# Virtualization and Containerisation

# Virtualisation

Virtualization is the division of *physical* computing resources. It's what makes cloud computing possible.

Through software called a *hypervisor*, virtualization slices the physical resources of a server (such as the RAM, CPU, and storage) and turns them into virtual resources. It's the technology that we are using right now to split the resources of your laptops to run the Virtual Box machines for this course.

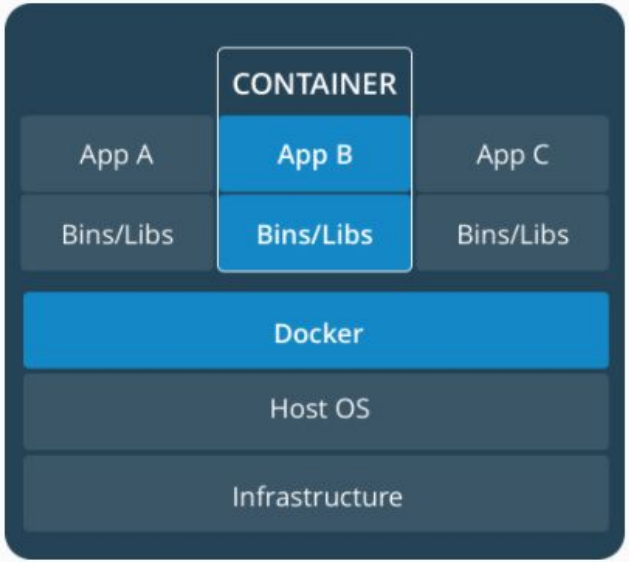
A physical server that's running a hypervisor is referred to as a virtual host.

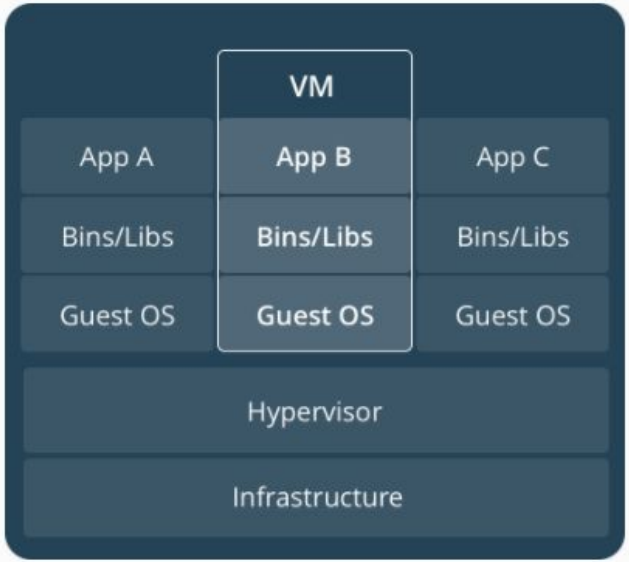
**Discussion:**

# Containerisation

Any Ideas... how does containerization relate to virtualization?

# Containerisation




### CONTAINERS

Containers are an abstraction at the app layer that packages code and dependencies together. Multiple containers can run on the same machine and share the OS kernel with other containers, each running as isolated processes in user space. Containers take up less space than VMs (container images are typically tens of MBs in size), and start almost instantly.

### VIRTUAL MACHINES

Virtual machines (VMs) are an abstraction of physical hardware turning one server into many servers. The hypervisor allows multiple VMs to run on a single machine. Each VM includes a full copy of an operating system, one or more apps, necessary binaries and libraries - taking up tens of GBs. VMs can also be slow to boot.

***Knowledge check:*** What is the biggest difference between virtualization and containerization?

# The Major Benefits

**Group Exercise:**

# What are the Major Benefits

**5 mins**

Have a quick discussion and come back with a (preferably ranked) list of what you feel are the biggest benefits of the cloud. What benefits does it provide for companies?

# Elasticity

The cloud provides elasticity, allowing companies to scale resources up and down quickly.

Think of buying a car versus renting one. If you buy a car, you're saddled with the expense until you pay it off. If you rent a car, you commit to paying for it as long as you want to use it; once you've paid for that use, you have no further obligation to pay! You only pay for what you use, just like with your water service or electrical utility.

# Scalability

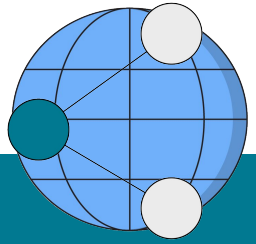
It is scalable. Cloud infrastructure scales much more quickly than onsite infrastructure.

# It’s a Service

This is especially beneficial for smaller companies that may not have the required technical support onsite. Servers can be provisioned in a matter of a few clicks and disposed of just as easily. Cloud providers boast better uptimes than most businesses.

# We could go on....

### Reliability



Cloud services are built from the ground up to be fault tolerant and are spread across multiple physical locations.

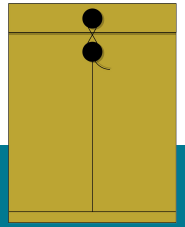
### Speed

Computing resources can be provisioned in minutes with just a few clicks.

### Productivity

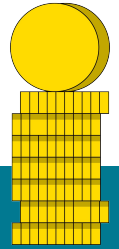
Teams are able to bypass hardware setup, software patching and other time-consuming IT chores to focus on delivering value and solving business problems.

### Security



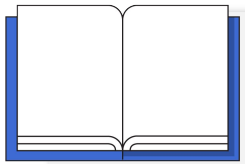
Automatically leverage the expertise and resources of large tech organizations in regards to securing infrastructure.

### Cost Savings



Cloud computing eliminates the capital expense of buying hardware / software and maintaining it.

# Case Study: Capital One



## On-Demand Computing Helps Capital One Teams Move Faster Than Ever

In 2015, Capital One announced that all new company applications would run in—and all existing applications would be rearchitected—for the cloud.

Now by using AWS, they've cut the time needed to build new application infrastructure by more than 99 percent.

**Source:** AWS

# Providers

**Discussion:**

# Main Providers

Who are the main contenders in the world of cloud computing?

**Group Exercise:**

# Major Players

10mins




There are many offerings in the Platform as a Service landscape.

Each group will chose **one provider**. Your task is to do some research and provide some details of where your platform shines. Here are some prompts but feel free to build on these

- What is this platform great at?
- Where is it lacking?
- What are the best use cases for this platform?

# Serverless Computing

AWS' Lambda service leads the way in serverless computing due to supporting every programming language with its Runtime API, integrating with dozens of AWS services natively, or with other monitoring, visibility, and security tools via AWS Lambda Extensions. Presently, AWS is the only provider in our cloud services comparison to offer Provisioned Concurrency in order to keep functions initialized and hyper-ready to respond to an event.

# Serverless Computing

Storage optimized vms are ideal for big data so lets look at these as an example of how providers differ:

AWS and Azure are fairly similar in their ranges of Storage Optimized VMs—respectively offering 7.625GiB and 8GiB of memory per vCPU. Google doesn't offer a designated Storage Optimized VM option, but instead allows you to add one of the following to an existing VM:

.

- Zonal Standard (HDD) Persistent Disks
  - slow but cheap, better suited to data processing workloads that primarily use sequential I/Os
- Regional Standard Persistent Disks
  - same as above but with synchronous replication across two zones in a region
- Zonal Balanced Persistent Disks
  - suitable for most general purpose applications at a price point between that of standard and SSD persistent disks.
- Regional Balanced Persistent Disks
  - above but with synchronous replication across two zones in a region
- Zonal SSD Persistent Disks
  - faster and more suitable for enterprise applications and high-performance database workloads
- Regional SSD Persistent Disks
  - suitable for workloads that may not have application-level replication
- Local SSD Disks
  - have much higher throughput and lower latency

# Containers

As with the VM modeling, different cloud providers will have different offerings when it comes to containerisation.

While AWS will provide Amazon Elastic Kubernetes Service, Amazon Elastic Container Service and Fargate, Azure provide Azure Kubernetes Service. Azure Container Service was retired as a standalone service in January 2020.

# Pricing

All providers have slightly different pricing models

# Developer Experience

A large proportion of developers will lean towards a certain provider based on the user/developer experience. The ease and speed of setting up services through a well developed UI is important.

# Remember When....

We asked you to define and describe several aspects of the cloud? Well, now you should be able to answer all of these questions:

- What are the drivers?
- What are the blockers?
- What is the cloud?
- How does it work?

# Infrastructure

# Section Objectives

- Explain the benefits of adopting a cloud-based infrastructure
- Differentiate between the main service models (SaaS, IaaS, PaaS).
- Explain the principles of 12 factor app design

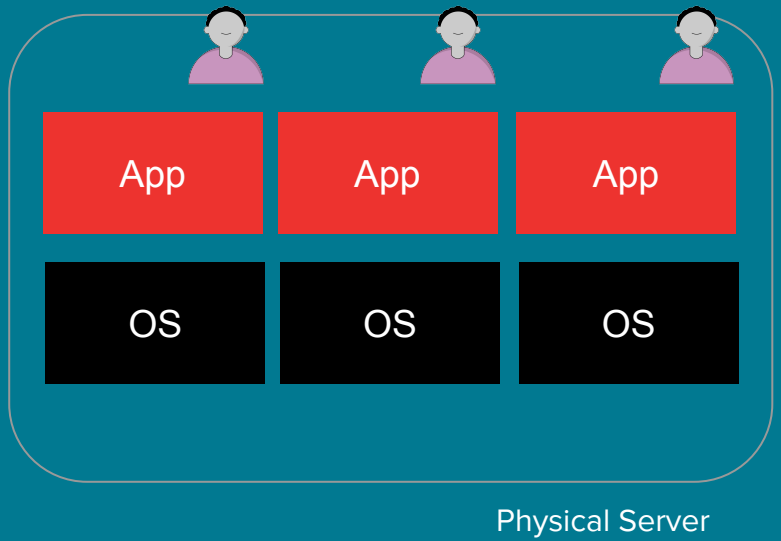
# Essential Characteristics

- On-demand self-service provisioning of resources
- Broad network access
- Resource pooling
- Rapid elasticity
- Measured service

# Shared Resources

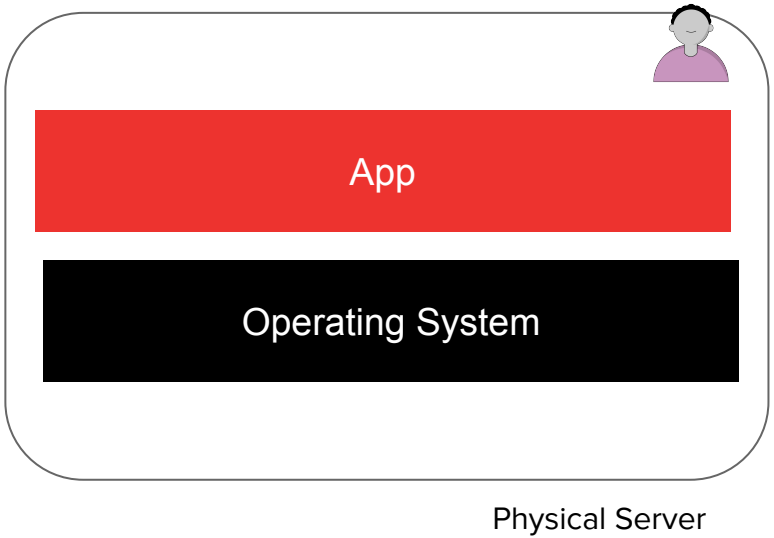
Cloud computing is enabled by virtualization and shared infrastructure

## Virtualization




# Owned Resources

## Traditional Architecture



# Let's Take a look

# Access From Anywhere

One of our 5 characteristics of cloud computing is broad network access. Broad network access refers to resources hosted in a private cloud network (operated within a company's firewall) that are available for access from a wide range of devices, such as tablets, PCs, Macs and smartphones. These resources are also accessible from a wide range of locations that offer online access.

# Access From Anywhere

A huge benefit of cloud computing is the availability of services. For example, businesses may choose to leverage services from a provider such as

- Authentication
- User management
- Monitoring
- Analytics

The benefits of using such services include leveraging the expertise of the developers who created the software. For small development teams this can be hugely beneficial - rather than spending resources creating an authentication application, developers can use authentication services provided by their cloud provider.

# Rapid Elasticity

Rapid elasticity allows users to automatically request additional space in the cloud or other types of services. The capabilities of the cloud should appear unlimited to the user. These capabilities should be elastically scalable both horizontally and vertically in accordance with demand, whatever the quantity of resources required, and at any time.

# Measured Service

The NIST talks about measured service as a setup where cloud systems may control a user or tenant's use of resources by leveraging a metering capability somewhere in the system. The general idea is that in automated remote services, these measurement tools will provide both the customer and the provider with an account of what has been used. In more traditional systems, items like invoices and service change agreements would fill these same roles.

# Cloud Service Models

# Cloud Service Models

Cloud computing is generally offered in three different service models which each satisfy a unique set of business requirements. The primary difference between these models is the amount of control you have over the infrastructure.

These three models are known as:

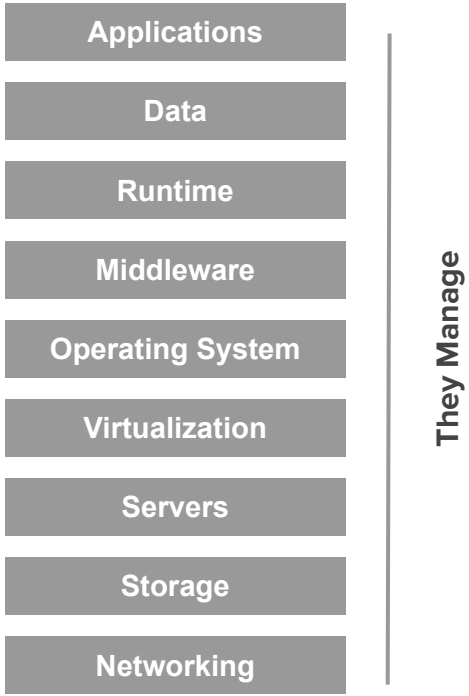
- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

# Software as a Service (SaaS)

SaaS is a cloud-based service where instead of downloading software onto your computer to run, you instead access an application via an internet browser.

Examples include:

- Gmail
- Dropbox
- Salesforce
- Confluence



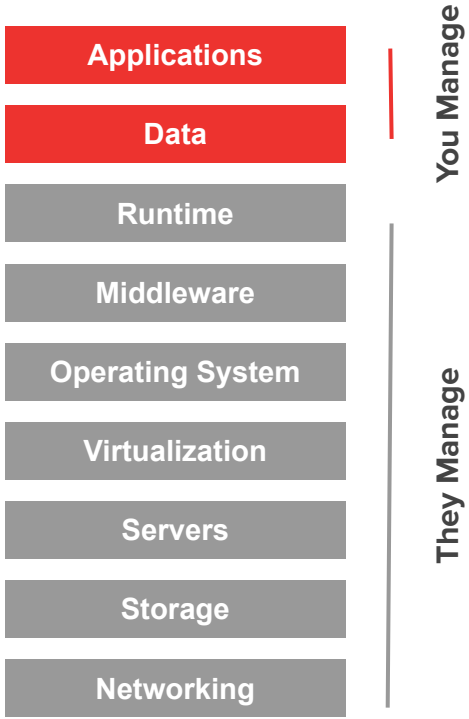

 

# Platform as a Service (PaaS)

PaaS allows developers the freedom to concentrate on building software without having to worry about operating systems, software updates, storage, or infrastructure.

Examples include:

- AWS Elastic Beanstalk
- Google App Engine
- Heroku



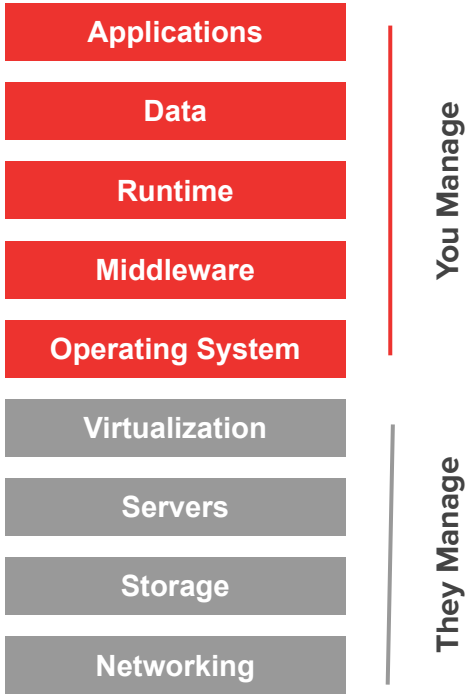

 

# Infrastructure as a Service (IaaS)

IaaS provides the same technologies and capabilities as a traditional data center without having to physically maintain or manage all of it.

Examples include:

- Amazon Web Services (AWS)
- Microsoft Azure
- Google Compute Engine



# Pizza as a Service?