



Computer Network

Lecture 11

Cloud Computing

2019. 03. 01

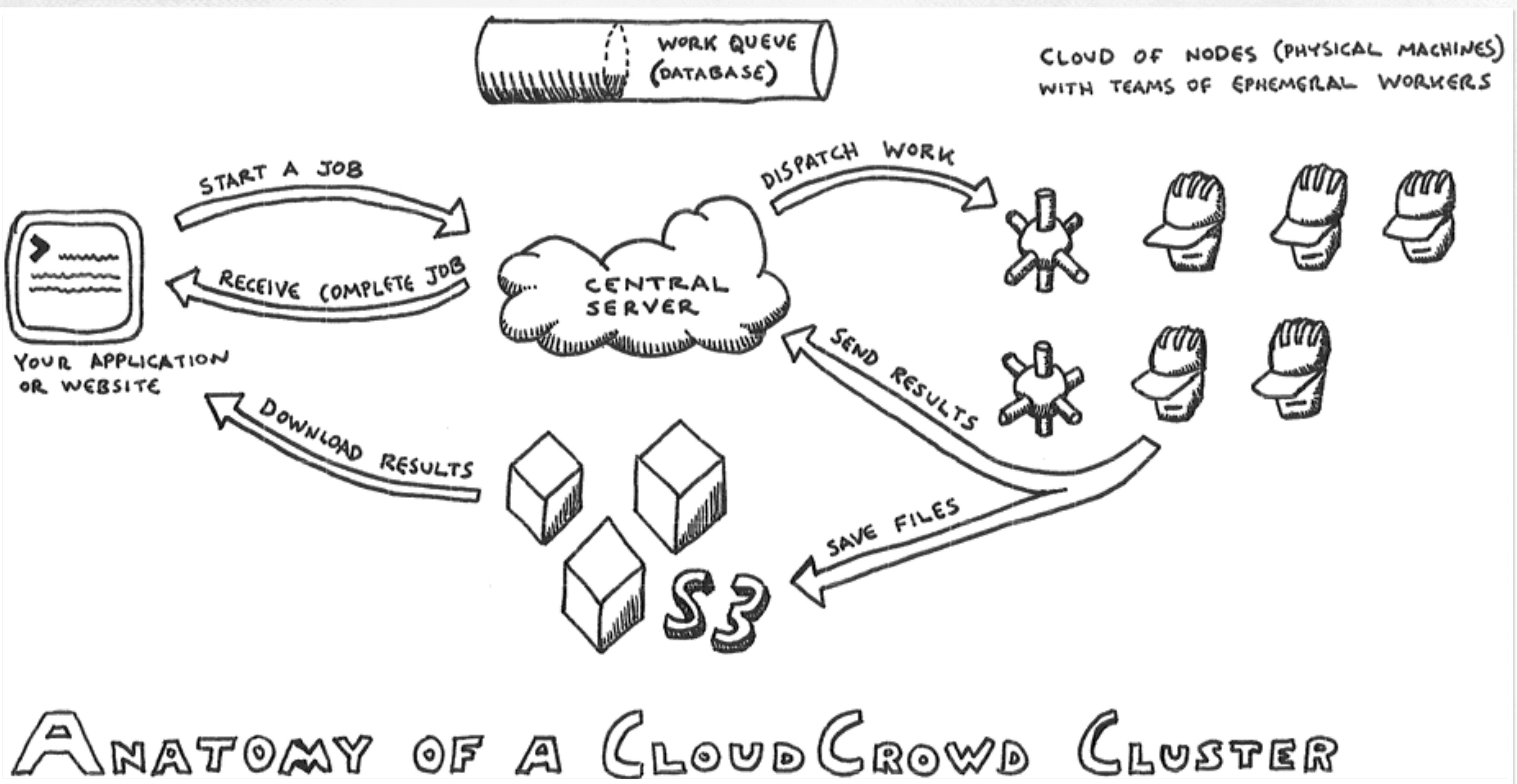
Sungwon Lee
Department of Software Convergence

Contents

- **Introduction to Cloud Computing**
- Public Cloud Services
- Cloud Computing APIs
- Thin Client
- Private Cloud Solutions
- Beyond Cloud Computing

Basic Idea

General Idea of Cloud Computing

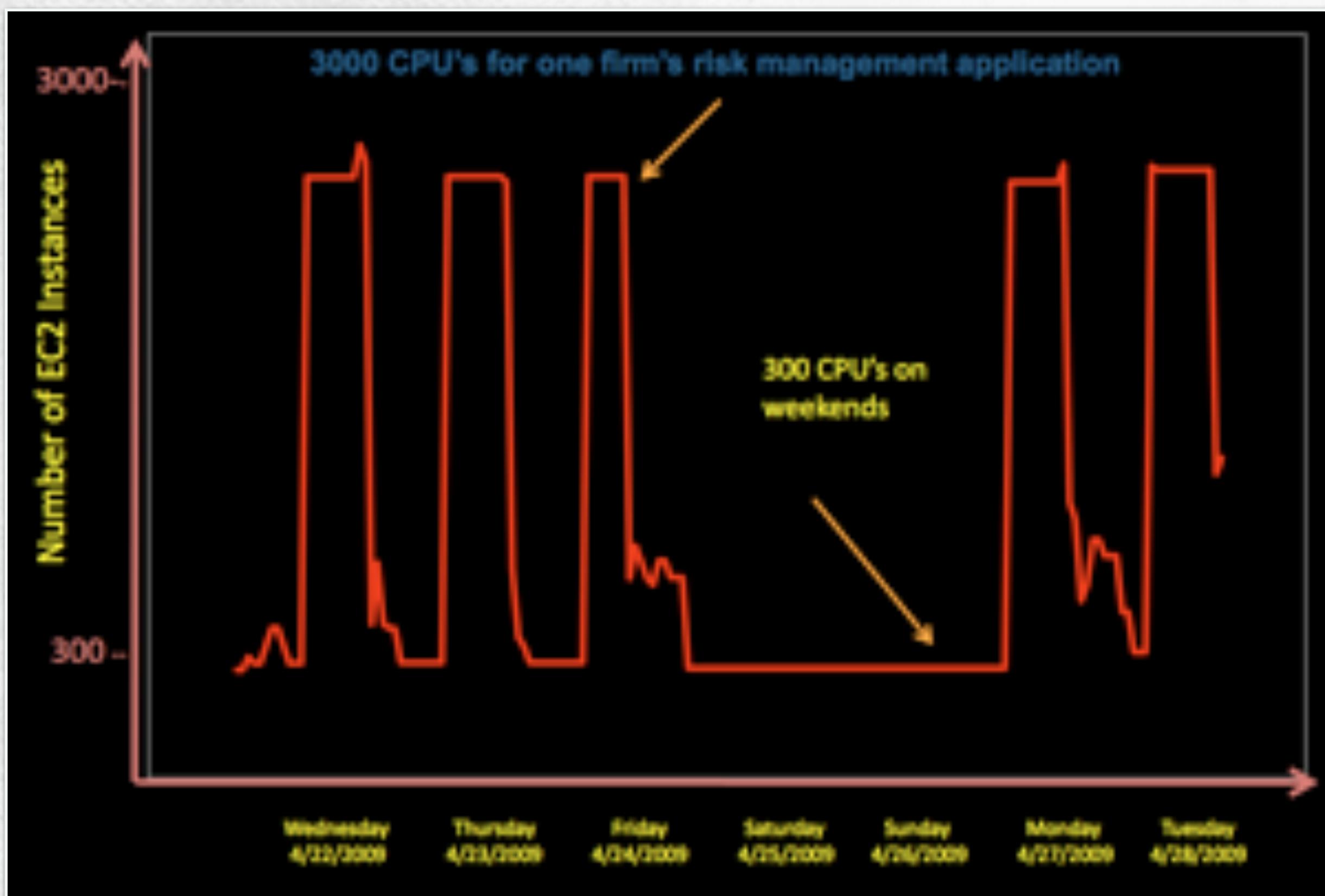


Background

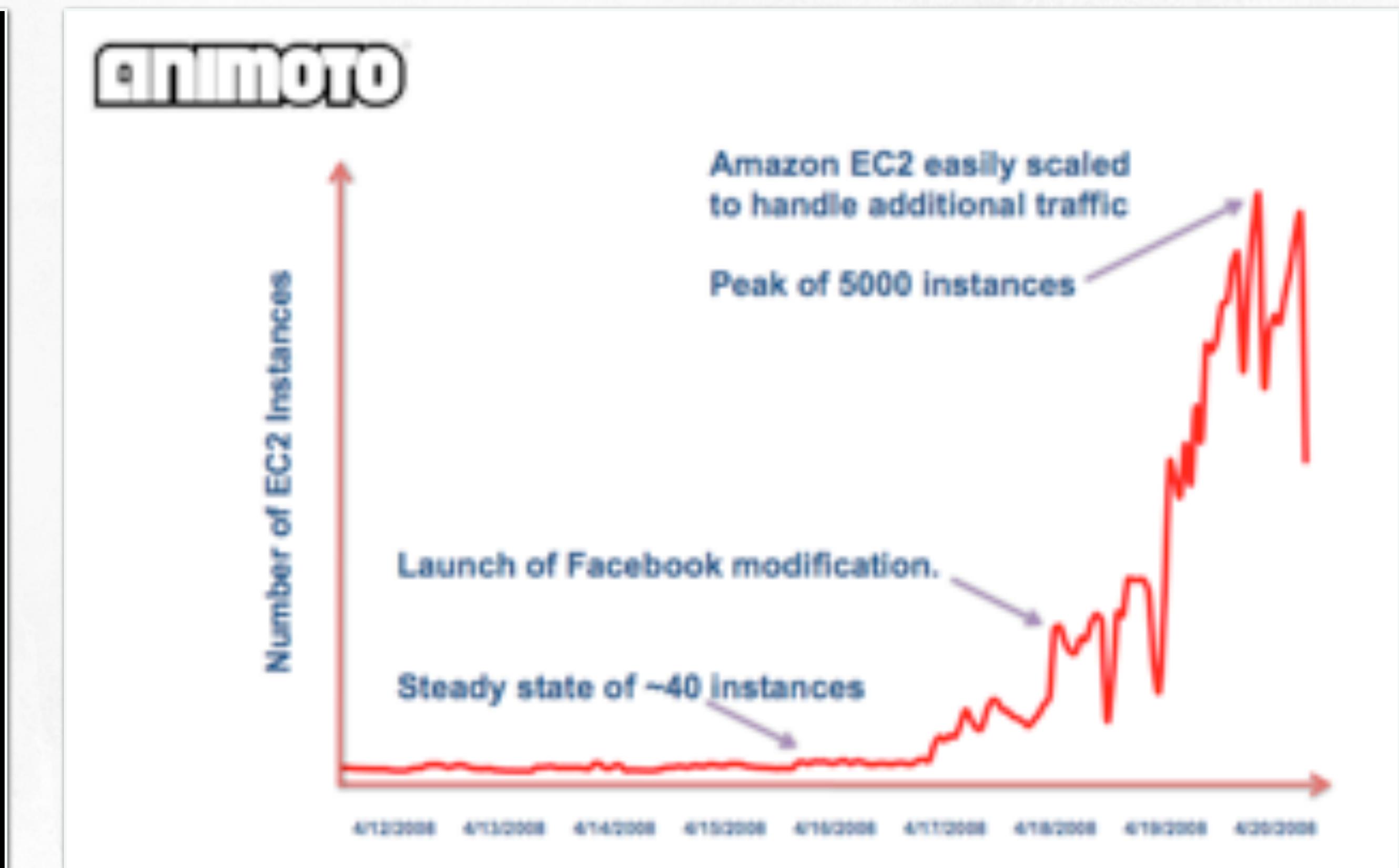
Why Cloud Computing?

- Explosion of data volume

- Easily mention about Petabyte, Exabyte, and Big-Data
- Compute and Storage limitation for private organization



Cloud Provider's View



Cloud Customer's View

Why Cloud Computing?

- Reliability on Hardware and Software

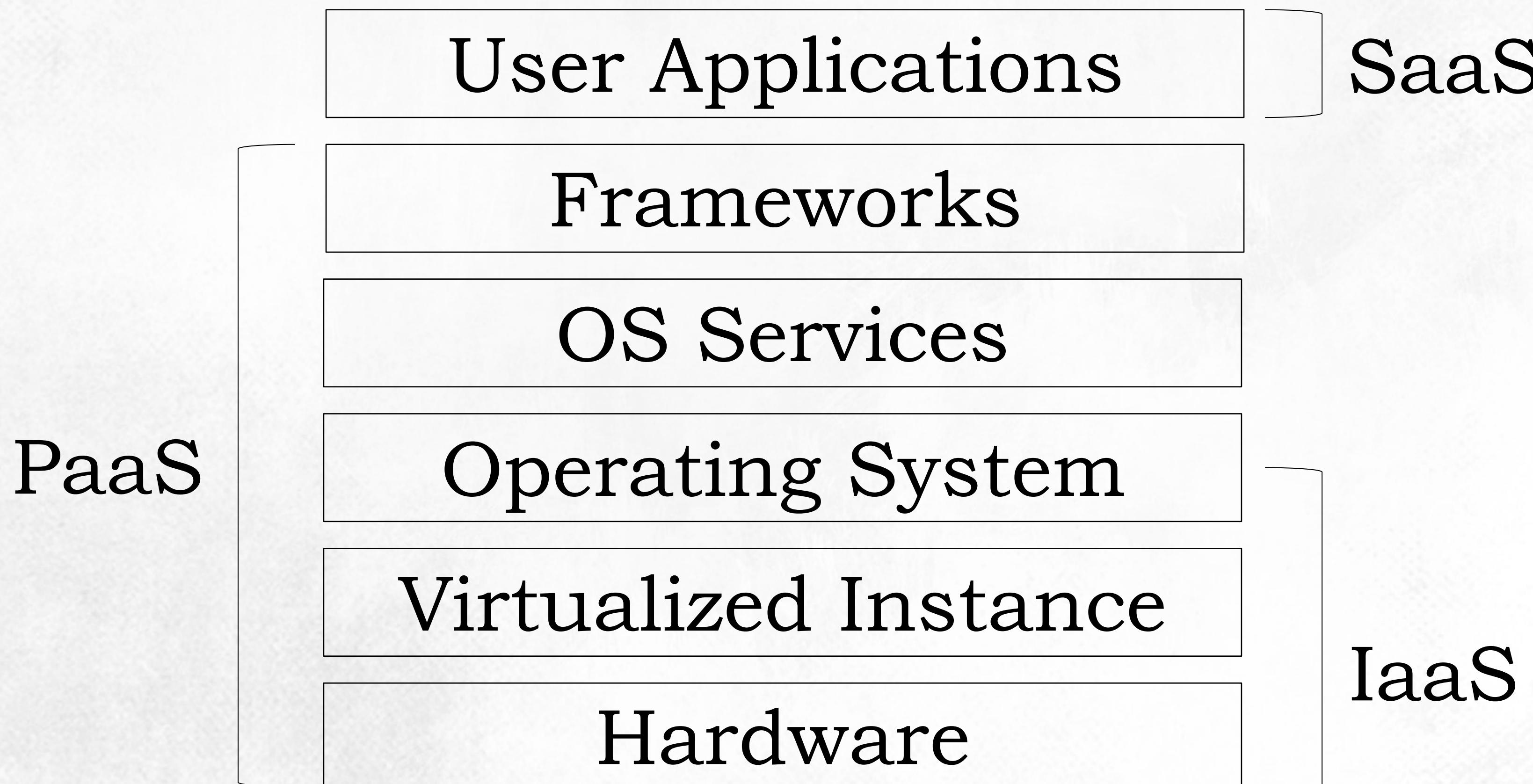
- 2-4% of servers will die annually
- 1-5% of disk drives will die every year
- 20% admin issues have unintended consequences (Human errors)

Cloud Computing Types

XaaS

- Well known cloud computing approaches

- Infrastructure as a Service (**IaaS**)
- Platform as a Service (**PaaS**)
- Software as a Service (**SaaS**)



Cloud Computing Types

Let's visit and Enjoy free trials !!

- Well known cloud computing approaches
 - Infrastructure as a Service (IaaS)
 - Amazon Web Services (<https://aws.amazon.com/ko/>)
 - Platform as a Service (PaaS)
 - Microsoft Azure (<https://azure.microsoft.com/ko-kr/free/>)
 - Google App Engine (<https://cloud.google.com/appengine/>)
 - Software as a Service (SaaS)
 - Microsoft Office Live (<https://office.live.com/start/Word.aspx>)
 - Microsoft Office Online (<https://products.office.com/ko-kr/office-online/>)
 - Google Apps (https://gsuite.google.com/intl/en_sg/)

Build Types

Public or Private

- Public cloud computing solutions
 - Hosted by global IT companies through huge scale CPU/Disks
 - Proprietary interfaces with own technical solutions
 - Examples
 - Amazon Web Service (IaaS)
 - MS Azure, Google App Engine (PaaS)
 - MS Office Live, Google Apps (SaaS)

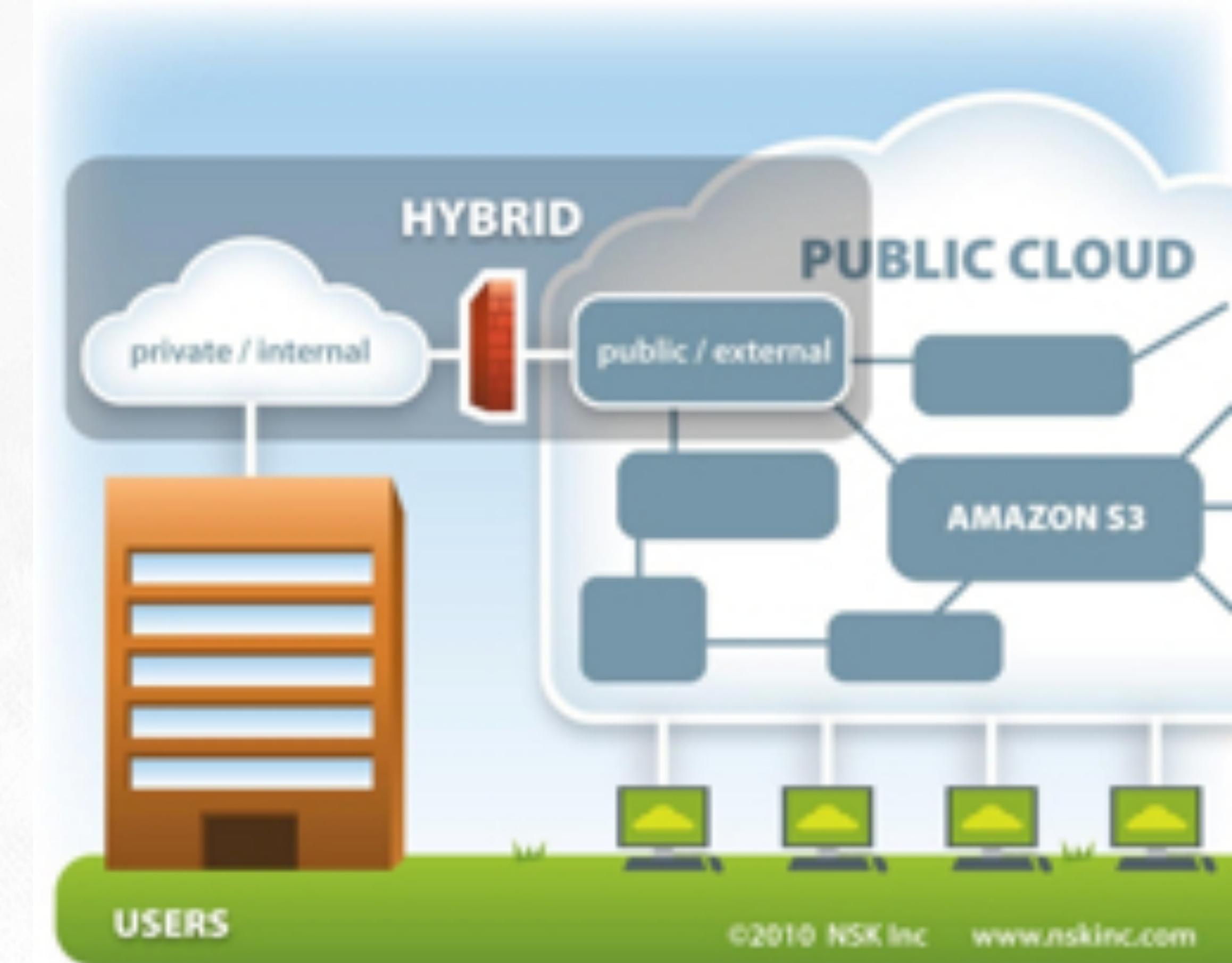
- Private cloud computing solutions
 - Hosted by private companies for own cloud construction
 - Open sources are widely used
 - Examples: OpenStack (opensource software)

Build Types

Hybrid

- Public cloud to Hybrid cloud

- Organization wants to have control over their data storage, but needs additional space for archiving data
- They have the security and supervision of the Private Cloud in their network, but can store excess data in a scalable on-demand Public Cloud

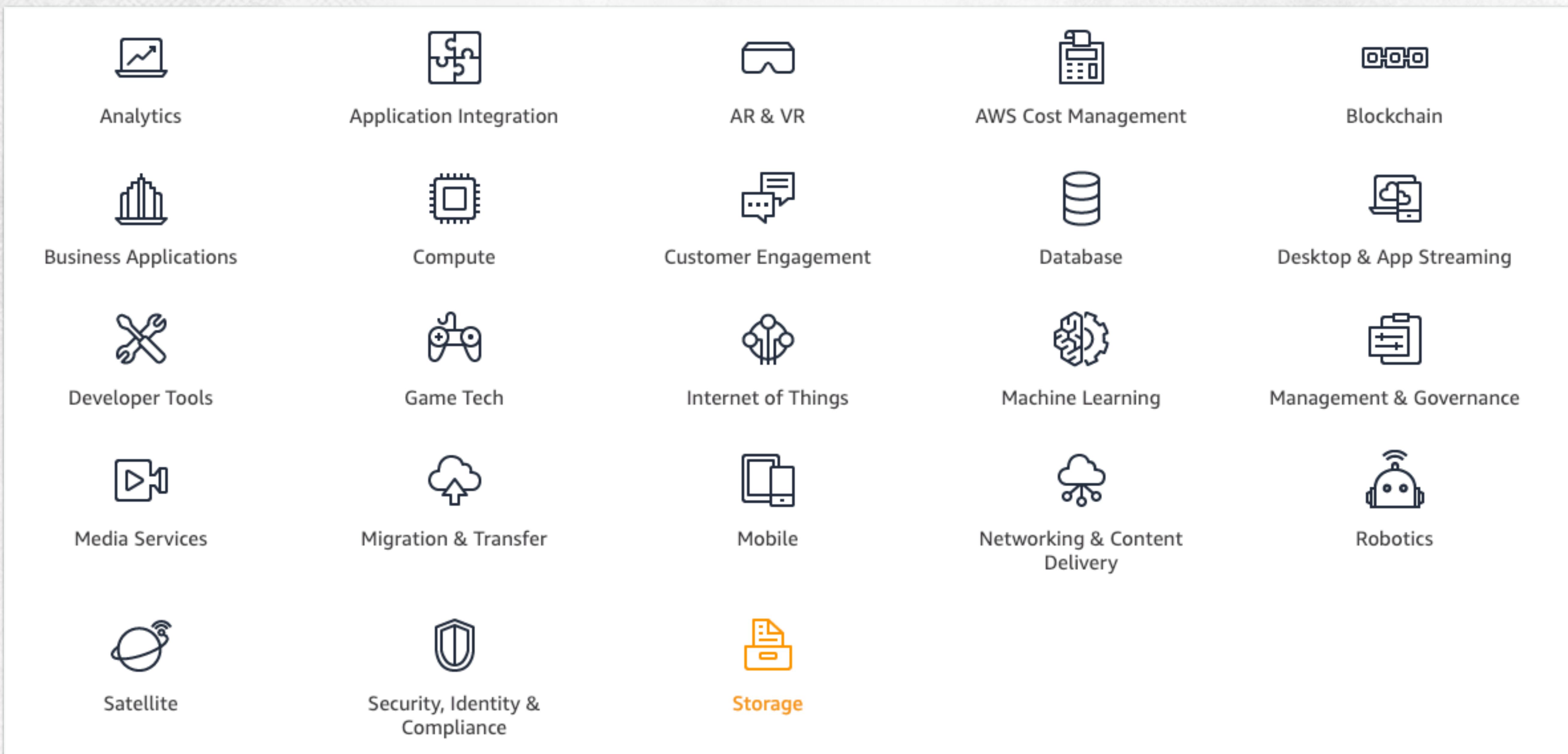


Contents

- Introduction to Cloud Computing
- **Public Cloud Services**
- Cloud Computing APIs
- Thin Client
- Private Cloud Solutions
- Beyond Cloud Computing

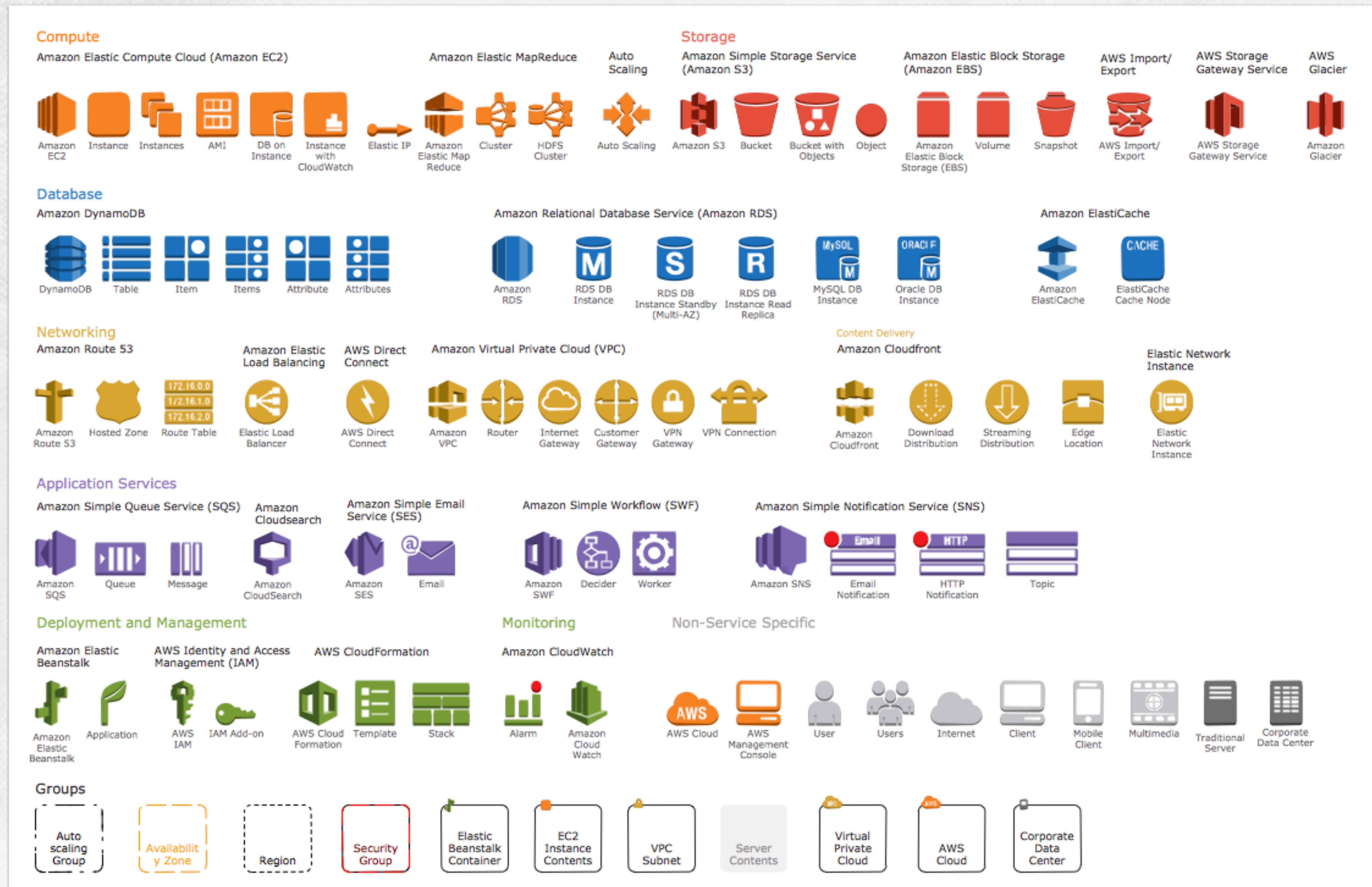
Public Cloud Computing Services

Amazon Web Service (AWS)



Public Cloud Computing Services

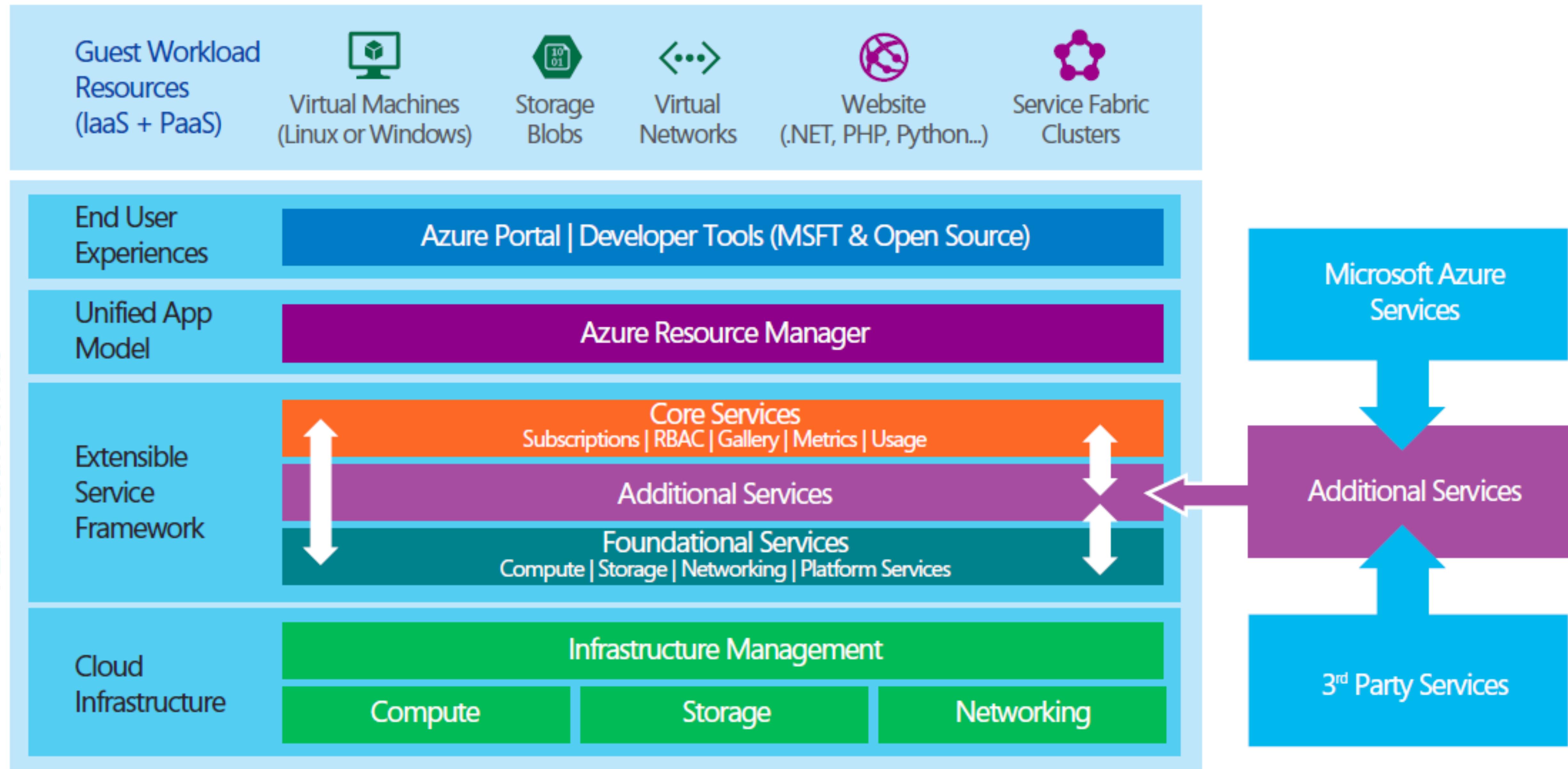
Amazon Web Service (AWS)



Public Cloud Computing Services

Microsoft Azure

Azure Stack architecture summary



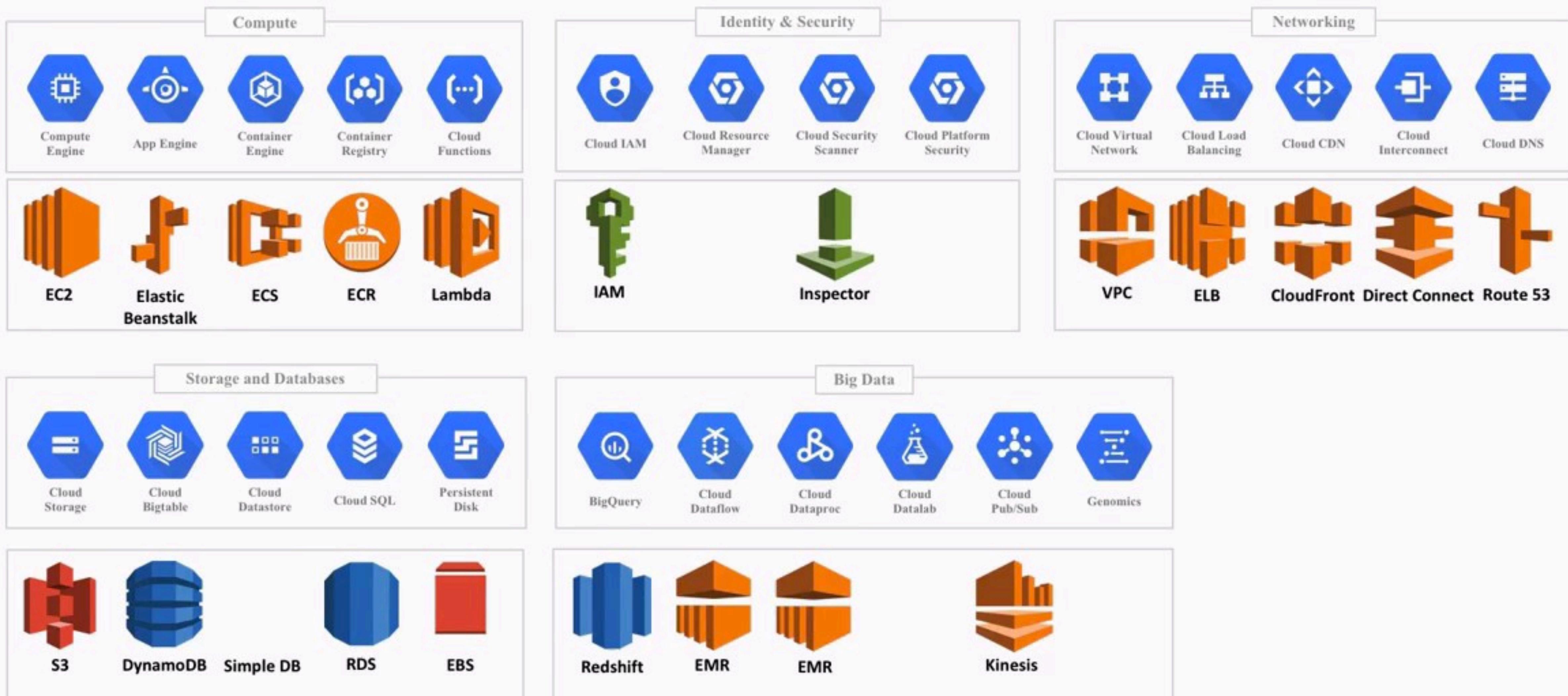
Public Cloud Computing Services

Google (1/4)



Linux Academy

How do these services compare?

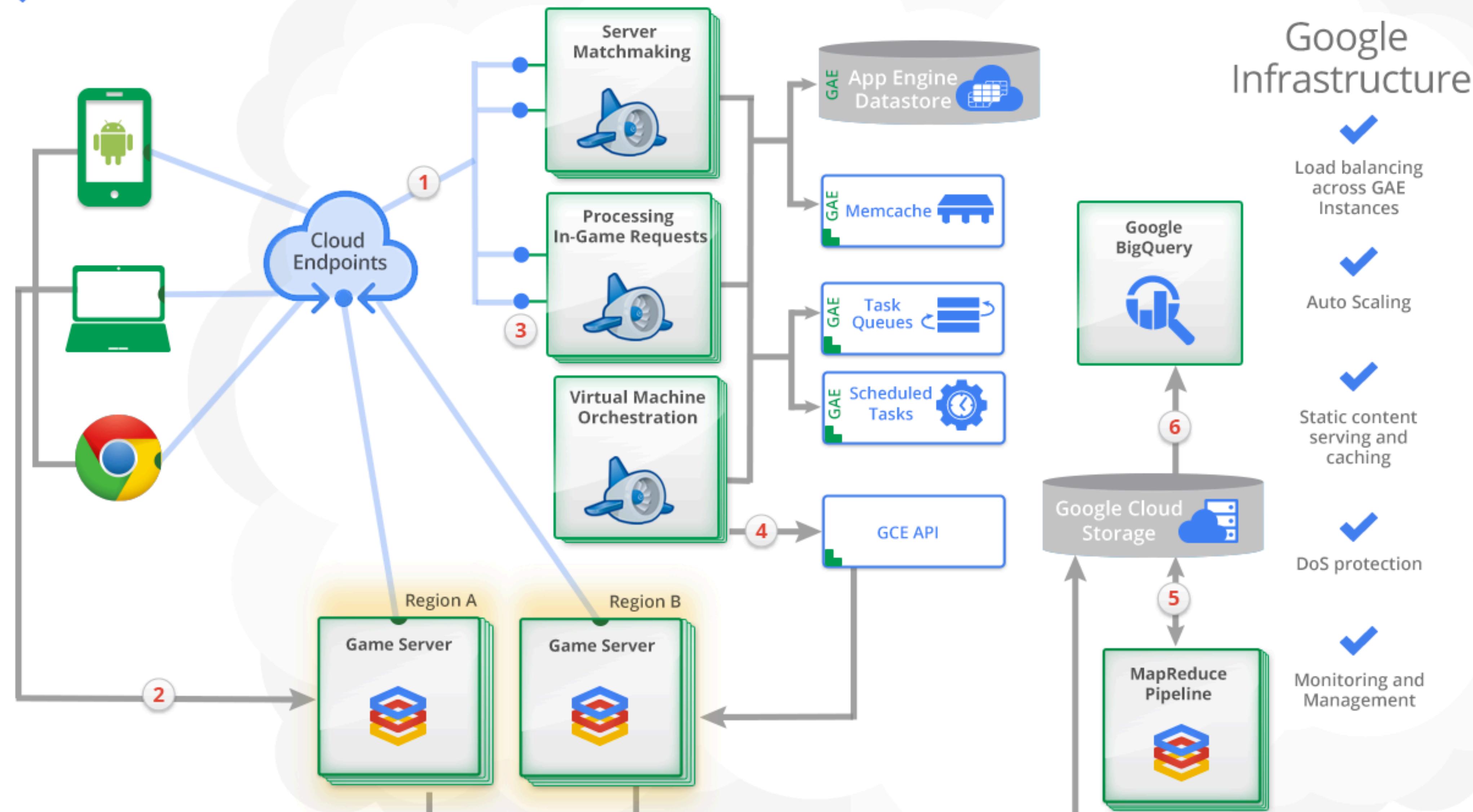


Public Cloud Computing Services

Google (2/4 - Game Server Example)

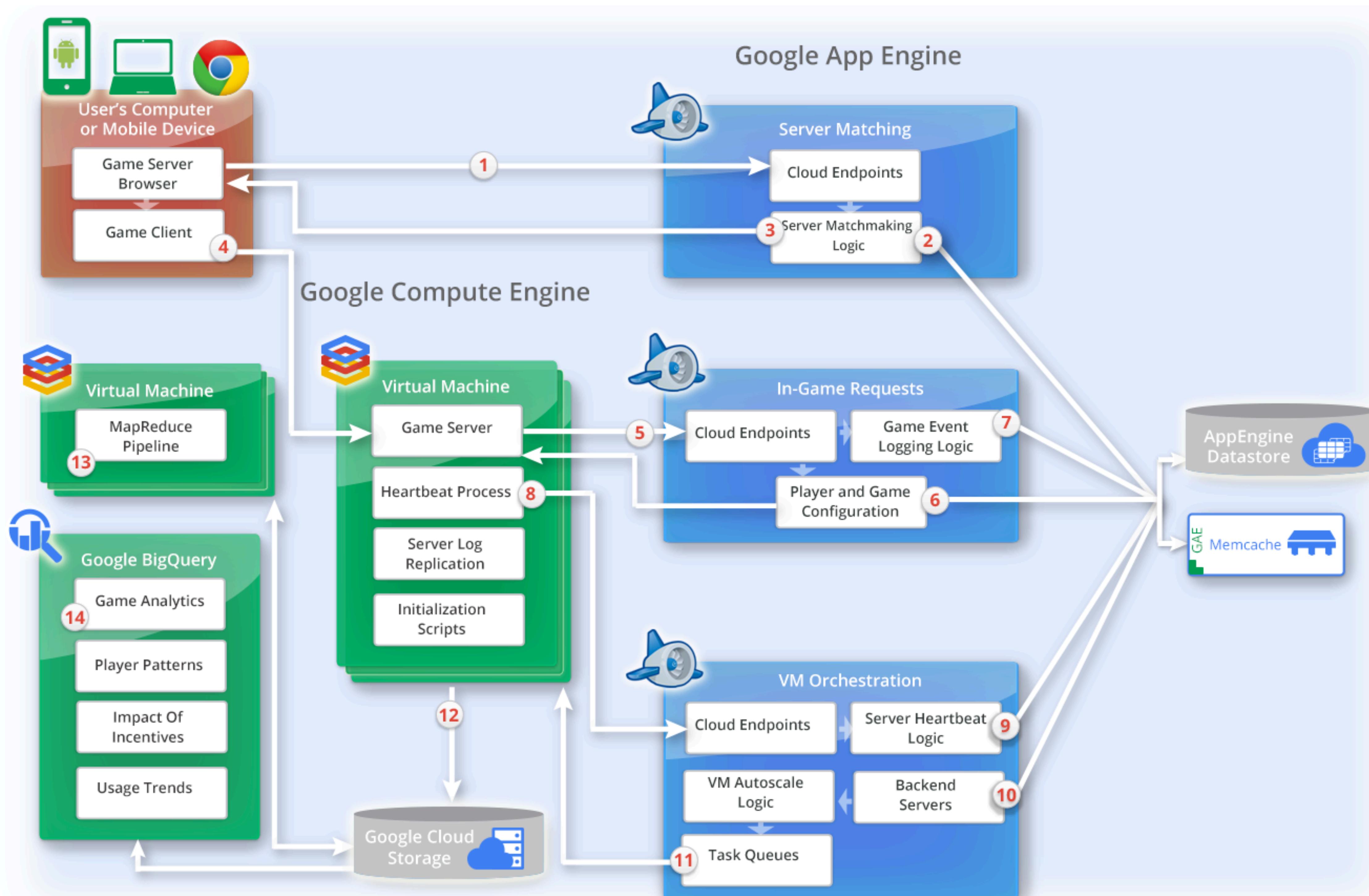
Dedicated Server Gaming Solution on the Google Cloud Platform

- Your Application Code running on Google App Engine (GAE), Google Compute Engine (GCE), and Client Devices
- Google Cloud Platform Services
- Capabilities Included



Public Cloud Computing Services

Google (3/4 - Game Server Example)

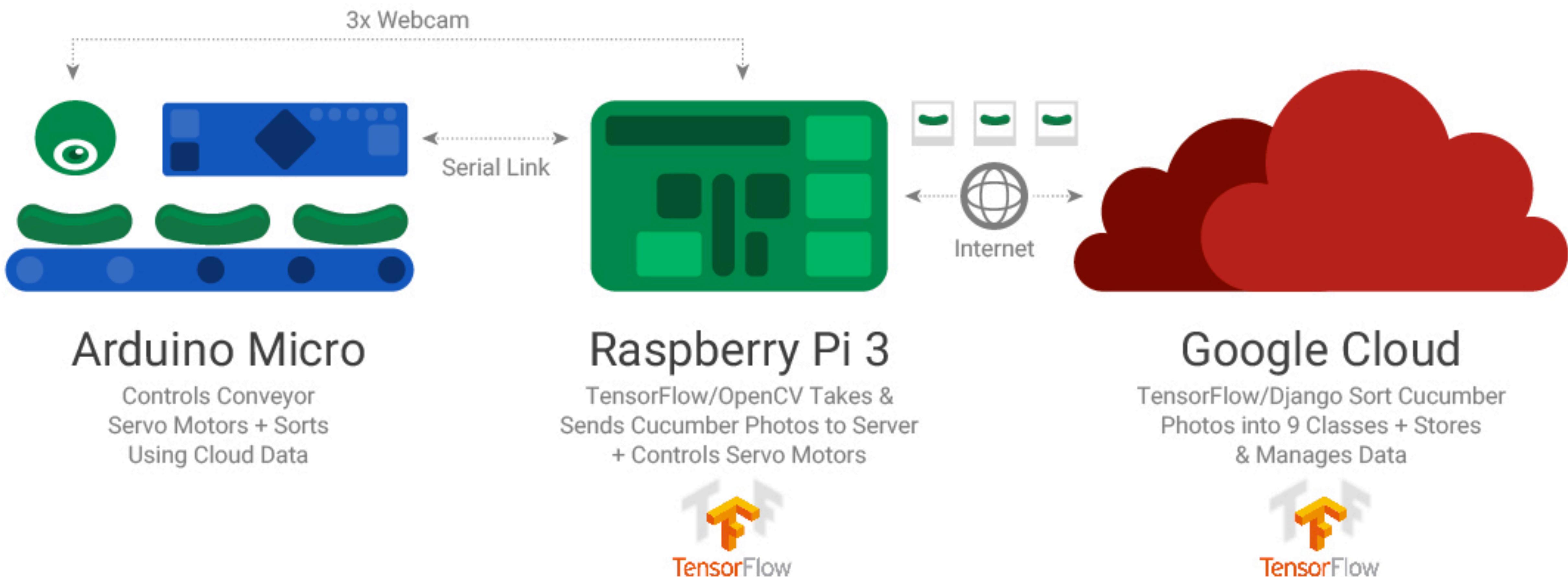


Public Cloud Computing Services

Google (4/4 - Smart Farming: Let's Visit)

“How a Japanese cucumber farmer is using deep learning and TensorFlow”

<https://cloud.google.com/blog/big-data/2016/08/how-a-japanese-cucumber-farmer-is-using-deep-learning-and-tensorflow#showImage>



Different Approach

APPLE

- Steve Job's said:

“10여 년 전 우리는 가장 중요하다고 할 수 있는 통찰을 갖고 있었습니다. PC가 디지털 라이프의 허브가 된다는 것이었지요. 동영상, 사진, 음악의 허브가 된다고 말입니다.

하지만 지난 몇 년 동안 그것이 무너졌습니다.

왜 그랬을까요?

모든 콘텐트를 여러 가지 기기에 동시에 동기화 하기가 어렵습니다. 아이패드에 다운로드 한 노래, iPhone으로 찍은 사진, 컴퓨터에 저장한 동영상 등과 같은 콘텐트를 다른 기기에서 이용하려면 USB 케이블을 꽂았다 뺏다 하면서 마치 옛날의 전화교환원이 된 기분을 느껴야 합니다.

우리가 해결책을 갖고 있습니다.

그 해결책이 우리의 두 번째 커다란 통찰이지요.

우리는 PC와 맥을 그저 일개 기기로 전락시킬 것입니다. 그리고 디지털 허브를 클라우드로 바꿀 겁니다.”

Different Approach APPLE (before iCloud)



Different Approach

APPLE (iCloud @ Oct. 2011)

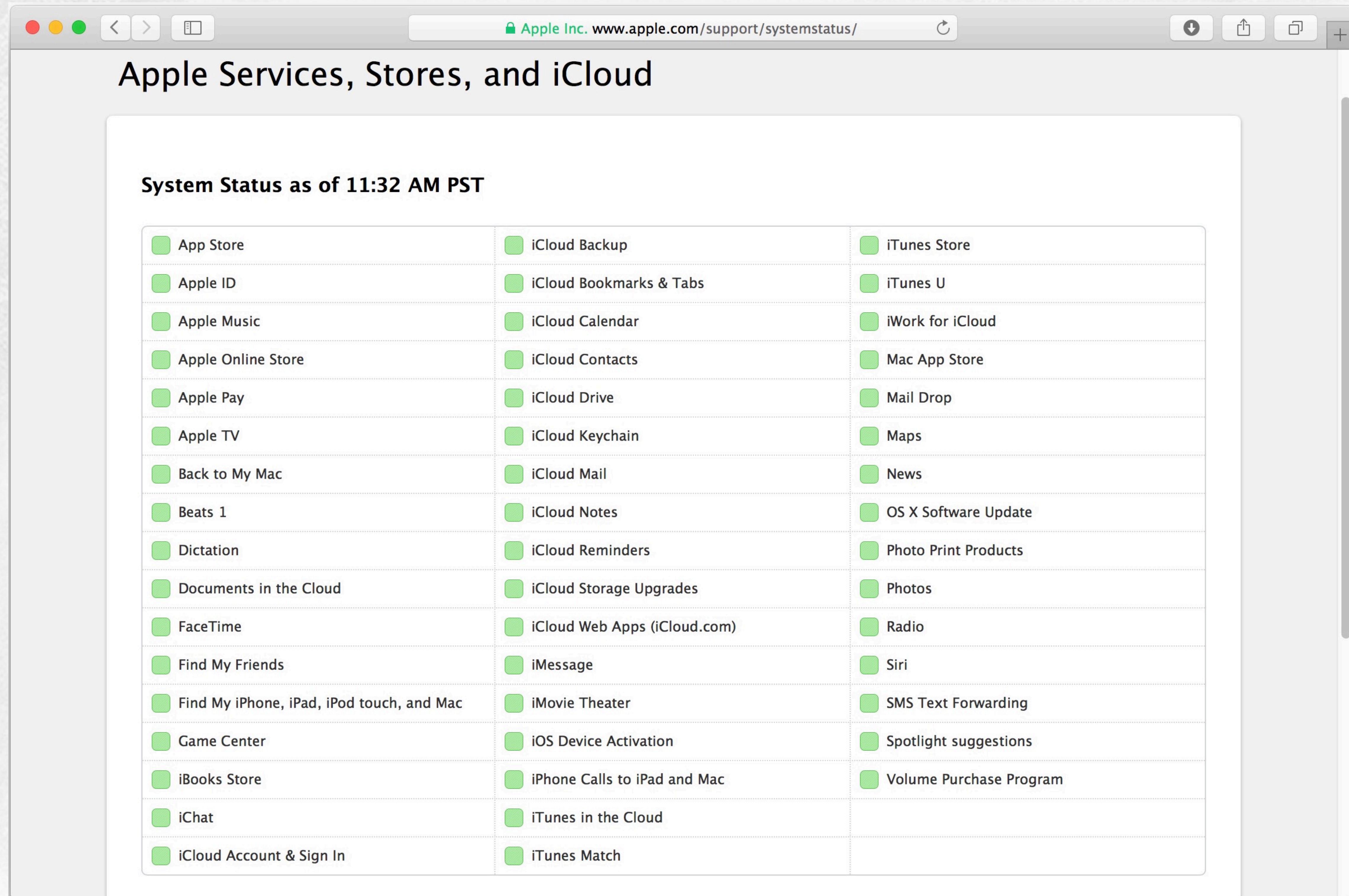


APPLE ID and iCloud Features (@ 2011)

- Single ID for
 - Multiple devices
 - iPhone, iPad, iPod, Apple TV, Mac(-intosh), Web-site
 - Whole services
 - Tightly coupled with user's credit & pay-card **account**
 - **Purchasing** music, movie, book and application
 - **Synchronization** of contents, files and data between devices
 - Supporting multimedia messaging and video **communication**
 - Automated adjacent device aware and content **sharing**
 - Automated and background device **backup** service
 - **Location** tracking and finding service
 - Automated remote Mac aware and **remote desktop** service

Different Approach

APPLE iCloud Features (@ 2017)



The screenshot shows a web browser window displaying the Apple System Status page at www.apple.com/support/systemstatus/. The page title is "Apple Services, Stores, and iCloud". Below the title, it says "System Status as of 11:32 AM PST". The content is organized into a grid of service status indicators.

Service	Service	Service
App Store	iCloud Backup	iTunes Store
Apple ID	iCloud Bookmarks & Tabs	iTunes U
Apple Music	iCloud Calendar	iWork for iCloud
Apple Online Store	iCloud Contacts	Mac App Store
Apple Pay	iCloud Drive	Mail Drop
Apple TV	iCloud Keychain	Maps
Back to My Mac	iCloud Mail	News
Beats 1	iCloud Notes	OS X Software Update
Dictation	iCloud Reminders	Photo Print Products
Documents in the Cloud	iCloud Storage Upgrades	Photos
FaceTime	iCloud Web Apps (iCloud.com)	Radio
Find My Friends	iMessage	Siri
Find My iPhone, iPad, iPod touch, and Mac	iMovie Theater	SMS Text Forwarding
Game Center	iOS Device Activation	Spotlight suggestions
iBooks Store	iPhone Calls to iPad and Mac	Volume Purchase Program
iChat	iTunes in the Cloud	
iCloud Account & Sign In	iTunes Match	



Contents

- Introduction to Cloud Computing
- Public Cloud Services
- **Cloud Computing APIs**
- Thin Client
- Private Cloud Solutions
- Beyond Cloud Computing

EC2 (Amazon Elastic Compute Cloud)

- Concept

- allowing users to ***rent virtual computers*** on which to run their own computer applications
- encourages scalable ***deployment of applications by providing a web service*** through which a user can boot an Amazon Machine Image (AMI) to configure a virtual machine, which Amazon calls an "instance", containing any software desired
- provides users with ***control over the geographical location of instances*** that allows for latency optimization and high levels of redundancy

EC2 (Amazon Elastic Compute Cloud)

- Operation

- Select a pre-configured, templated Amazon Machine Image (AMI) to get up and running immediately. Or create an AMI containing your applications, libraries, data, and associated configuration settings.
- Configure security and network access on your Amazon EC2 instance.
- Choose which instance type(s) you want, then start, terminate, and monitor as many instances of your AMI as needed, using the web service APIs or the variety of management tools provided.
- Determine whether you want to run in multiple locations, utilize static IP endpoints, or attach persistent block storage to your instances.
- Pay only for the resources that you actually consume, like instance-hours or data transfer.

EC2 (Amazon Elastic Compute Cloud)

- Features (1/2)

- Bare Metal instances
- Optimize Compute Performance and Cost with Amazon EC2 Fleet
- Pause and Resume Your Instances
- GPU Compute Instances
- GPU Graphics Instances
- High I/O Instances
- Dense Storage Instances
- Optimized CPU Configurations
- Flexible Storage Options
- Paying for What You Use
- Multiple Locations
- Elastic IP Addresses

EC2 (Amazon Elastic Compute Cloud)

- Features (2/2)

- Amazon EC2 Auto Scaling
- High Performance Computing (HPC) Clusters
- Enhanced Networking
- Elastic Fabric Adapter (Fast interconnect for HPC clusters)
- Available on AWS PrivateLink
- Amazon Time Sync Service

EC2 (Amazon Elastic Compute Cloud)

Amazon Machine Images (AMIs)

- Amazon Machine Images (AMIs) are preconfigured with an ever-growing list of operating systems.
- Amazon work with partners and community to provide you with the most choice possible.
- Users are also empowered to use Amazon's bundling tools to upload own operating systems.
- The operating systems currently available to use with your Amazon EC2 instances include:



S3 (Amazon Simple Storage Service)

- Amazon Simple Storage Service

- provides ***storage through web services interfaces*** (REST, SOAP, and BitTorrent)
- uses the ***same scalable storage infrastructure that Amazon.com uses*** to run its own global e-commerce network
- uses include ***web hosting, image hosting, and storage for backup*** systems
- guarantees 99.9% monthly uptime ***service-level agreement (SLA)***

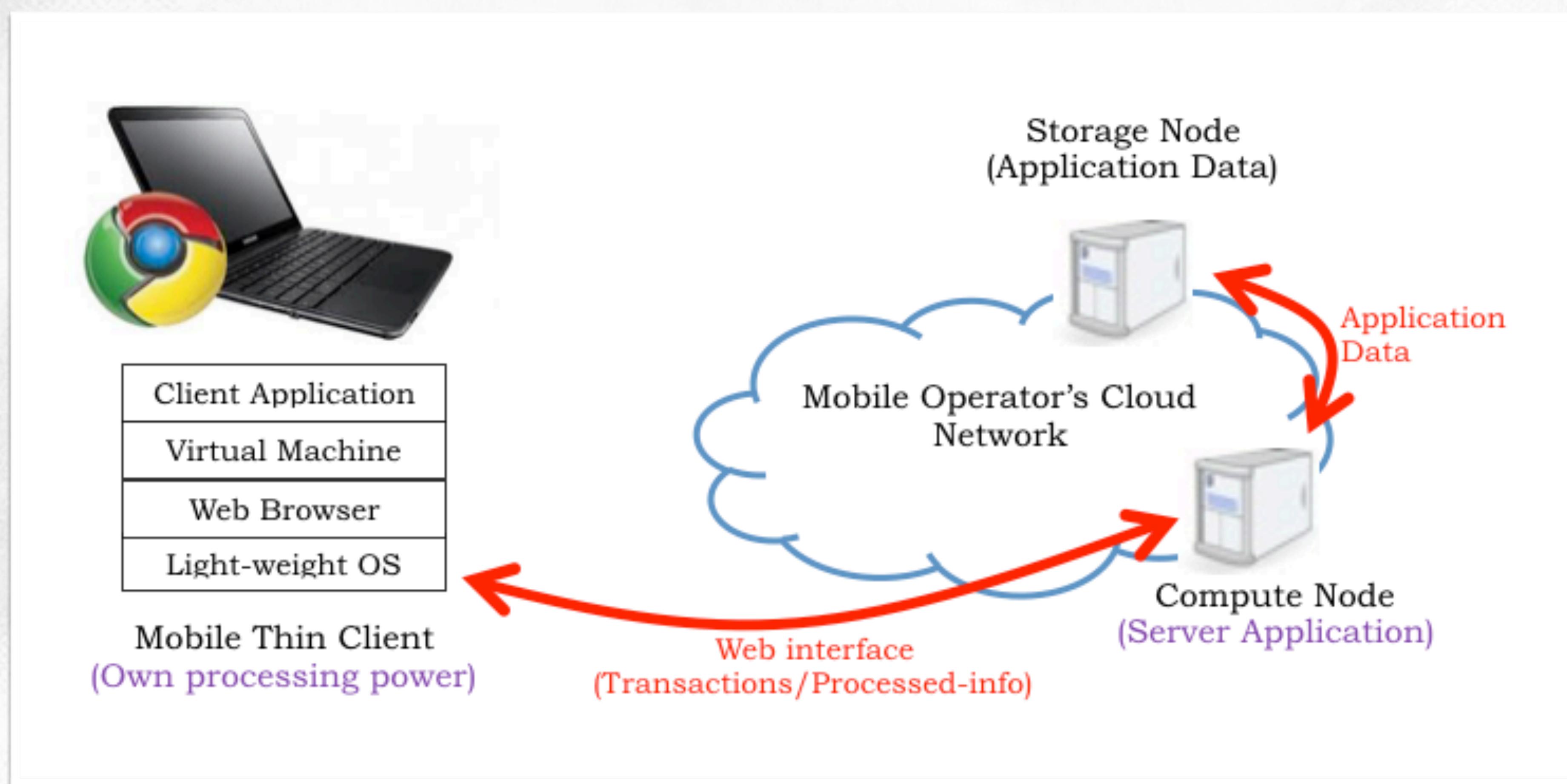
Contents

- Introduction to Cloud Computing
- Public Cloud Services
- Cloud Computing APIs
- **Thin Client**
- Private Cloud Solutions
- Beyond Cloud Computing

Thin Client Concept

Web-Client and Server Architecture

- Mobile node has processing power, and execute client application
- Server application executed on Mobile Cloud
- Conventional web based client/server interface used
- Example: Google Chromium OS, Conventional Smart-Pad/Phone



Thin Client Solutions (Let's Visit)

- Google
 - The Chromium Projects: <https://www.chromium.org/>
 - Chromium on RPi3: https://www.youtube.com/watch?v=IUX_1R21NXs
- Firefox
 - Firefox OS Projects: <https://support.mozilla.org/ko/products/firefox-os>
 - Discontinued
- FlinsOS
 - FLINT OS: <https://flintos.io/>
 - Derived from Google Chromium Projects
 - FLINT OS on RPi3: <https://www.youtube.com/watch?v=amWoimp0qes>

Thin Client meets Cloud Computing Education

The G Suite for Education Core Services

Tools that your entire school can use, together



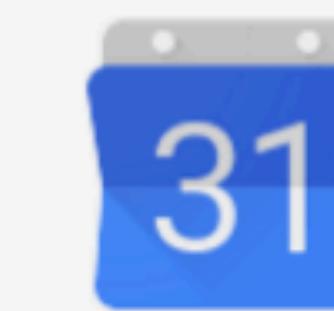
Classroom



Gmail



Drive



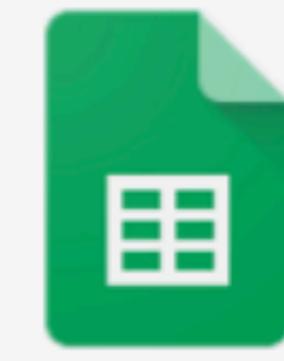
Calendar



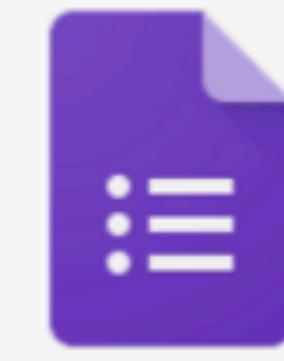
Vault



Docs



Sheets



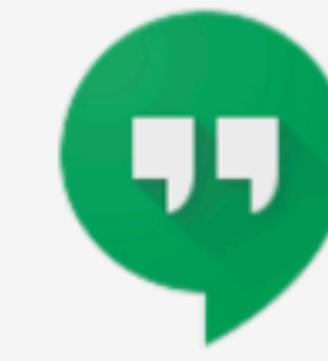
Forms



Slides

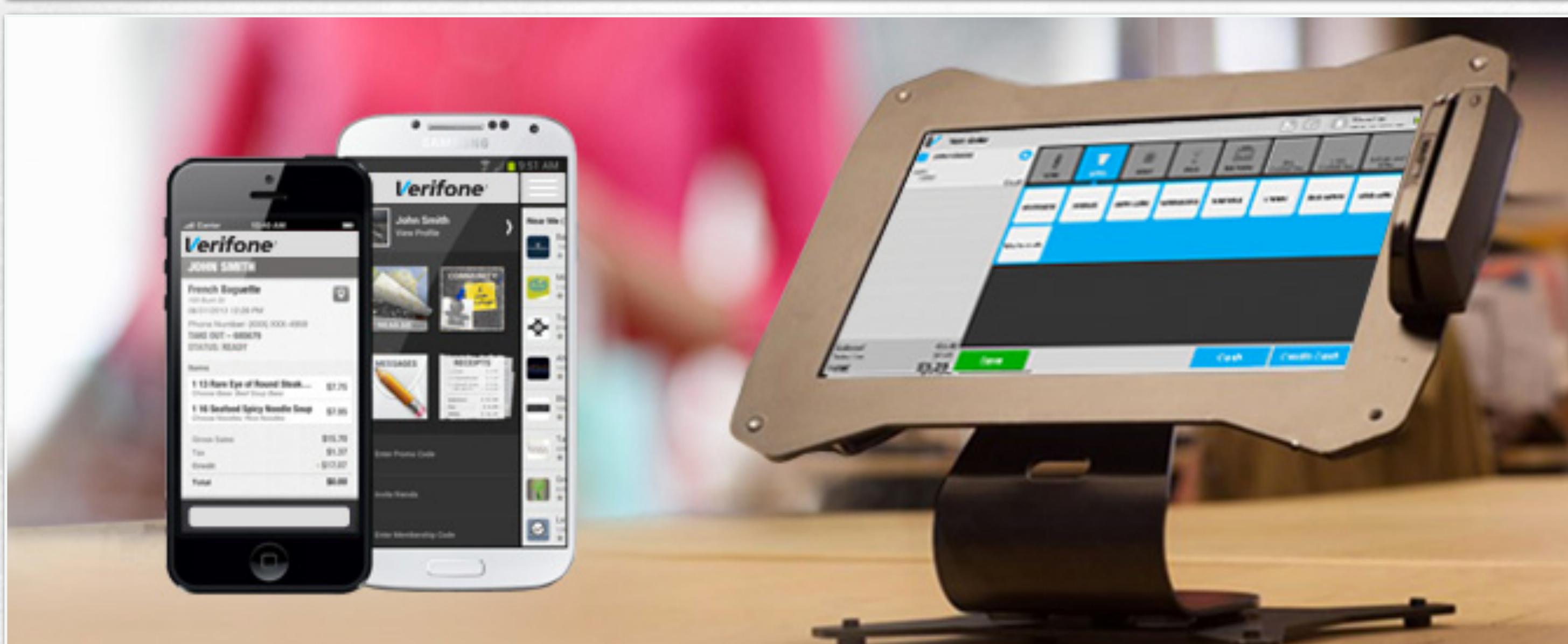
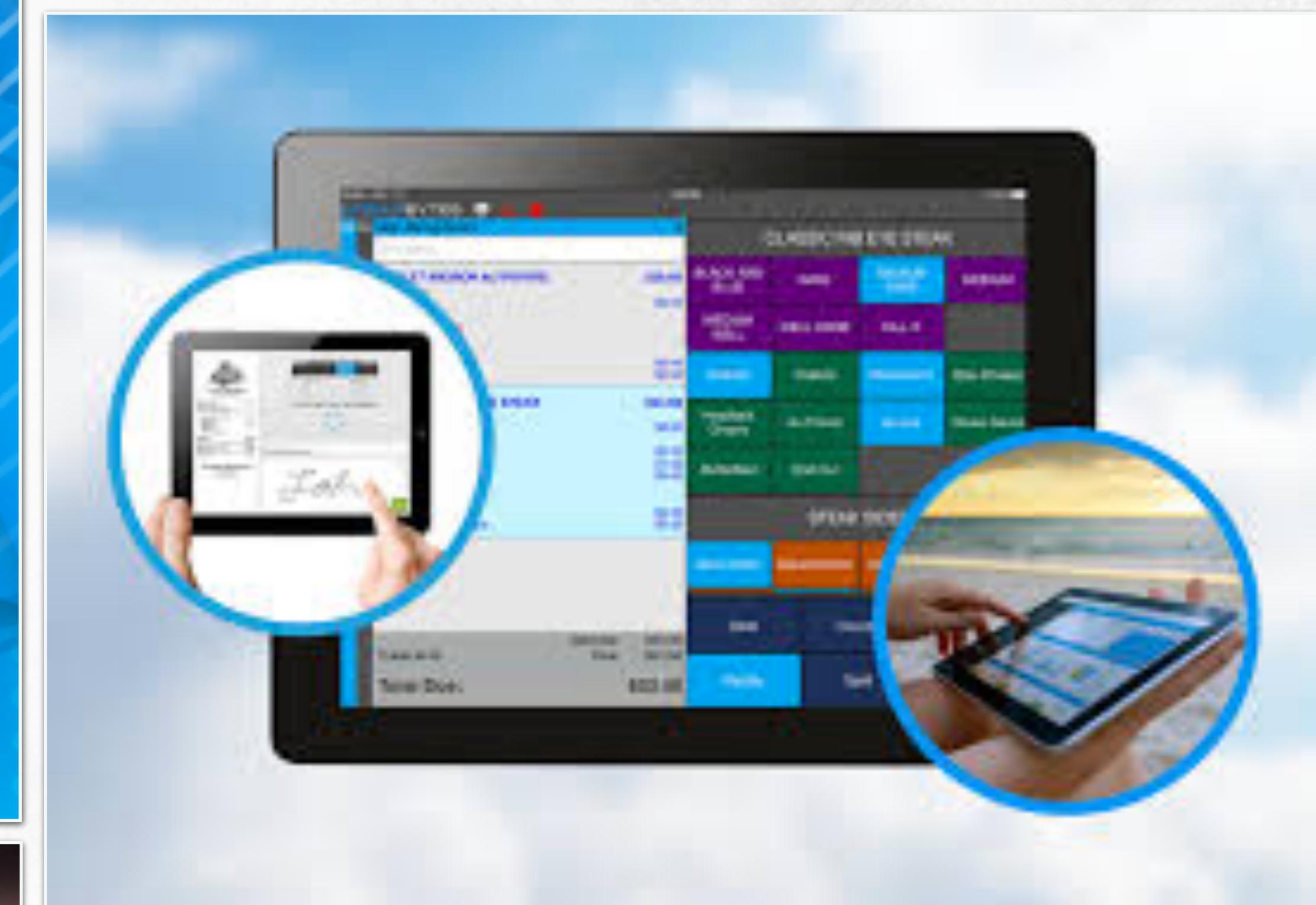


Sites



Hangouts

Thin Client meets Cloud Computing POS System



Reference: <https://www.pinterest.co.kr/pin/436286282628587238/>

<http://www.cervion.com/news/the-right-kind-of-cloud-for-restaurant-pos-success>

<https://www.pymnts.com/company-spotlight/2015/verifone-launches-cloud-based-pos/>

<http://www.restaurantposinfo.com/cloud-tablet-pos-features/seven-benefits-of-ipad-cloud-pos/>

Contents

- Introduction to Cloud Computing
- Public Cloud Services
- Cloud Computing APIs
- Thin Client
- **Private Cloud Solutions**
- Beyond Cloud Computing

REAL Cloud Enabler & Private Cloud Computing Construction

OpenStack:

The screenshot shows the official OpenStack website at <https://www.openstack.org>. The header includes the OpenStack logo, a search bar, and navigation links for Software, Users, Community, Marketplace, Events, Learn, and Docs. A 'JOIN' button and a 'LOG IN' button are also present. The main banner features a blue-toned image of the Sydney Opera House and the text '50 things you missed from the OpenStack Summit Sydney'. Below the banner, a sub-section highlights 'Open source software for creating private and public clouds.' It describes OpenStack's capabilities in managing compute, storage, and networking resources through a dashboard or API, and mentions its use by major brands like Google, IBM, and HP. It also notes its strong ecosystem and support through the Marketplace. Another section shows a terminal window with command-line examples like 'nova boot' and 'swift upload'. A call-to-action button at the bottom right says 'Watch a Demo of the Dashboard'.

Home - OpenStack is open SOL X

안전한 | https://www.openstack.org

Lee

OpenSource GoormEDU WebSites TechTips FutureWorks Entertainment C++17

openstack SEARCH SOFTWARE USERS COMMUNITY MARKETPLACE EVENTS LEARN DOCS JOIN LOG IN

50 things you missed from the OpenStack Summit Sydney

From new users to product launches, updates and demos, here's a news roundup from the Sydney Summit.

Catch the replay >

Open source software for creating private and public clouds.

OpenStack software controls large pools of compute, storage, and networking resources throughout a datacenter, managed through a [dashboard](#) or via the [OpenStack API](#). OpenStack works with [popular enterprise](#) and [open source technologies](#) making it ideal for heterogeneous infrastructure.

[Hundreds of the world's largest brands](#) rely on OpenStack to run their businesses every day, reducing costs and helping them move faster. OpenStack has a strong [ecosystem](#), and users seeking commercial support can choose from different OpenStack-powered products and services in the [Marketplace](#).

The software is built by a [thriving community](#) of developers, in collaboration with users, and is designed in the open at our [Summits](#).

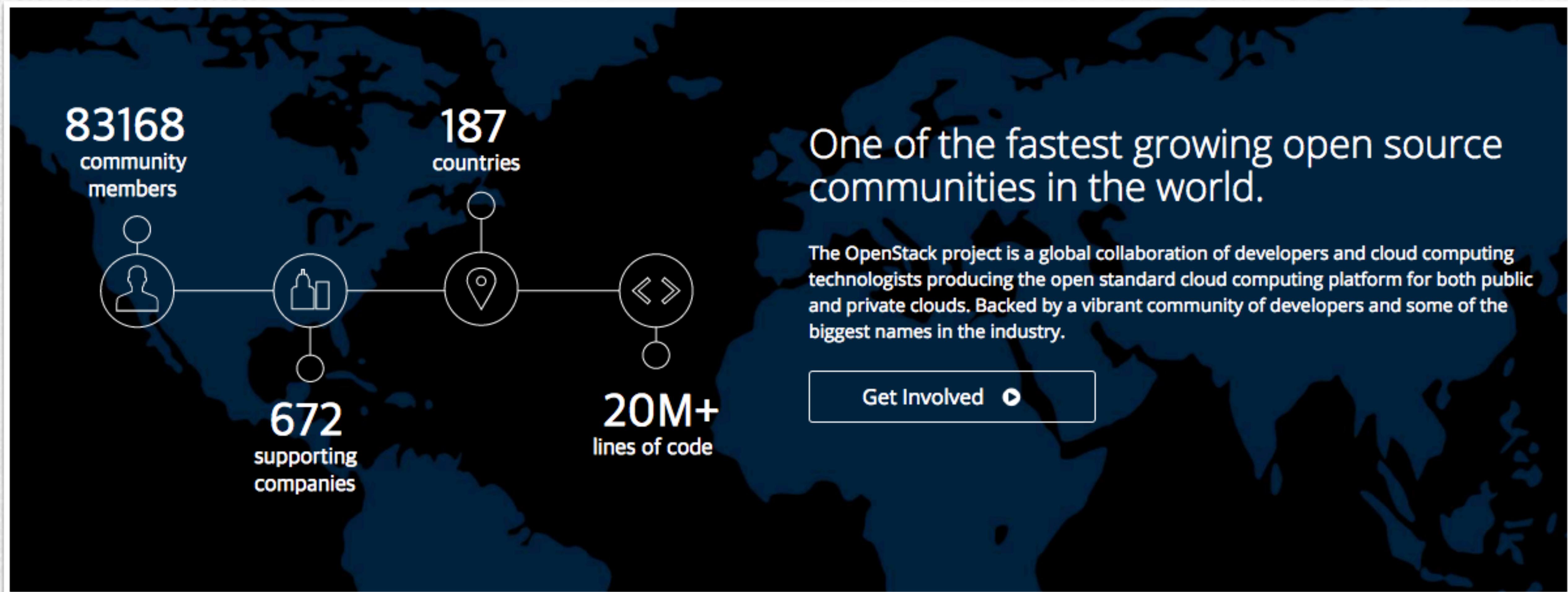
HOW IT WORKS GET THE SOFTWARE

Latest Release: [Pike](#) (August 2017)

Watch a Demo of the Dashboard

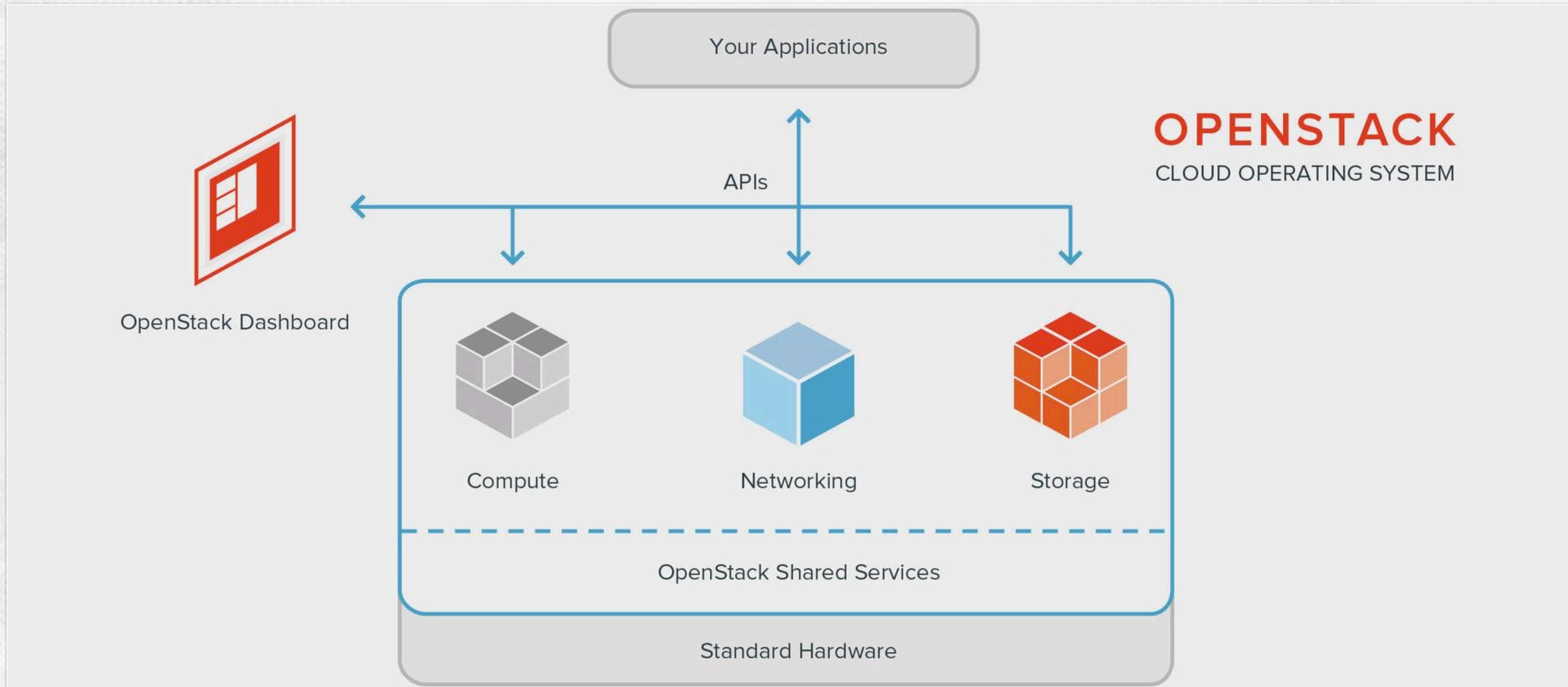
REAL Cloud Enabler & Private Cloud Computing Construction

OpenStack:



REAL Cloud Enabler & Private Cloud Computing Construction

OpenStack Architecture



Contents

- Introduction to Cloud Computing
- Public Cloud Services
- Cloud Computing APIs
- Thin Client
- Private Cloud Solutions
- **Beyond Cloud Computing**

Beyond Cloud Computing

Almost 10 years old technology.

	OpenStack	Ceph	KVM	Open vSwitch
시작 년도	2010년	2007년 이전	2007년 이전	2007년 (공식은 2009년)
최초 개발	NASA, RackSpace	Sage Well (박사학위논문)	Kivity	-
관리 주체	OpenStack Founcaction	RedHat	Open Virtualization Alliance	openvswitch.org
참여 기관	500여사	-	200여사	-
개발 언어	Python	C++, Perl	C	C

Beyond Cloud Computing

Anti Virtual Machine

- Apple's Comments

“When Apple moved to bare metal with Mesos, one of the big reasons why they did it was, first, they did not need the virtual machines and, second, they got a big performance improvement. The virtualization tax that we often talk about is very real and for Apple it was on the order of 30 percent.”

Container Technology

Concept

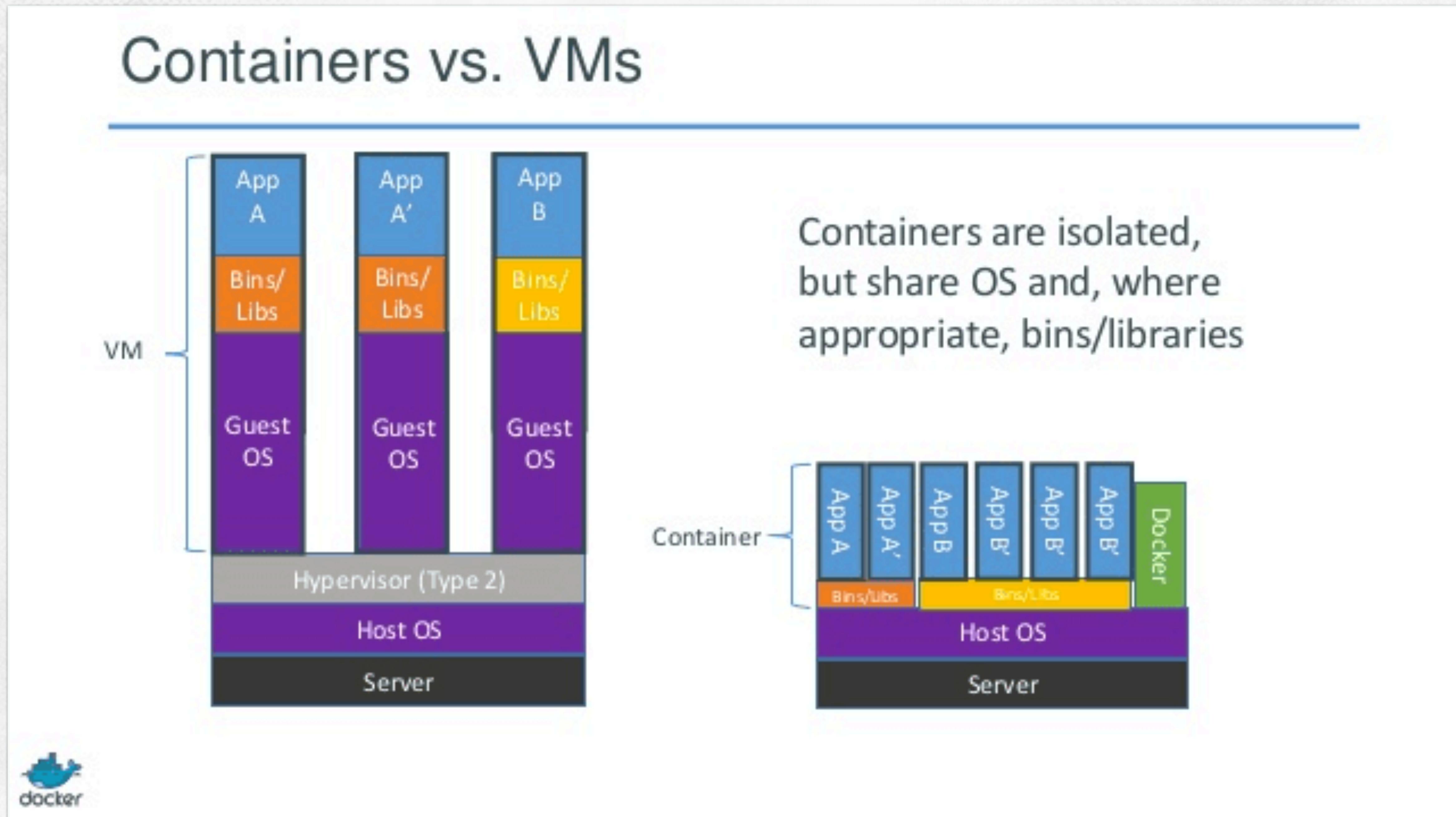
- Container

- share the host's OS and are therefore lighter weight
- really are processes with their full environment
 - A computer science textbook will define a process as having its own address space, program, CPU state, and process table entry
 - But in today's software environment this is no longer the full story
 - The program text is actually memory mapped from the filesystem into the process address space and often consists of dozens of shared libraries in addition to the program itself, thus all these files are really part of the process

Container Technology

Non Virtual Machine Approach

- Containers vs VMs



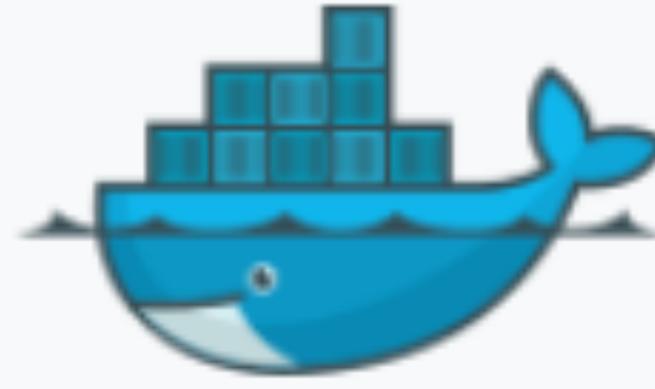
Most Popular Container Software

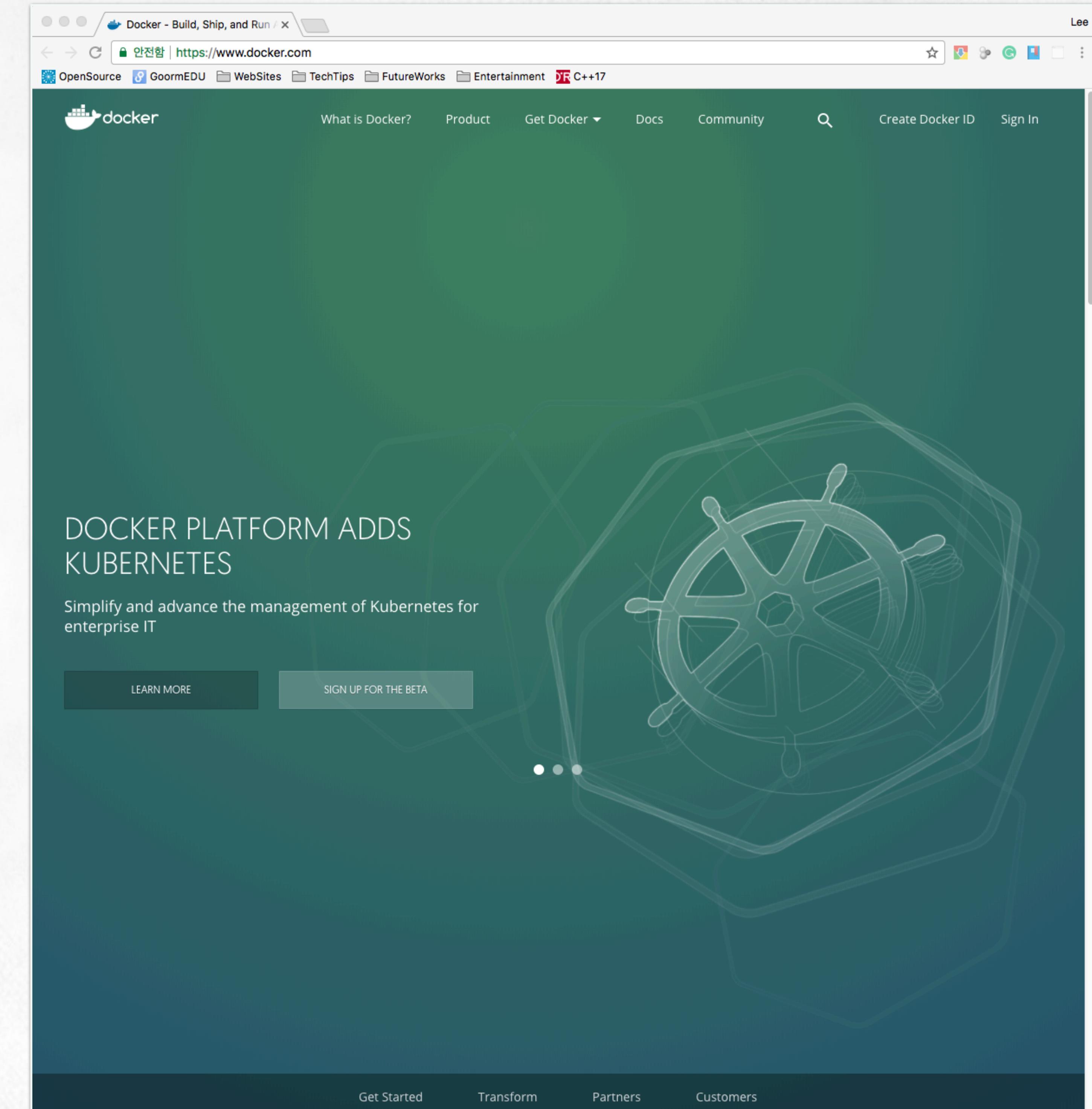
- Docker (<https://www.docker.com/>)

- is a software technology providing containers, promoted by the company Docker, Inc.
- provides an additional layer of abstraction and automation of operating-system-level virtualization on Windows and Linux
- uses the resource isolation features of the Linux kernel such as cgroups and kernel namespaces, and a union-capable file system such as OverlayFS and others to allow independent "containers" to run within a single Linux instance, avoiding the overhead of starting and maintaining virtual machines (VMs)

Container Technology

Container meets Orchestration

	docker
Original author(s)	Solomon Hykes
Developer(s)	Docker, Inc.
Initial release	13 March 2013; 4 years ago
Stable release	17.09.0-ce ^[1] / 26 September 2017; 57 days ago
Repository	github.com/docker/docker-ce ^[2]
Written in	Go ^[2]
Operating system	Linux, ^[a] Windows
Platform	x86-64, ARM (experimental) with modern Linux kernel, or x86-64 Windows with Hyper-V capabilities
Type	Operating-system-level virtualization
License	Apache License 2.0 / Proprietary ^[5]
Website	www.docker.com ^[4]



The screenshot shows the official Docker website at <https://www.docker.com>. The page features a prominent graphic of a ship's steering wheel on the right side. At the top, there is a navigation bar with links for "What is Docker?", "Product", "Get Docker", "Docs", "Community", a search icon, "Create Docker ID", and "Sign In". Below the navigation, there is a section titled "DOCKER PLATFORM ADDS KUBERNETES" with the subtext "Simplify and advance the management of Kubernetes for enterprise IT". Two buttons are visible: "LEARN MORE" and "SIGN UP FOR THE BETA". At the very bottom of the page, there is a dark footer bar with links for "Get Started", "Transform", "Partners", and "Customers".



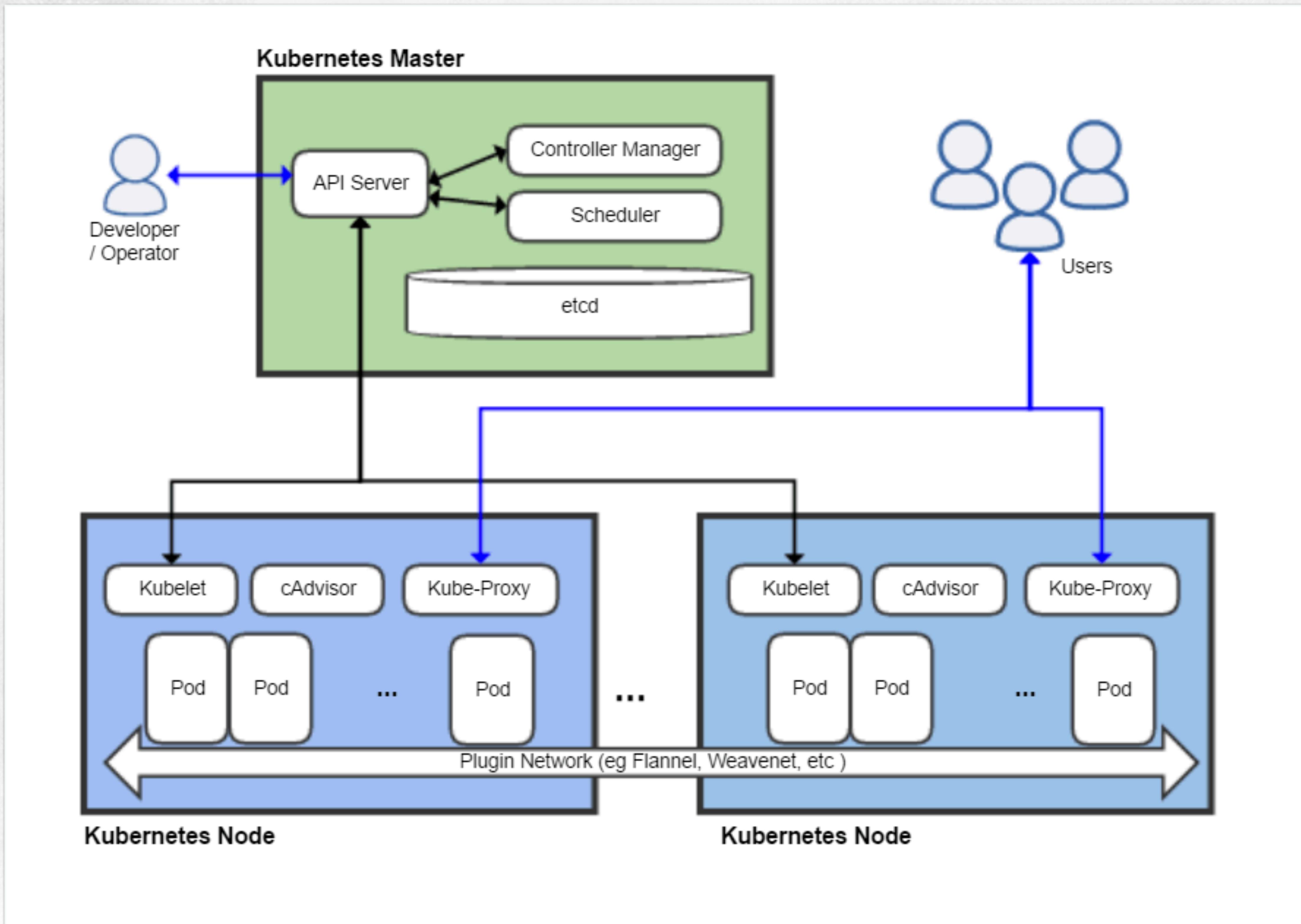
Container Orchestration

- **Kubernetes**

- is an open-source container-orchestration system for automating deployment, scaling and management of containerized applications.
- It was originally designed by Google and is now maintained by the Cloud Native Computing Foundation.
- It aims to provide a "platform for automating deployment, scaling, and operations of application containers across clusters of hosts".
- It works with a range of container tools, including Docker, since its first release.

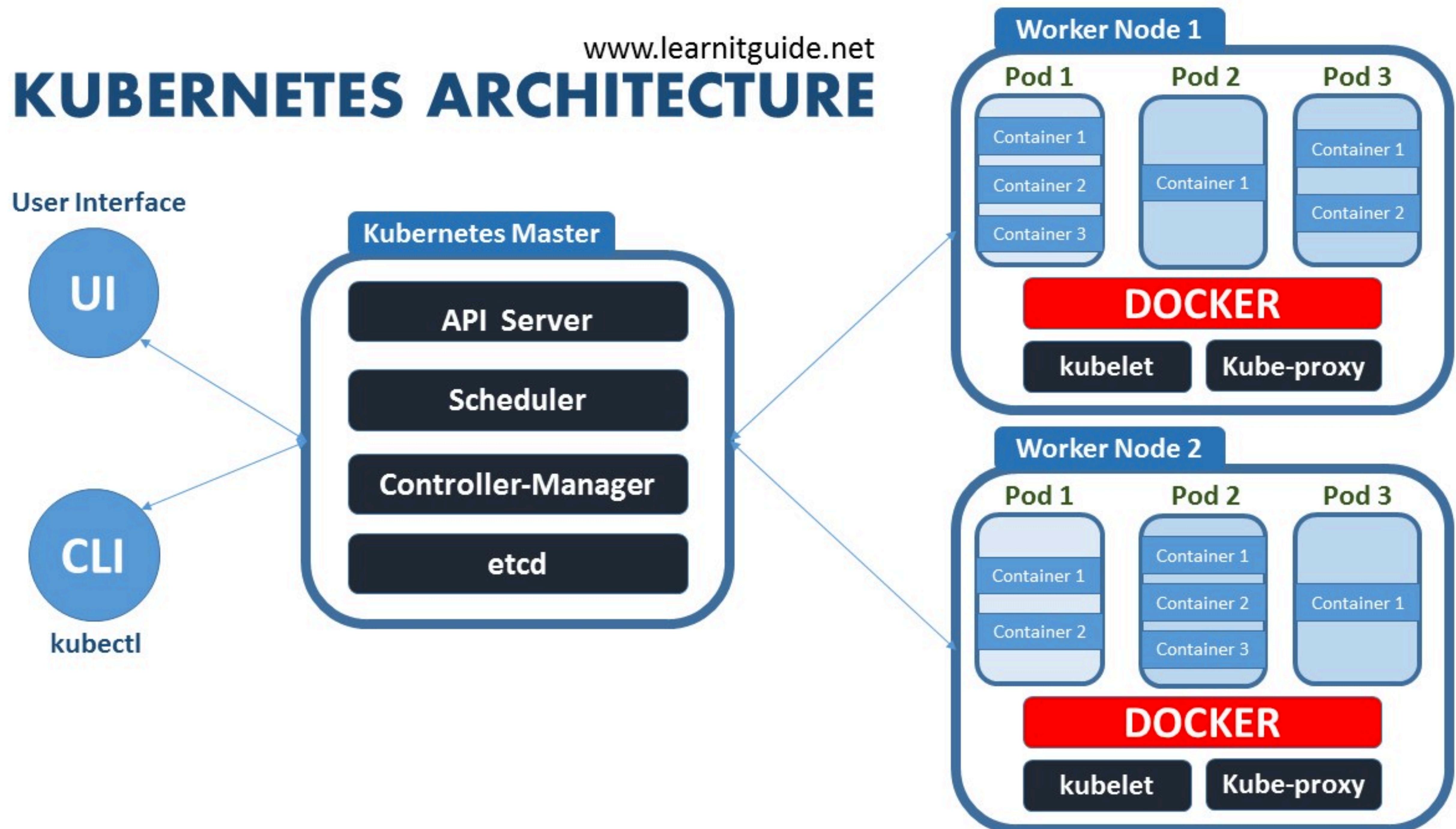
Container Technology

Container Orchestration



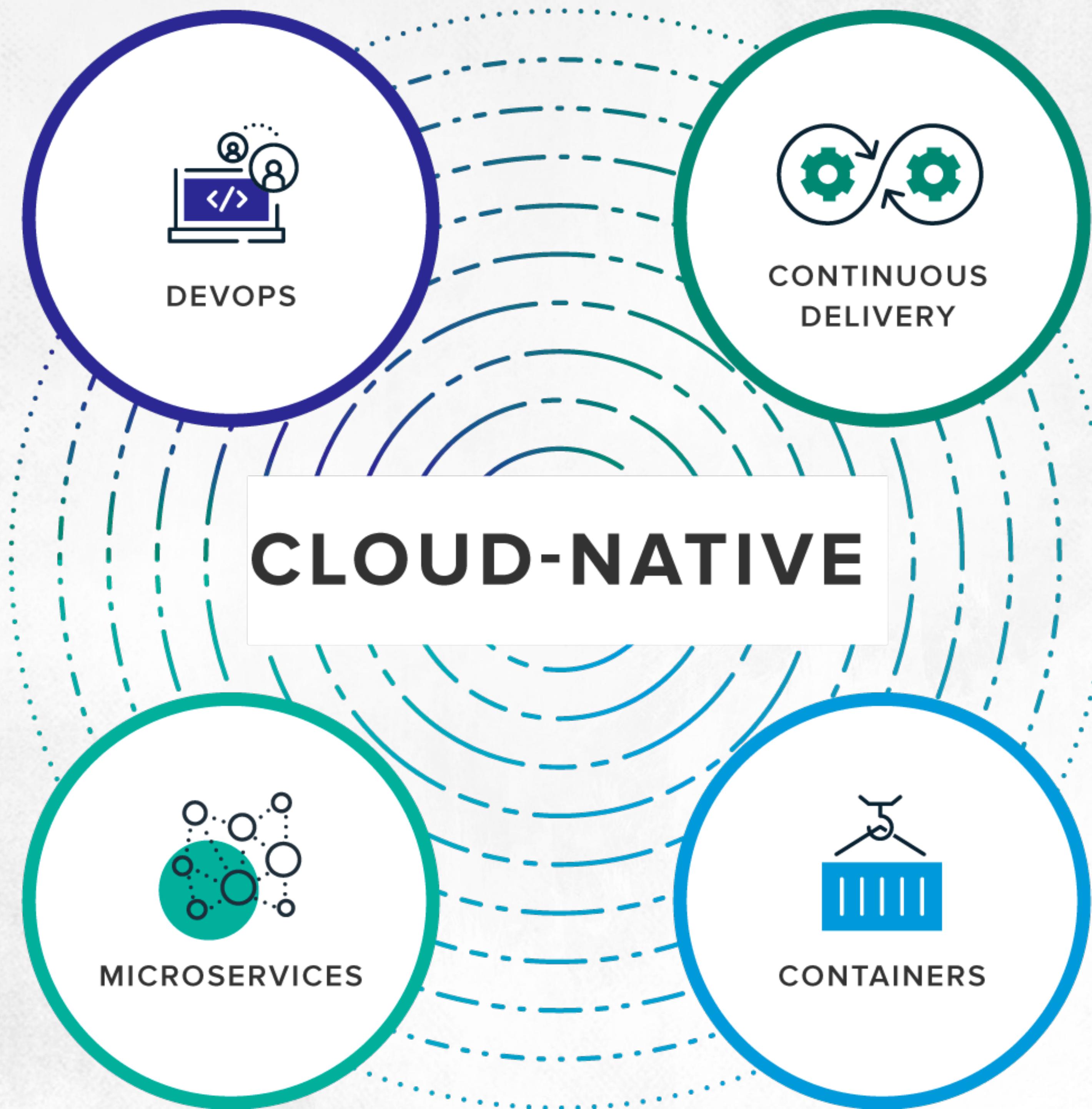
Container Technology

Container Orchestration



Development Culture Innovation

Cloud Native (from pivotal.io)



Development Culture Innovation

Cloud Native (from pivotal.io)

클라우드 네이티브 애플리케이션이란?

공유



클라우드 네이티브는 클라우드 컴퓨팅 모델의 장점을 모두 활용하는 애플리케이션을 개발하고 실행하기 위한 접근 방식입니다. 클라우드는 엔터프라이즈 데이터 센터를 운영할 자본 투자와 직원에 대한 집중도를 제거하는 대신 무한한 주문형 및 종량제 컴퓨팅 능력을 대체해 넓음으로써 실질적으로 모든 업계를 망라해 경쟁 패러다임을 재정의했습니다. IT 비용 감소란 경쟁 우위를 동반한 진입 장벽이 낮아져 팀이 시장에 신규 아이디어를 선보일 수 있는 속도가 기능의 한 부분이 된다는 의미입니다. 그렇기 때문에 소프트웨어가 세상을 집어삼키고 있는 것이고, 스타트업 단계에서 클라우드 네이티브 접근 방식을 사용해 기존의 업계를 파괴하는 것입니다.

그러나 조직에는 DevOps 개념, 지속적인 전달, 마이크로서비스, 컨테이너를 자동화하고 통합할 클라우드 네이티브 애플리케이션과 서비스를 구축하고 운영할 플랫폼이 필요합니다.

DevOps는 소프트웨어 전달과 인프라스트럭처 변경 프로세스 자동화를 목표로 소프트웨어 개발자와 IT 운영자가 협업한 결과물입니다. DevOps는 좀더 신뢰성 있는 소프트웨어 생성, 테스트, 릴리스를 신속하게 자주 진행할 수 있는 문화와 환경을 만듭니다.

Development Culture Innovation

Cloud Native (from pivotal.io)

지속적인 전달 방식으로 개별 애플리케이션을 변경하면 다른 변경 사항과 함께 하나의 릴리스나 유지 관리 창 등의 이벤트에 통합해 들어갈 때까지 기다리지 않고 준비되는 즉시 릴리스할 수 있습니다. 지속적인 전달 방식은 릴리스 작업을 단조롭고 신뢰 가능하게 만들어 주므로 조직은 최종 배포본이 비즈니스 프로세스와 엔터프라이즈 경쟁력의 일부로 통합될 때까지 위험이 낮은 상황에서 자주 전달하고 최종 사용자들로부터 빠르게 피드백을 얻을 수 있습니다.

마이크로서비스는 소규모 서비스 집합의 일부로서 애플리케이션을 개발하려는 아키텍처형 접근 방식입니다. 각 서비스는 비즈니스 성능을 구현하고, 해당 프로세스에서 실행되며, HTTP API를 통해 통신합니다. 각 마이크로서비스는 보통 자동화 시스템의 일부인 애플리케이션에서 다른 서비스를 독립적으로 배포, 업그레이드, 크기 조정, 재시작할 수 있으므로 최종 사용자에 영향을 미치지 않고 운영 중인 애플리케이션을 자주 업데이트할 수 있습니다.

컨테이너는 가상 머신 (VM)과 비교했을 때 효율성과 속도를 모두 제공합니다. 운영 시스템 (OS) 레벨의 가상화를 사용하는 경우 단일 OS 인스턴스는 하나 이상의 격리된 컨테이너에 동적으로 분리되고, 각 컨테이너에는 고유한 쓰기 가능 파일 시스템과 리소스 할당량이 있습니다. 단일 VM에 높은 패킹 밀도로 결합된 컨테이너들을 제작하고 파괴하는 등의 낮은 오버헤드는 컨테이너를 완벽한 개별 마이크로서비스 배포 수단으로 만들어 줍니다.

Development Culture Innovation

Cloud Native Foundation

The screenshot shows the homepage of the Cloud Native Computing Foundation. At the top, there's a banner with a quote: "At this point, literally every major U.S. based technology vendor / cloud provider is a CNCF member and have an active Kubernetes strategy. Kubernetes has to be the de facto standard for container orchestration." -ARCHITECHT. Below the banner, there's a section titled "Currently Hosted Projects" featuring logos for various projects: kubernetes, Prometheus, OPENTRACING, fluentd, linkerd, gRPC, CoreDNS, containerD, rkt, CNI, envoy, JAEGER, Notary, and Software Update Spec. Each project has a small description below its logo.

The screenshot shows the members page of the Cloud Native Computing Foundation. It displays logos of members categorized by membership level: Platinum Members (aws, CISCO, Core OS, DELL Technologies, docker, FUJITSU, Google Cloud, HUAWEI), Gold Members (IBM, intel, Joyent, MESOSPHERE, Microsoft, ORACLE, Pivotal, redhat, SAMSUNG SDS, SAP, SUPERNAP, vmware), and Silver Members (Alibaba Cloud, AT&T, NetApp, SUSE, ZTE, Tencent Cloud).

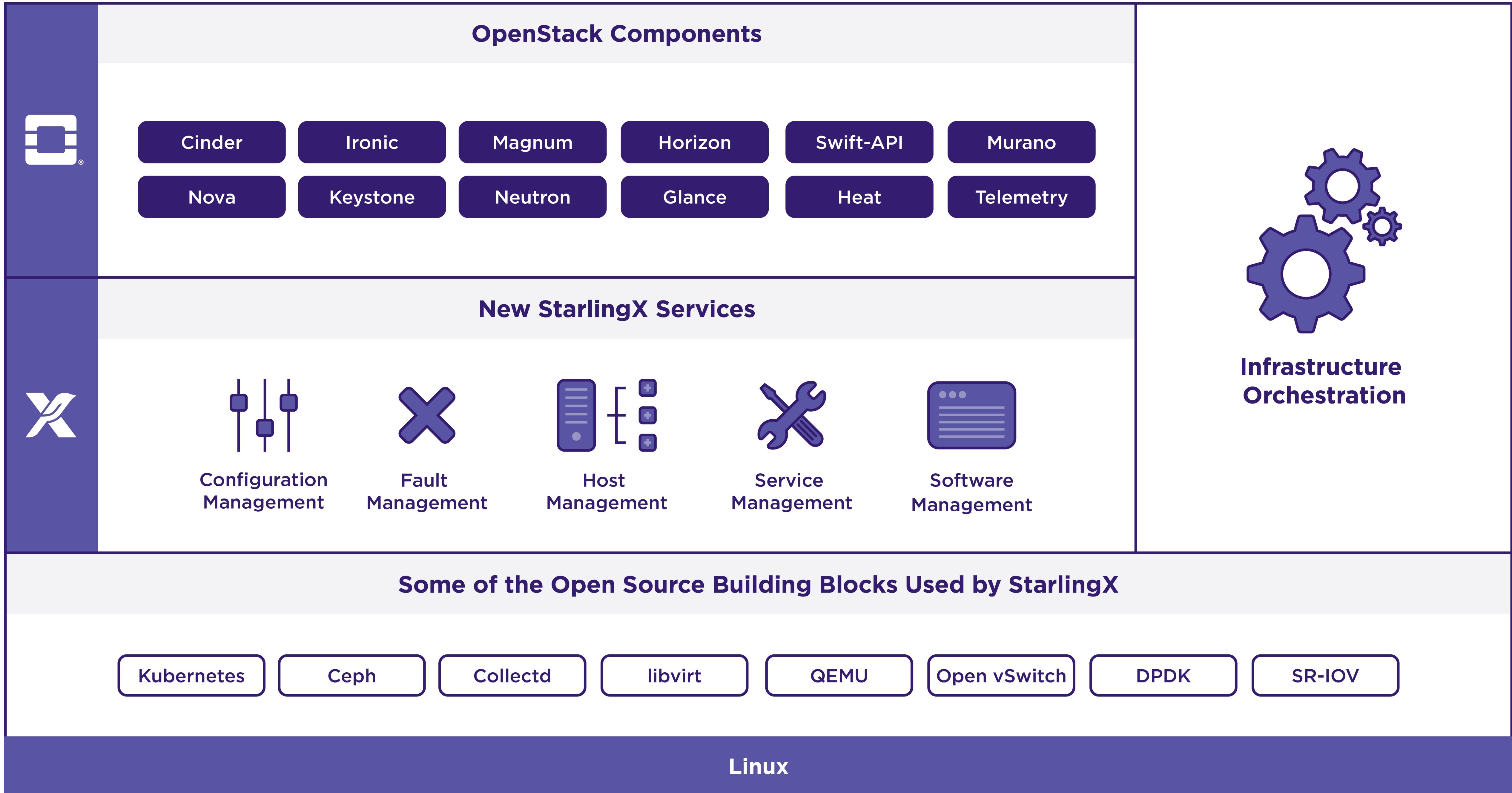
Edge Computing

StarlingX



- A fully featured cloud for the distributed edge
- StarlingX is a complete cloud infrastructure software stack for the edge used by the most demanding applications in industrial IOT, telecom, video delivery and other ultra-low latency use cases. Based on mature software deployed for mission critical applications, newly open sourced StarlingX code is the base for edge implementations in scalable solutions that is ready for production now.
- StarlingX—the open source edge computing and IoT cloud platform optimized for low latency and high performance applications—is now available in its first release. The project was established in May as a pilot project supported by the OpenStack Foundation (OSF) and builds on code contributed by Wind River and Intel.

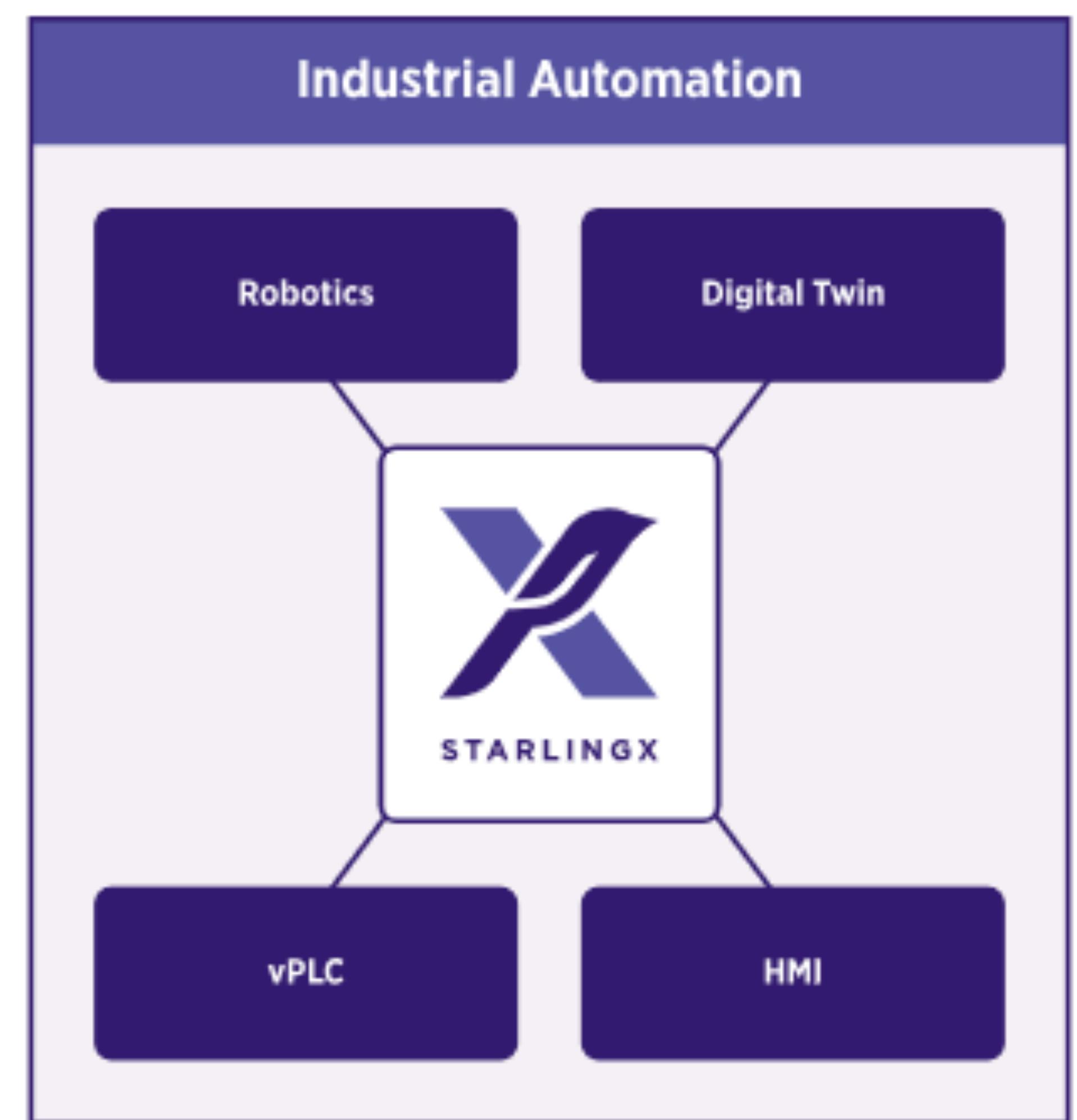
Edge Computing StarlingX



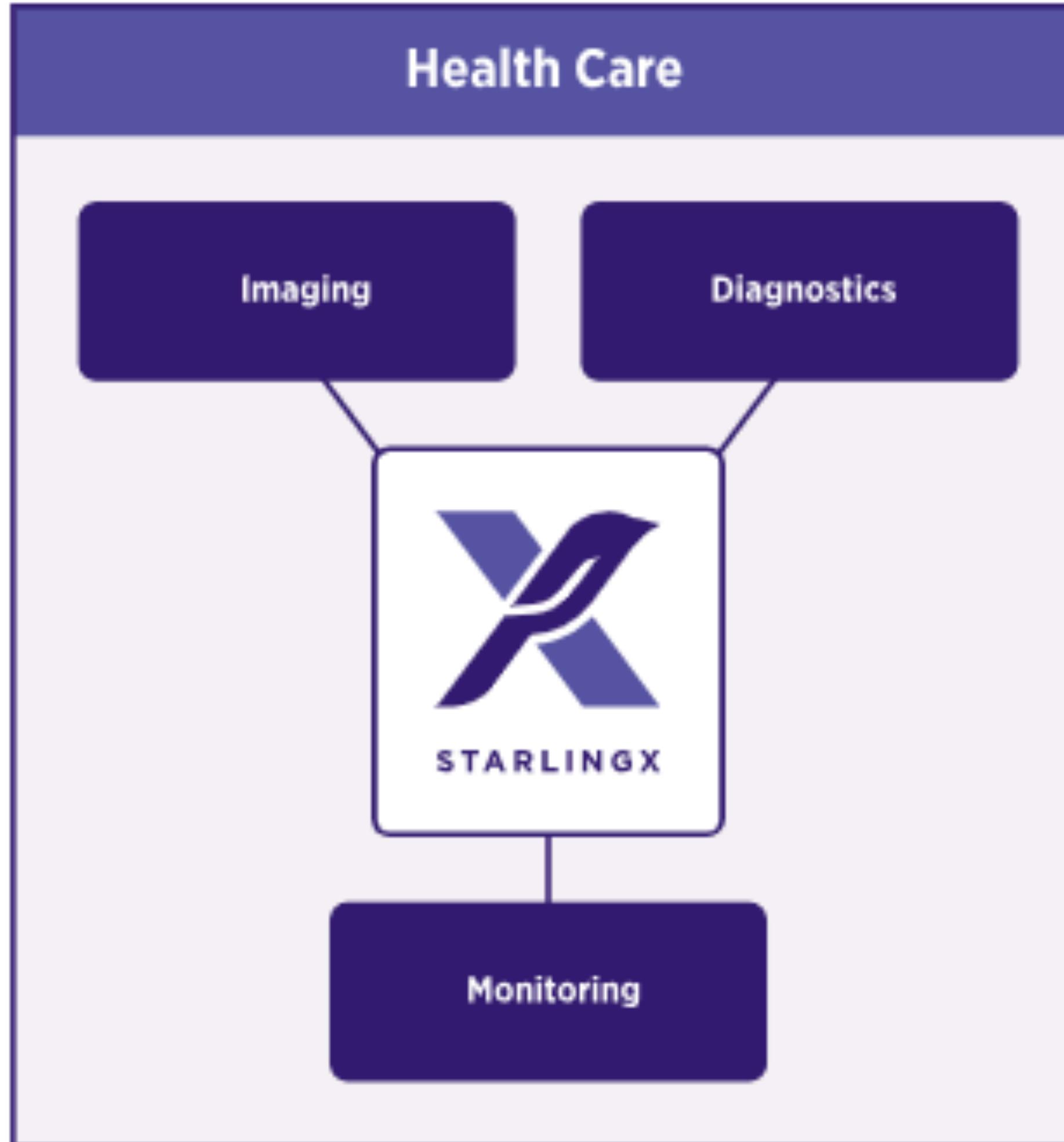
Edge Computing StarlingX Use Cases

Ultra Low-latency 5G and Industrial IoT (IIoT)

- Autonomous vehicles (drones, cars and trucks)
- Industrial automation (robotics and virtual Programmable Logic Controller (vPLC))
- Cloud/virtual Radio Access Network (cRAN/vRAN)
- Smart city/buildings (metering and monitoring)



Edge Computing StarlingX Use Cases



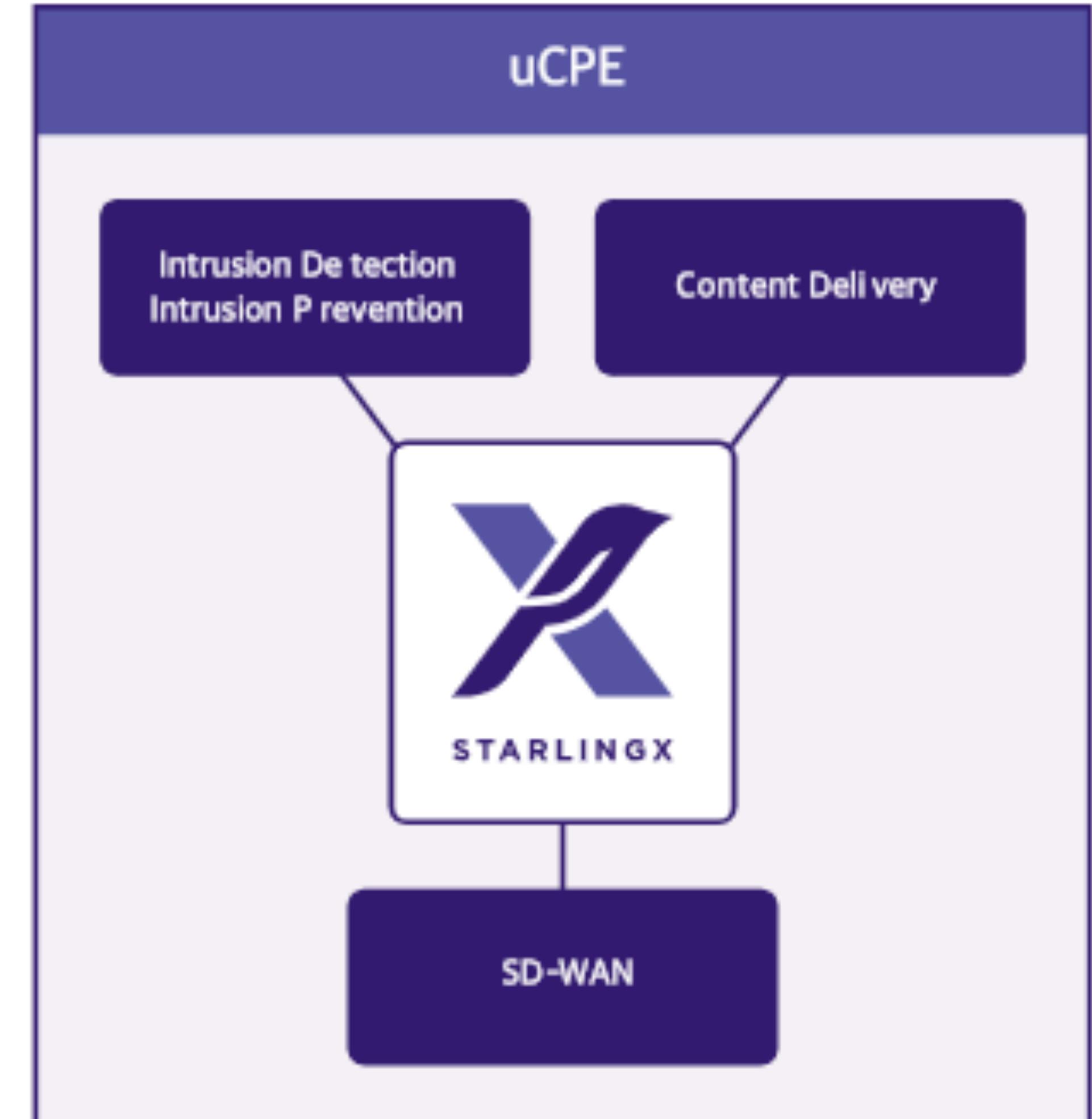
High Bandwidth, Large Volume Applications

- Mobile HD video
- Content delivery
- Healthcare (imaging and diagnostics)
- Caching and surveillance

Edge Computing StarlingX Use Cases

Multi-access Edge Computing (MEC)

- Augmented and virtual reality (AR/VR)
- Enterprise focused small cell services for stadiums and high-density locations
- Unified Customer Premise Equipment (uCPE) applications
- Retail



스크린샷

Edge Computing StarlingX



Reliability

Fault management, fast secure VM failover and live migration minimizes downtime



Scalability

Deployable on one to thousands of distributed nodes allowing for a single system to be used from edge to core



Small footprint

Providing a platform for edge and IoT use cases even for environments with tight resource constraints



Ultra-low latency

Deterministic, tunable performance optimized for the use case



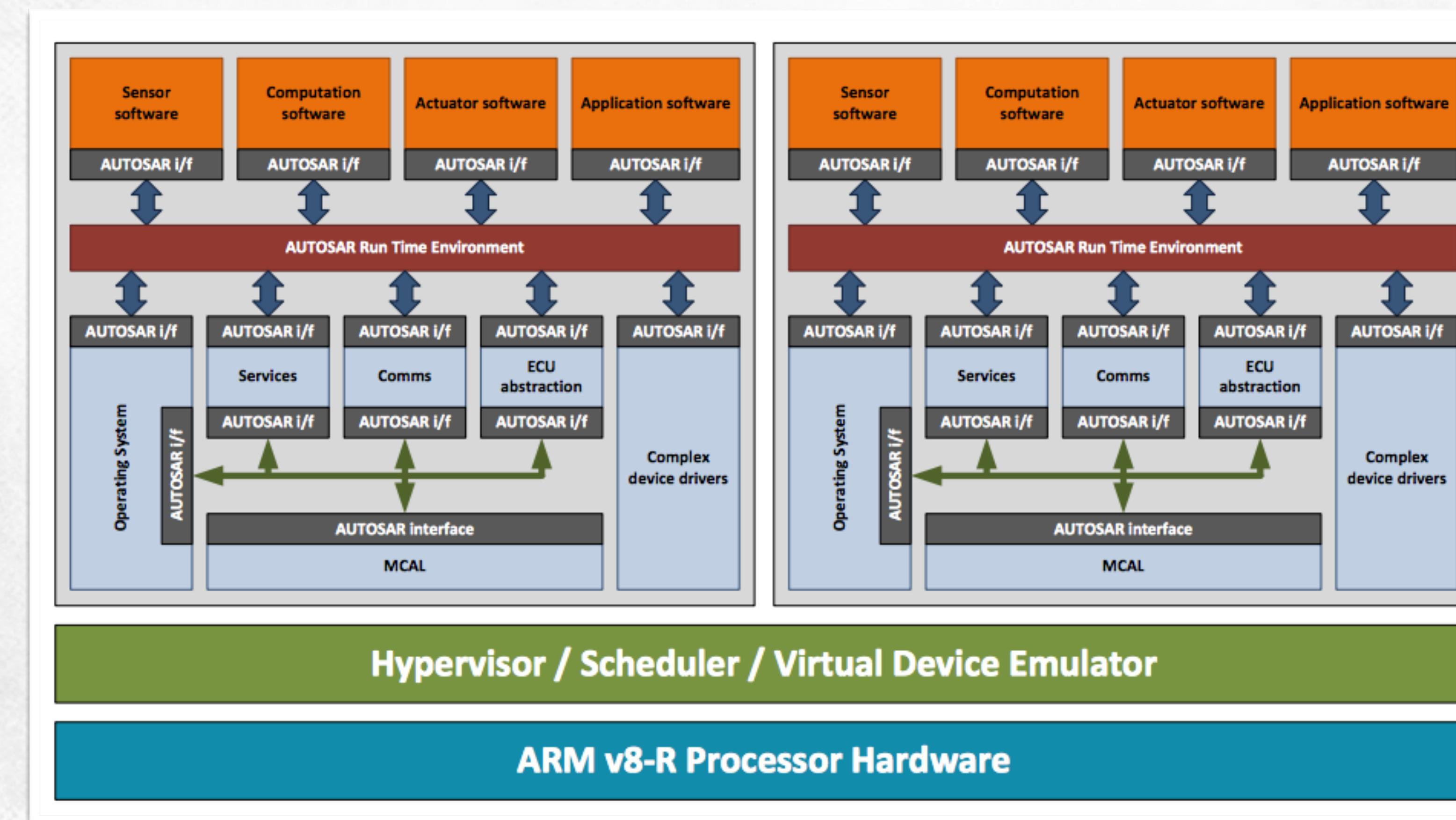
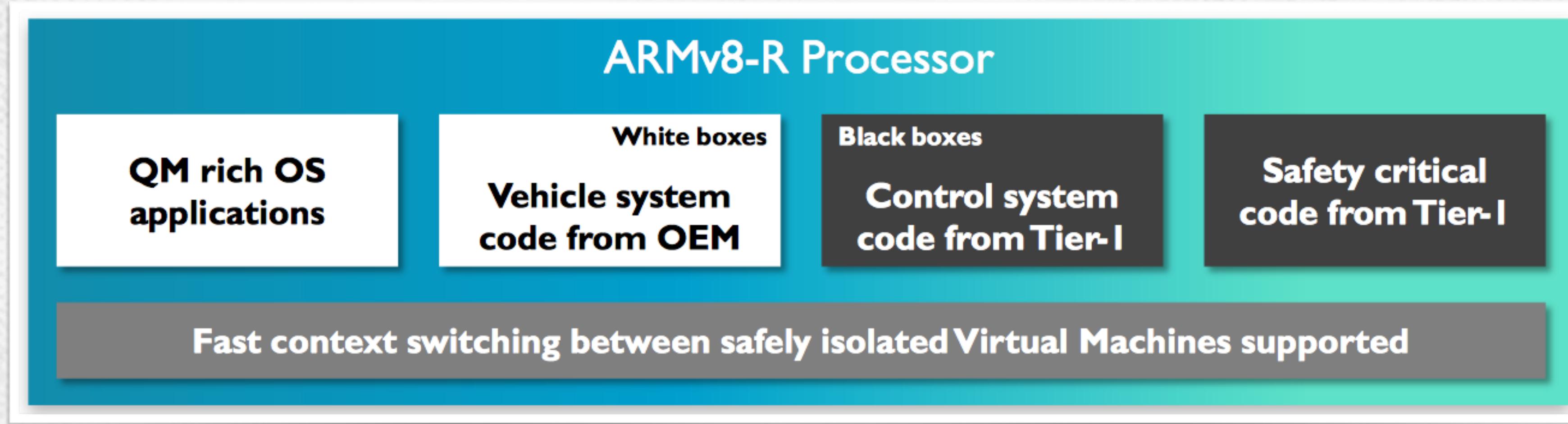
Edge security

Software security to avoid tampering at the edge, where physical security may be limited

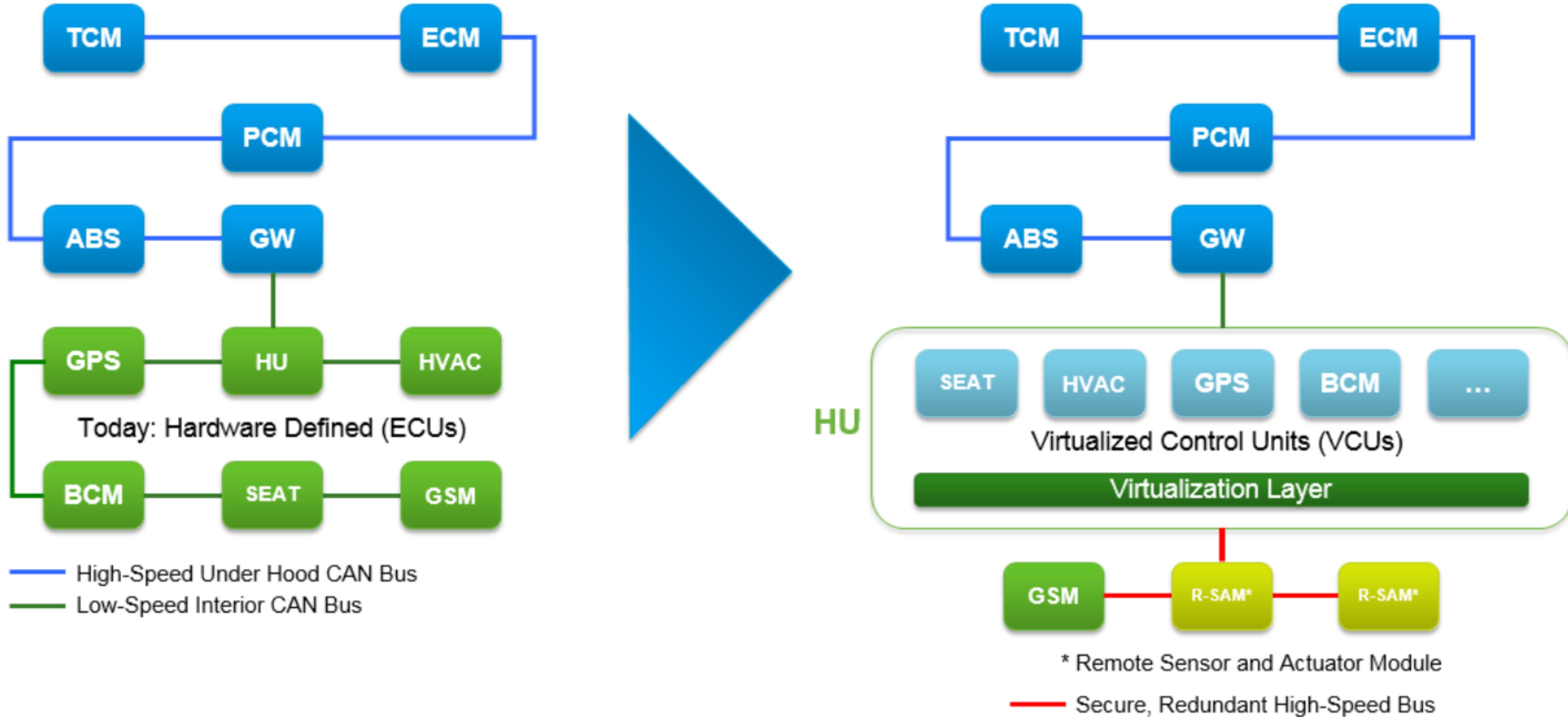


Lifecycle management

Simplified deployment and operations with full system management through comprehensive orchestration suited for the edge



Vision – The Software-Defined Car





Thank you