 컨테이너에서의 변화

Brendan Burns, co-founder of Kubernetes joins
Microsoft Azure team

by Pradeep @pradeepviswam Jul 11, 2016 at 18:36 GMT

A solid blue rectangle with the text "Microsoft Azure" in white, sans-serif font, centered horizontally and vertically.

Brendan Burns, Software Engineer and a founder of Kubernetes at Google is joining Microsoft. Brendan will be joining Azure team where he will be working as development manager for Azure Resource Manager and he will continue to work on Kubernetes too.

A screenshot of a Twitter post from the user 'brendandburns' (@brendandburns). The post text reads: 'Big changes for me! Tomorrow I start at Microsoft Azure. Continuing to work on Kubernetes and helping with Azure Resource Manager and more.' It shows 315 likes and was posted at 9:15 AM on Jul 11, 2016. At the bottom, it says '171 people are talking about this' with a right-pointing arrow.

brendandburns
@brendandburns

Big changes for me! Tomorrow I start at Microsoft Azure.

Continuing to work on Kubernetes and helping with Azure Resource Manager and more.

315 9:15 AM - Jul 11, 2016

171 people are talking about this

2016년 7월,
Kubernetes Founder 중
1명인 Brendan Burns,
마이크로소프트로

(이후 정말 많은 변화가...)

기존 WebApp에 대한 컨테이너
지원, IoT, 머신 러닝 서비스
컨테이너 지원 + 쉬운 쿠버네티스
연동 등 컨테이너 환경에 최적인
"Azure" 클라우드로 변신 중

Container



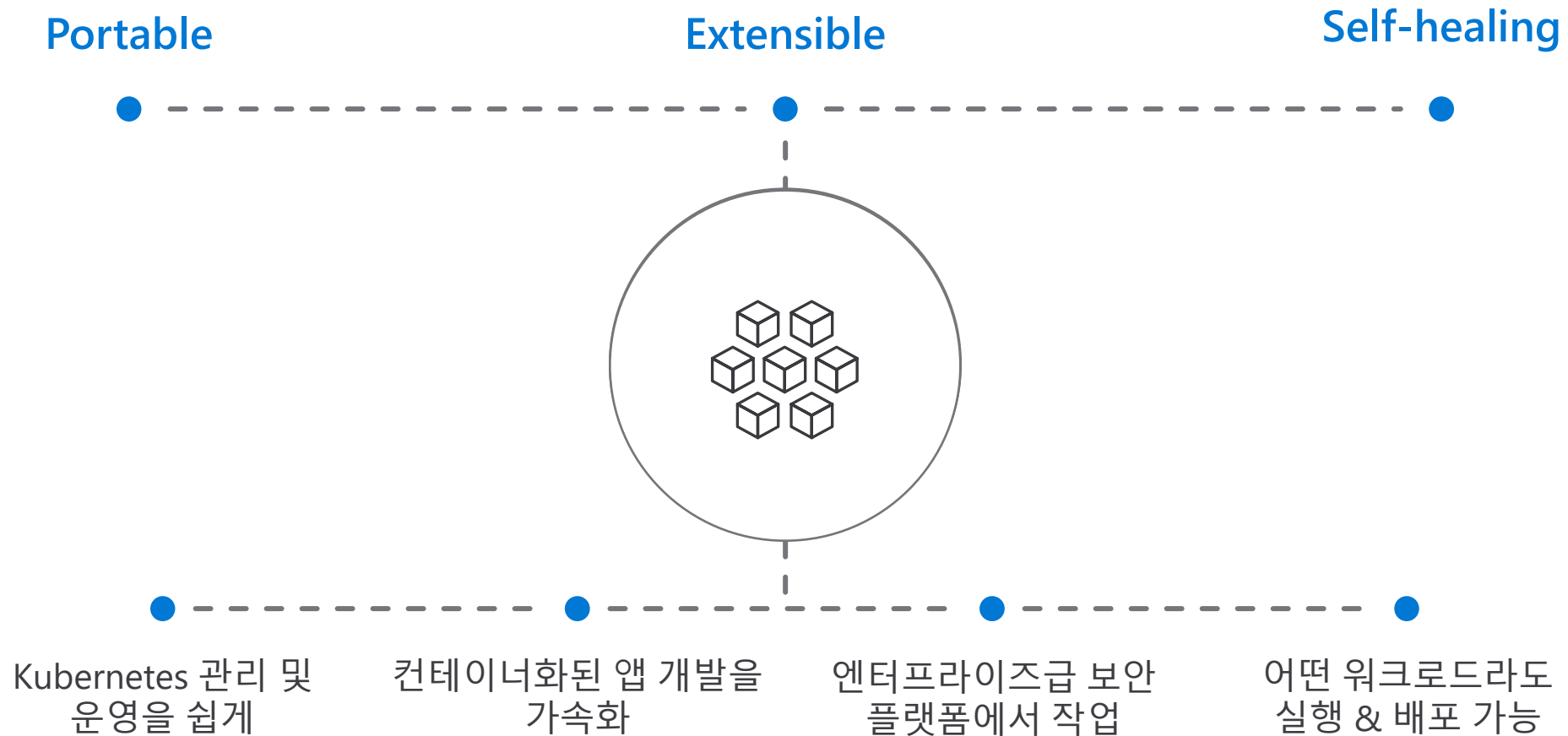
Windows



Linux

Kubernetes on Azure

















Kubernetes에 대한 배포, 관리 및 운영을 편하게 해 줍니다





인프라에서 혁신으로: Azure Kubernetes Service

Managed Kubernetes empowers you to do more

Focus on your containers and code, not the plumbing of them.

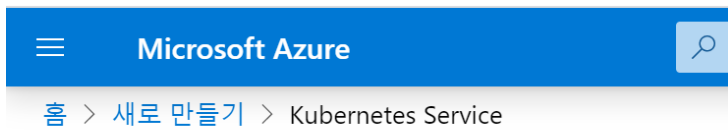
Responsibilities	DIY with Kubernetes	Managed Kubernetes on Azure
Containerization		
Application iteration, debugging		
CI/CD		
Cluster hosting		
Cluster upgrade		
Patching		
Scaling		
Monitoring and logging		

 Customer

 Microsoft

Azure Kubernetes Service (AKS)

오픈 소스 Kubernetes 사용 & 운영을 직접 Microsoft에서 “관리”



Kubernetes Service

Microsoft



Kubernetes Service

Microsoft

만들기

The screenshot shows the 'Overview' page of the Kubernetes Dashboard. The browser address bar indicates the URL '127.0.0.1:8001/#/overview?namespace=default'. The dashboard has a blue header with the 'kubernetes' logo and a search bar. A left sidebar lists various Kubernetes resources. The main content area is titled 'Overview' and features three green circular progress indicators for 'Deployments', 'Pods', and 'Replica Sets', all showing '100.00%'. Below these are three tables: 'Deployments', 'Pods', and 'Replica Sets'.

Cluster

- Namespaces
- Nodes
- Persistent Volumes
- Roles
- Storage Classes

Namespace

default

Overview

Workloads

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets

Discovery and Load Balancing

- Ingresses
- Services

Config and Storage

- Config Maps

Workloads Statuses

- Deployments: 100.00%
- Pods: 100.00%
- Replica Sets: 100.00%

Deployments

Name	Labels	Pods	Age	Images
gs-spring-boot-docker	run: gs-spring-boot-docker	1 / 1	22 hours	ianaksspringwinregistry.azurecr.io/gs-...

Pods

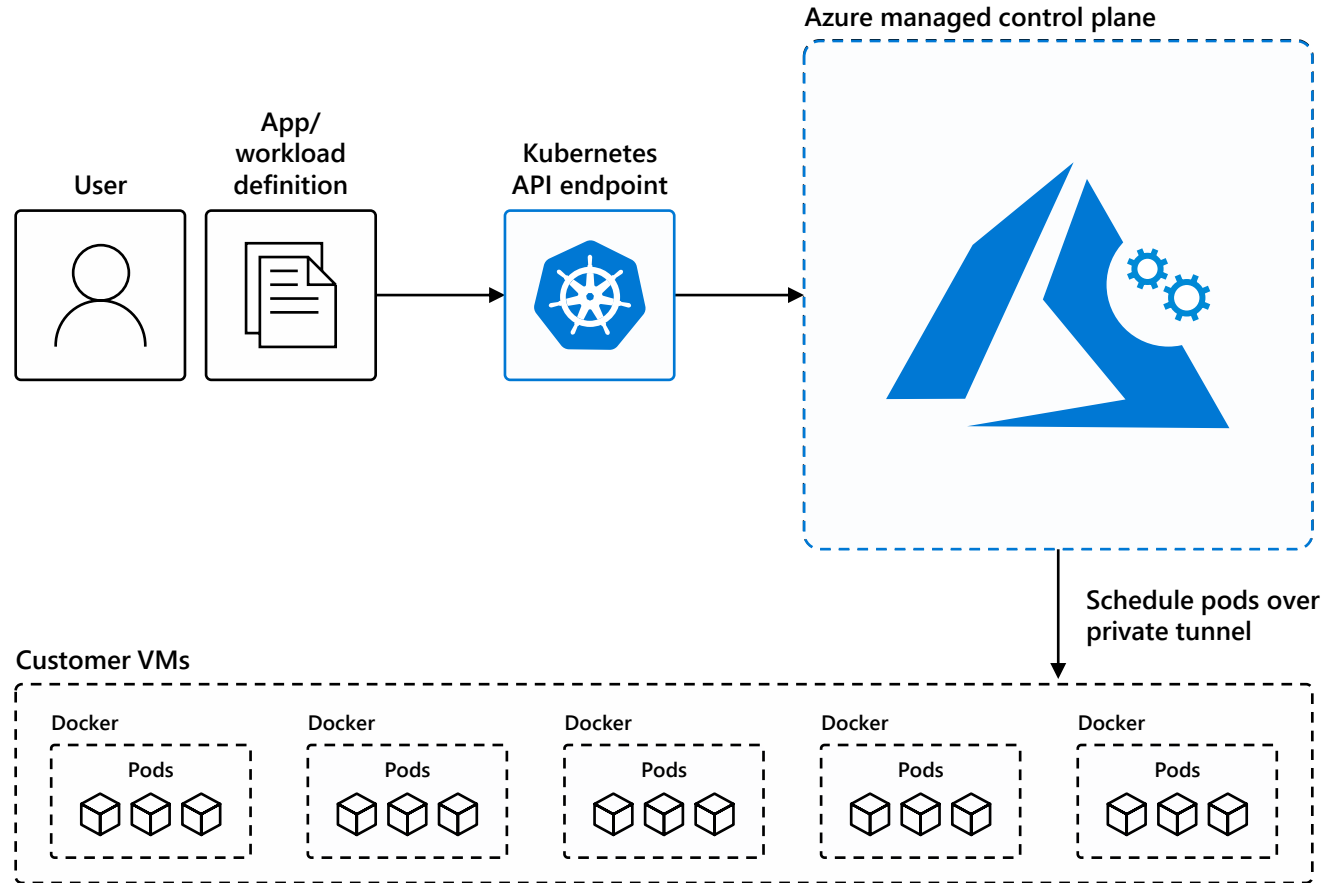
Name	Node	Status	Restarts	Age
gs-spring-boot-docker-7c6c4cddc5-jvbmj	aks-nodepool1-12766016-0	Running	0	22 hours

Replica Sets

Name	Labels	Pods	Age	Images
gs-spring-boot-docker-7c6c4cddc5	pod-template-hash: 7c6c4cddc5 run: gs-spring-boot-docker	1 / 1	22 hours	ianaksspringwinregistry.azurecr.io/gs-...

How managed Azure Kubernetes Service

- 자동 업그레이드, patches
- 높은 신뢰성, 가용성(availability)
- 쉽고 안전한 클러스터 scaling
- 자기 치유(Self-healing)
- API server monitoring
- Control Plane 비용 추가가 없음(Master Node 무료!)
- GPU Support
- 100% upstream Kubernetes



Azure Container Registry 만들기

Korea, 2020



Azure Kubernetes Cluster 만들기

Korea, 2020



Application 배포하기

Korea, 2020



배포된 Application 모니터링하기

Korea, 2020



Resources

- Kubernetes 101 Documentation – <https://aka.ms/LearnAKS>
- Kubernetes Best Practices – <https://aka.ms/aks/bestpractices>
- Field SME Webinars – <https://aka.ms/aks/SMEWebinars>
- AKS roadmap - <https://github.com/Azure/AKS/projects/1>