Microsoft Ignite 2019: Azure Fundamentals

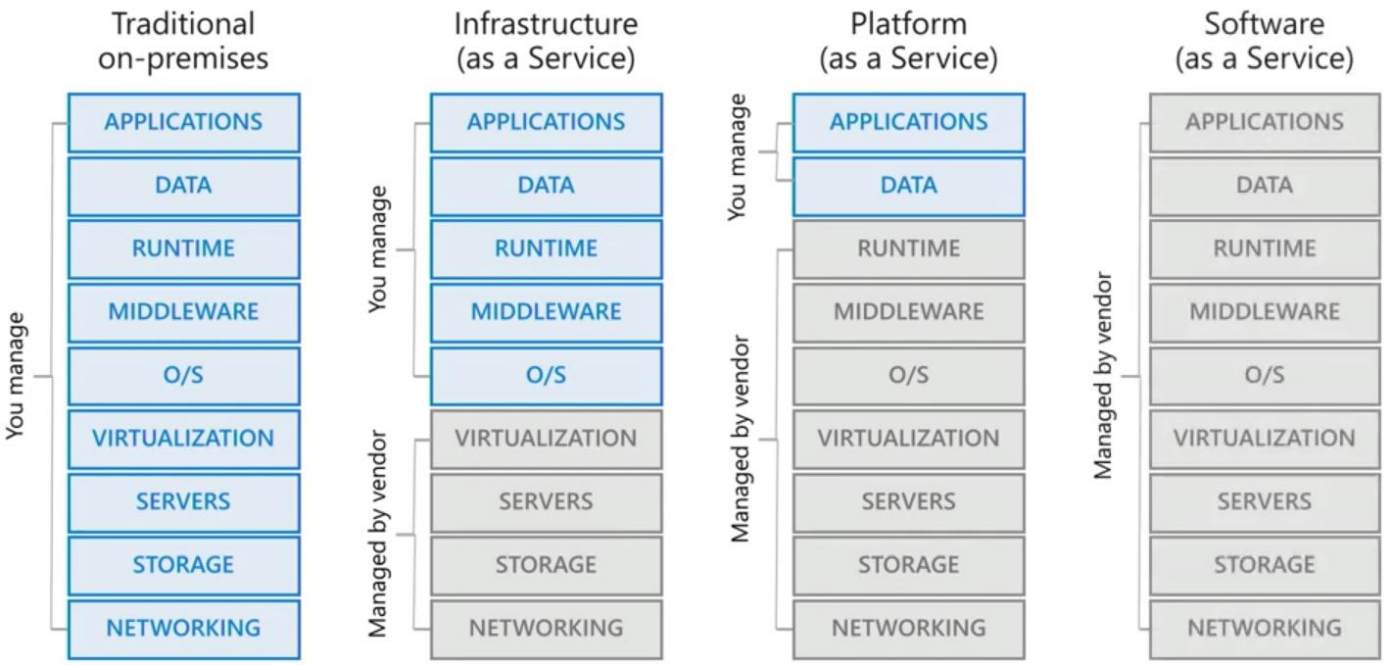
Discovering Microsoft Azure

Session Resources
aka.ms/AFUN10

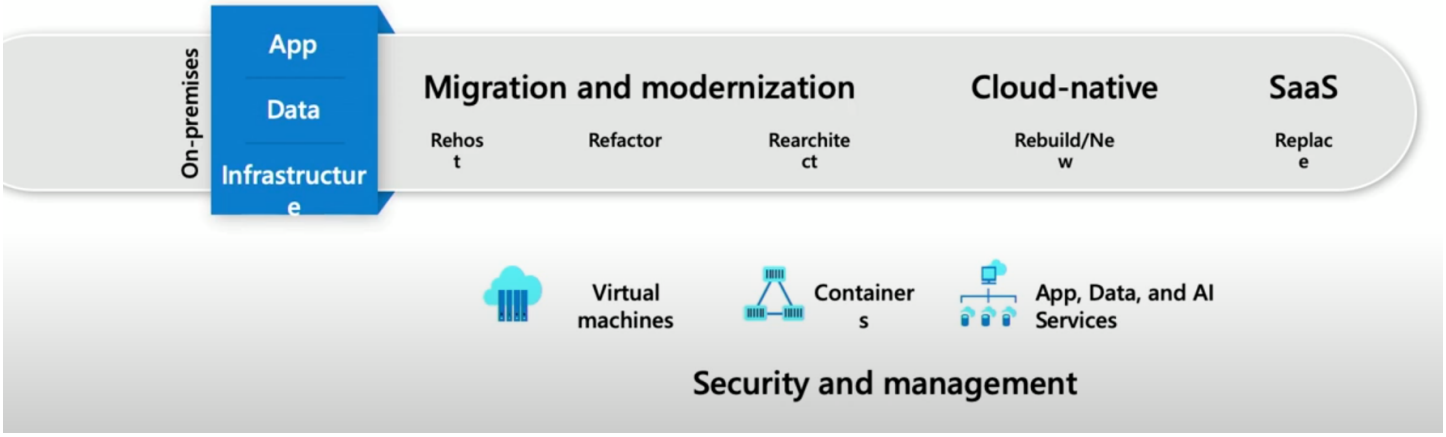
Session Code on GitHub
aka.ms/AFUN10Repo

All Event Resources
aka.ms/mymsignitethetour

Cloud Service Models



The journey to the cloud



Why move to the cloud?

Cost effective
- Pay-as-you-go pricing
- Pay only for the resources you use

Scalable
- Vertically scale resources
  - Adding a faster CPU
  - Adding memory
- Horizontally scale
  - Add more servers

Elastic
- Automatically add or remove resources
- Add resources when your application is most-heavily usec
- Remove resources when unnecessary

Why move to the cloud?

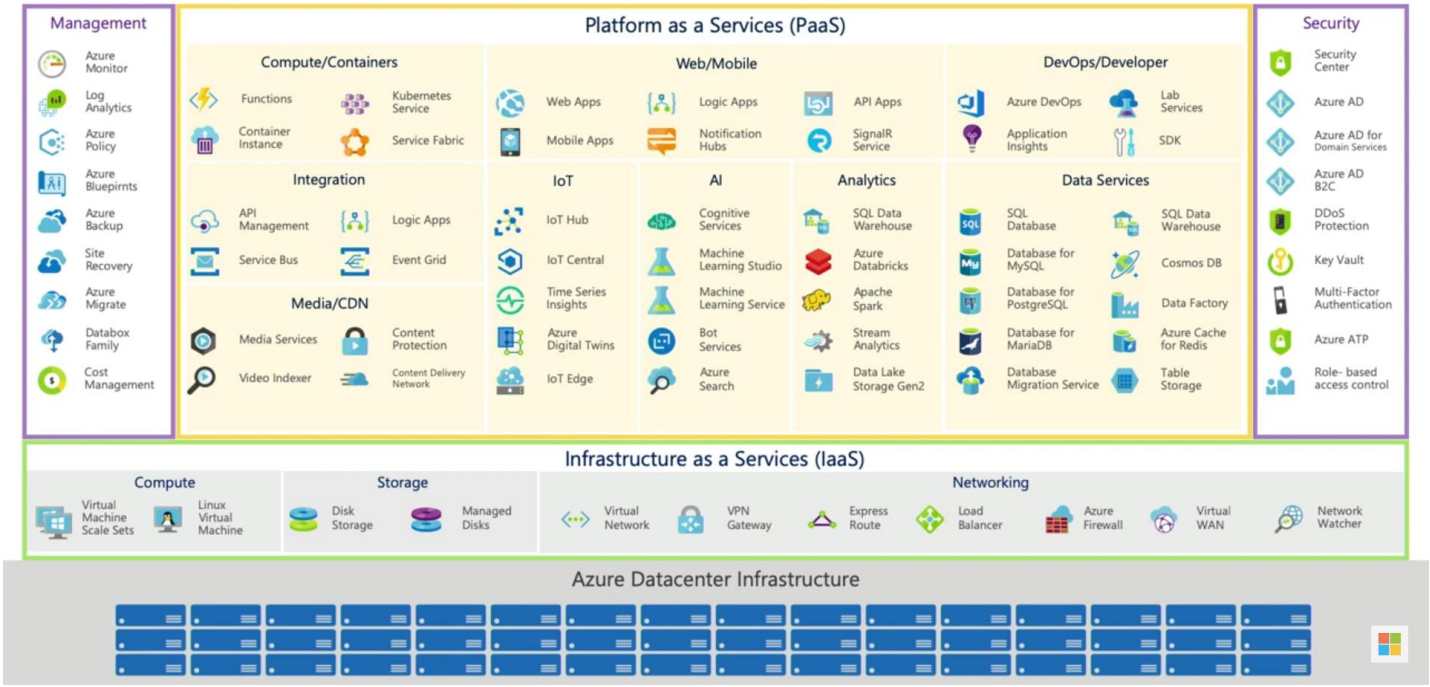
Current
- Focus on building and deploying applications
- Maintenance is done for you
  - No more software patching, hardware setup, upgrades and IT management

Reliable
- Your data is safe
- Azure provides:
  - Data backups
  - Disaster recovery
  - Data replication

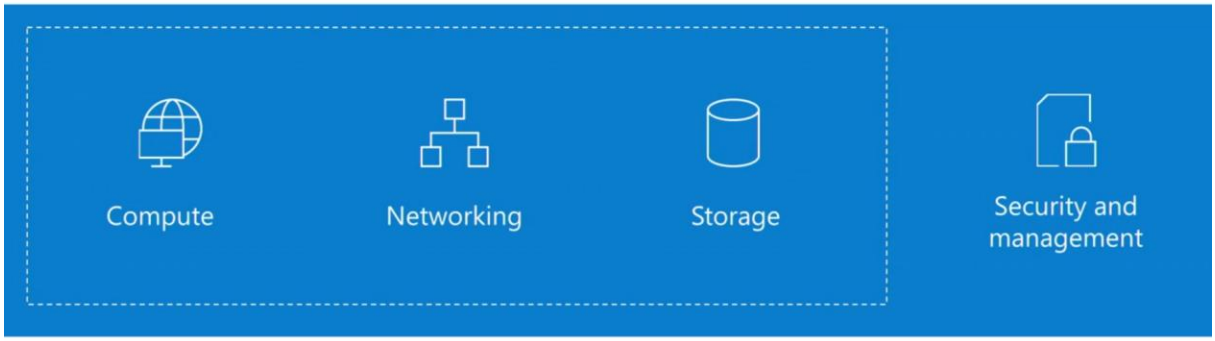
Secure
- Physical security
- Digital security

TO GETTING UP AND RUNNING MORE QUICKLY

What is Azure?



Azure is secure, cost-effective, and ubiquitous



Azure compute services

Azure Virtual machines

Windows or Linux VMs

Azure Kubernetes service

Manage clusters of VMs that run containerized services

Azure Functions

Event-driven, serverless computing

Azure networking services

Azure Virtual Network

Connects VMs to VPN connections

Azure Load Balancer

Balances inbound and outbound connections

Azure Traffic Manager

Distributes network traffic across Azure regions

Azure storage services

Azure blob storage

Stores objects like video files, JSON, images, IoT data

Azure file storage

Acts as a file server to share and access files

Azure table storage

NoSQL storage for unstructured data

What is Azure Active Directory?

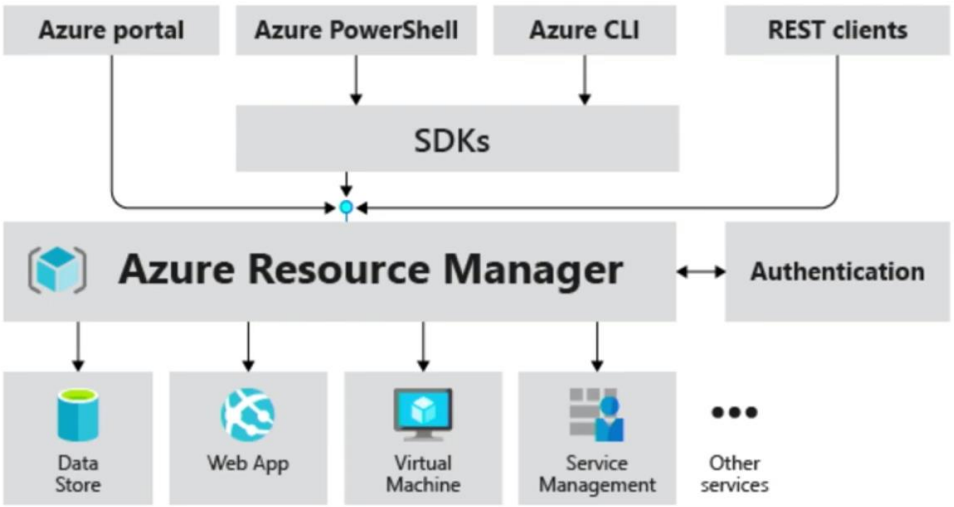
Azure AD is a multi-tenant, cloud-based directory and identify management service

Centralized directory store

Used by Azure and Office 365

Contains all the identities of users in your organization

Azure Resource Manager



How the Azure Resource Manager Works

Azure Resource Manager

Consistent management layer

See components as related and independent parts of your network

Deploy, manage, and monitor resources as a group

Provides security, auditing, and tagging

Azure Resource Manager

Resource

A manageable item available through Azure. VMs, web apps, databases, etc.

Resource group

A container that holds related resources

You decide how to allocate resources to groups

Resource provider

A service that supplies the resources you can deploy and manage

Resource manager template

A JSON file that defines resources to deploy to a resource group

Defines dependencies between resources

Common resource providers

Microsoft.Compute

Supplies the VM resource

Microsoft.Storage

Supplies the storage account resource

Microsoft.Web

Supplies resources related to web applications

Subscriptions, resource groups, and resources

azuredeploy.json

```
},
 "resources": [
  {
   "type": "Microsoft.Storage/storageAccounts",
   "apiVersion": "2018-11-01",
   "name": "[variables('storageAccountName')]",
   "location": "[parameters('location')]",
   "sku": {
    "name": "[variables('storageAccountType')]"
   },
   "kind": "Storage",
   "properties": {}
  },
```

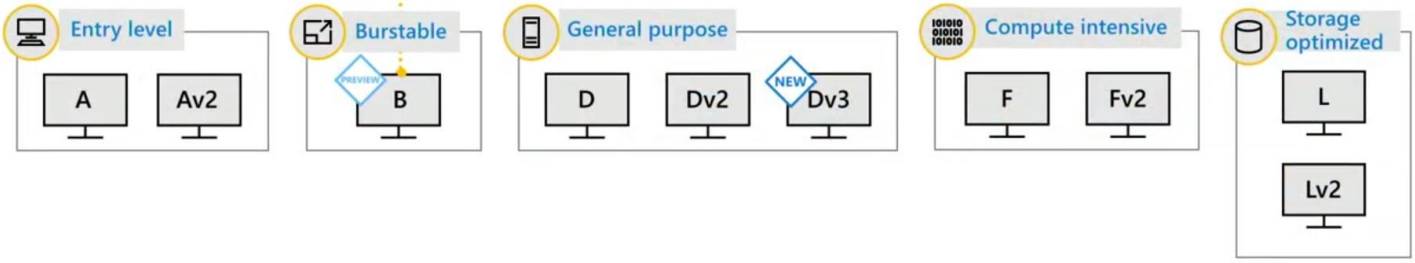
Azure Resource Manager Template

A JSON file that defines resources to deploy to a resource group

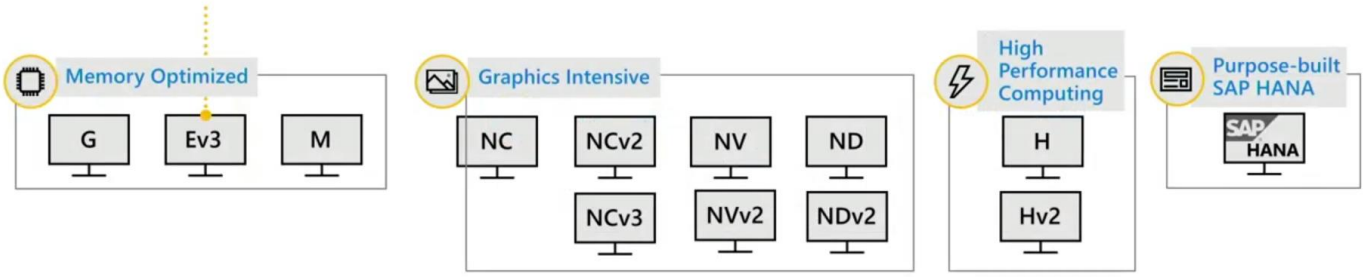
Defines dependencies between resources

Compute Options

Intel Haswell E5-2673 v3
Lowest cost, flexible CPUs
Up to 8vCPUs, 32GB RAM



Intel Xeon E5-2673 v4 (Broadwell)
Hyper-Threaded CPUs
Up to 64 vCPUs, 432GB RAM



What's a Virtual Machine (VM)?

Software emulation of a physical computer

Includes:

- Virtual processor
- Memory
- Storage
- Networking resources

Unlike containers, VMs host an OS

Creating a VM in Azure

Takes 5 minutes

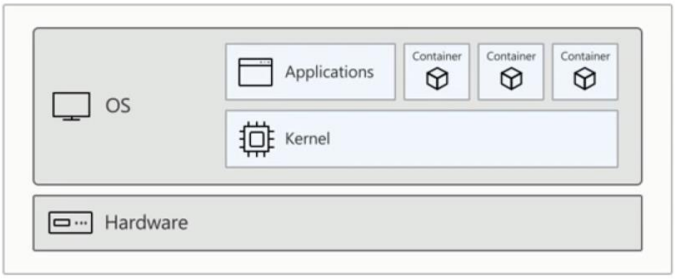
Select a pre-configured VM image

- A template used to create a VM
- Includes an OS and usually some other software

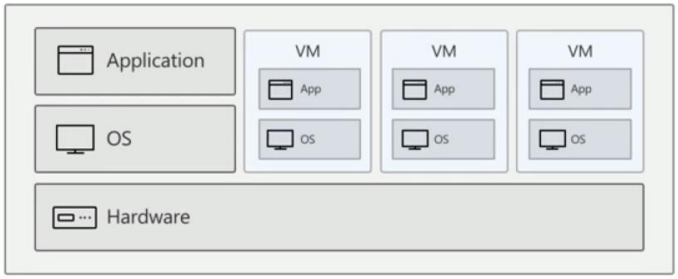
Host OS, RD connections as in real computers. Unlike containers, VM includes whole operating system

What's a container?

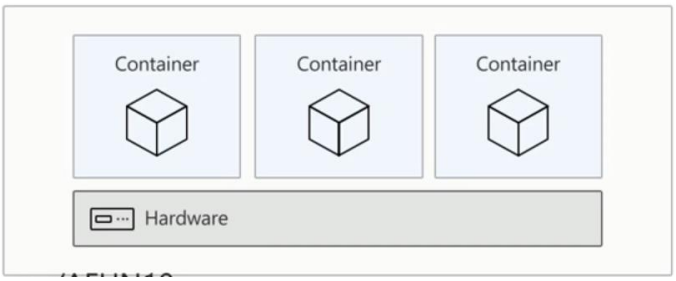
Containers = operating system virtualization



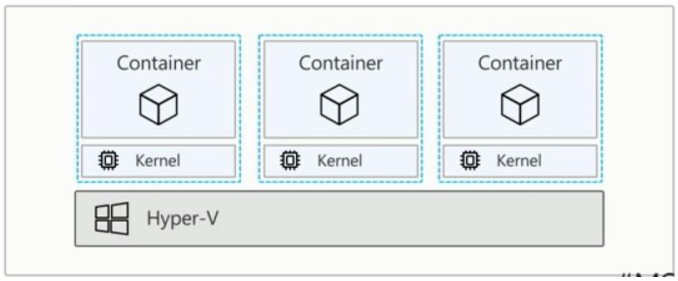
Traditional virtual machines = hardware virtualization



Windows Server containers: maximum speed and density

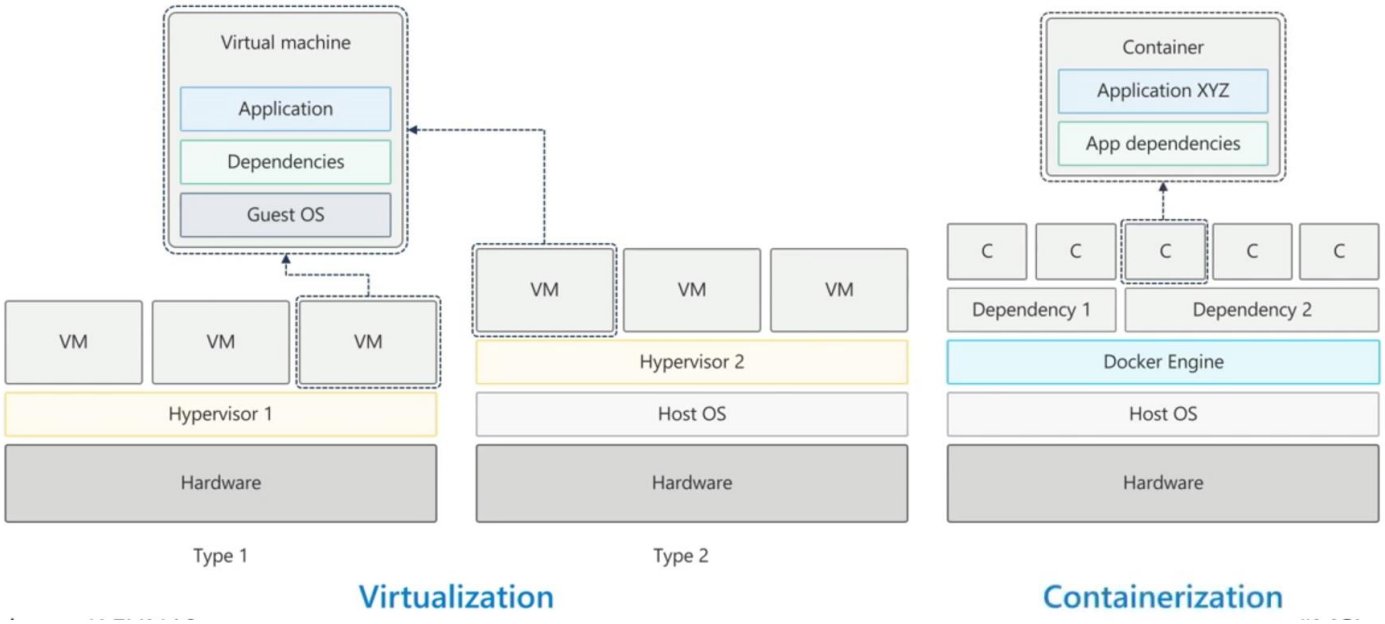


Hyper-V containers: isolation plus performance

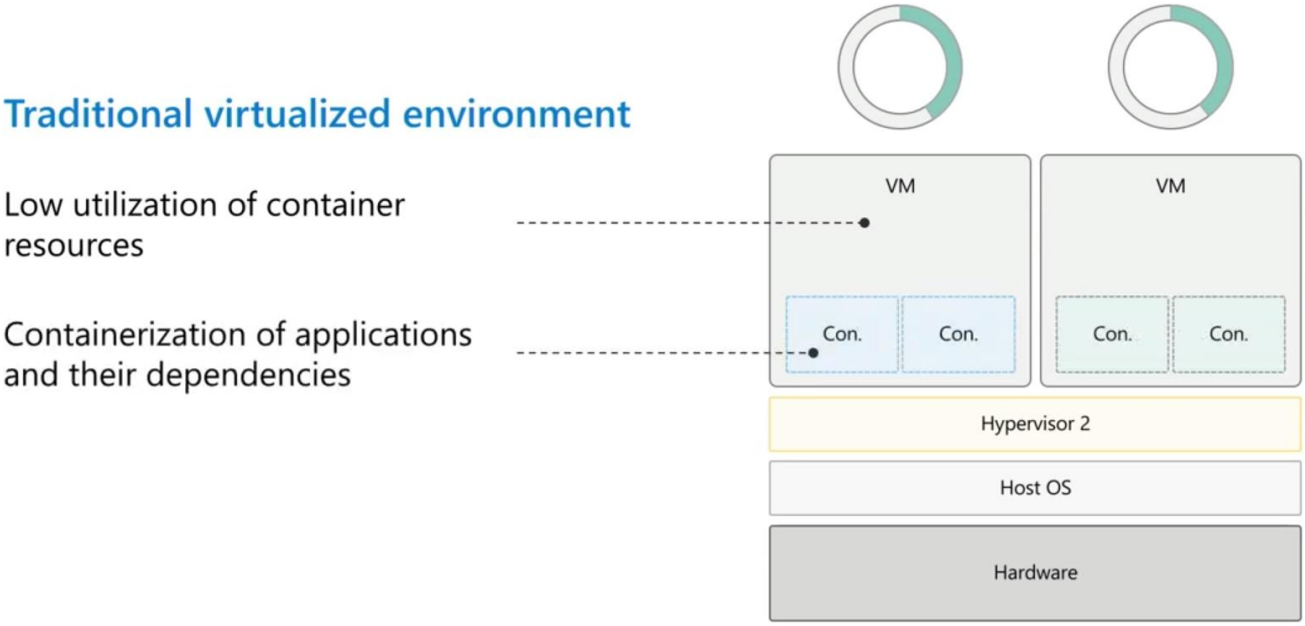


The biggest diff betn container vs VM is Container doesn't include OS. It references the OS, the KERNEL. Just includes whatever libraries & additional details to run the particular applications.

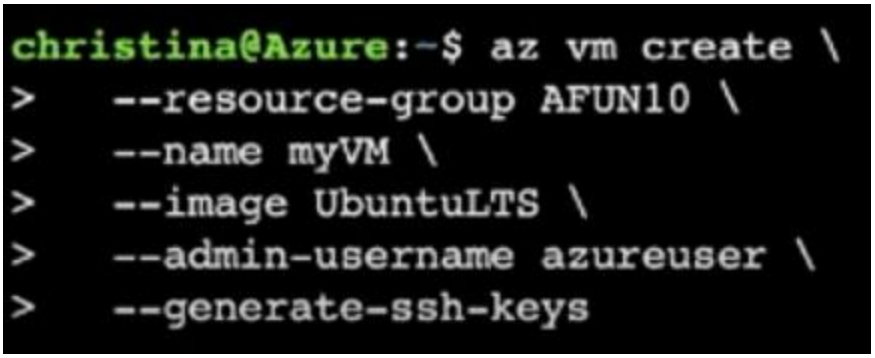
Virtualization vs. containerization



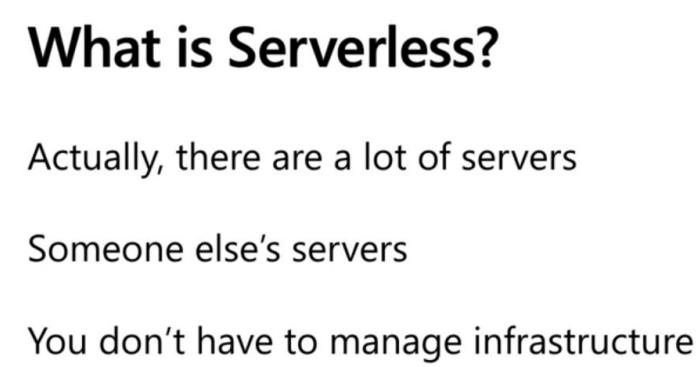
The container advantage



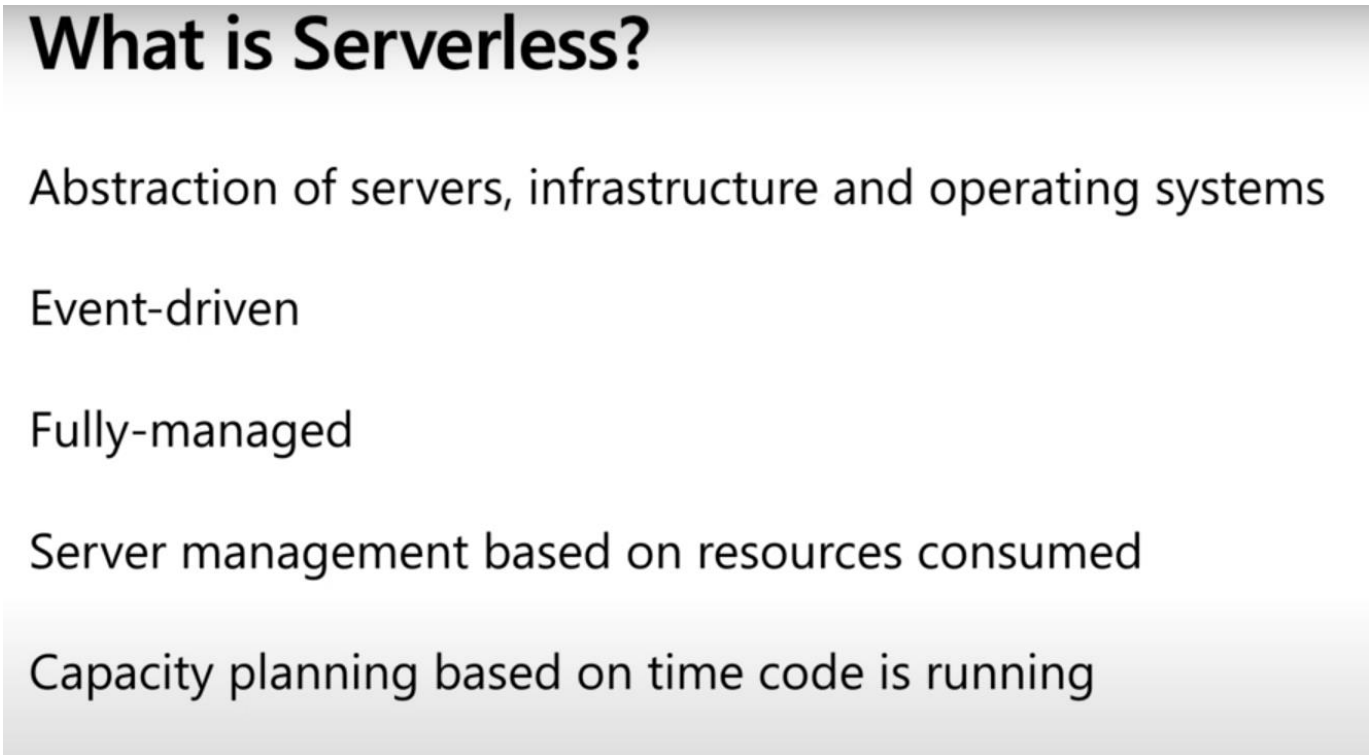
Installing VM with CLI



Using CLI you don't have to generate SSH keys separately. But you need permanent storage account to hold SSH Keys



Serverless is cloud-hosted execution environment. It let's you run your code, but you're completely abstracted from hosting environment.



Event-driven --> Only going to pay when code actually runs. You can have a lot of serverless code available unless it running you won't be charged

Serverless in Azure

Microsoft Flow
- Built on Logic Apps
- Create simple integrations
- Designed for non-tech employees

Azure Logic Apps
- Advanced Flow for developers, operations folks and IT pros

Azure Functions
- Serverless compute service
- Runs locally and in the cloud

Azure App Service Webjobs
- Run scripts in the context of an App Service

Azure Functions

Small pieces of code, easy to run

Develop serverless applications on Azure

Pay only for the time your code runs

Use your development language of choice
- Supported languages: C#, JavaScript, F#, Java, TypeScript

Simplify integration with Azure services

Functions runtime is opensource

also python

PAAS Service

Azure App Service

Build and host web apps in your language of choice

Auto-scaling and high availability

Automated deployments from GitHub and Azure DevOps

Quickstarts

- .NET Core
- PHP
- Node.js
- Java
- Python
- Ruby
- Docker

Takes advantage of Containers

With app services

App service will create a container instance, looks up the libraries of code and creates container with those dependencies. So everything is taken care of.

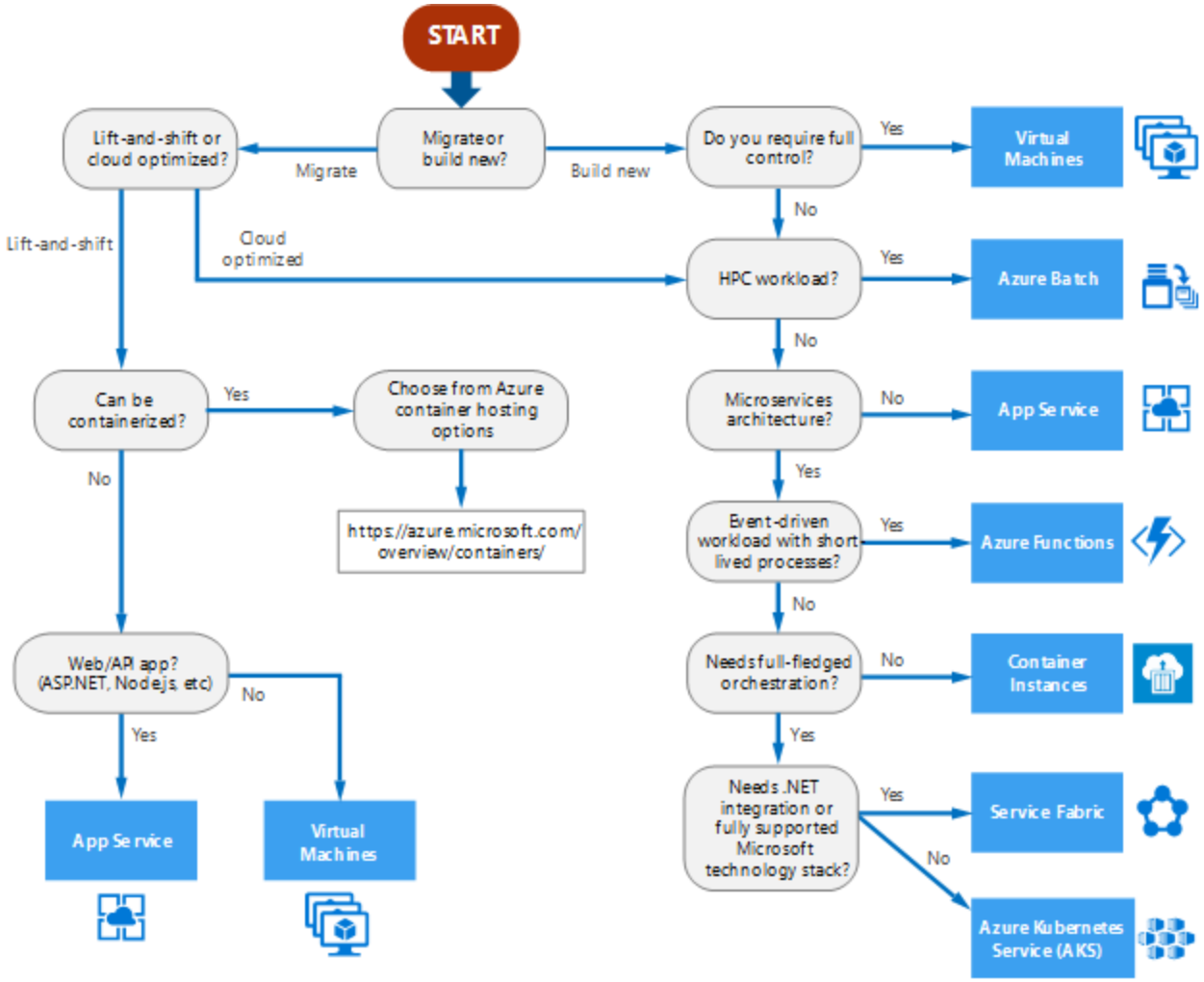
Azure App Service

Windows or Linux

Web App or Web App for Containers

VS Code Extension

Compute service decision tree



Getting Started with Azure

Aka.ms/afunstart