Microsoft

Modernizing Applications with Containers and Orchestrators

Microsoft Services

Microsoft

Module 5 – Container Orchestrators

Microsoft Services

Agenda

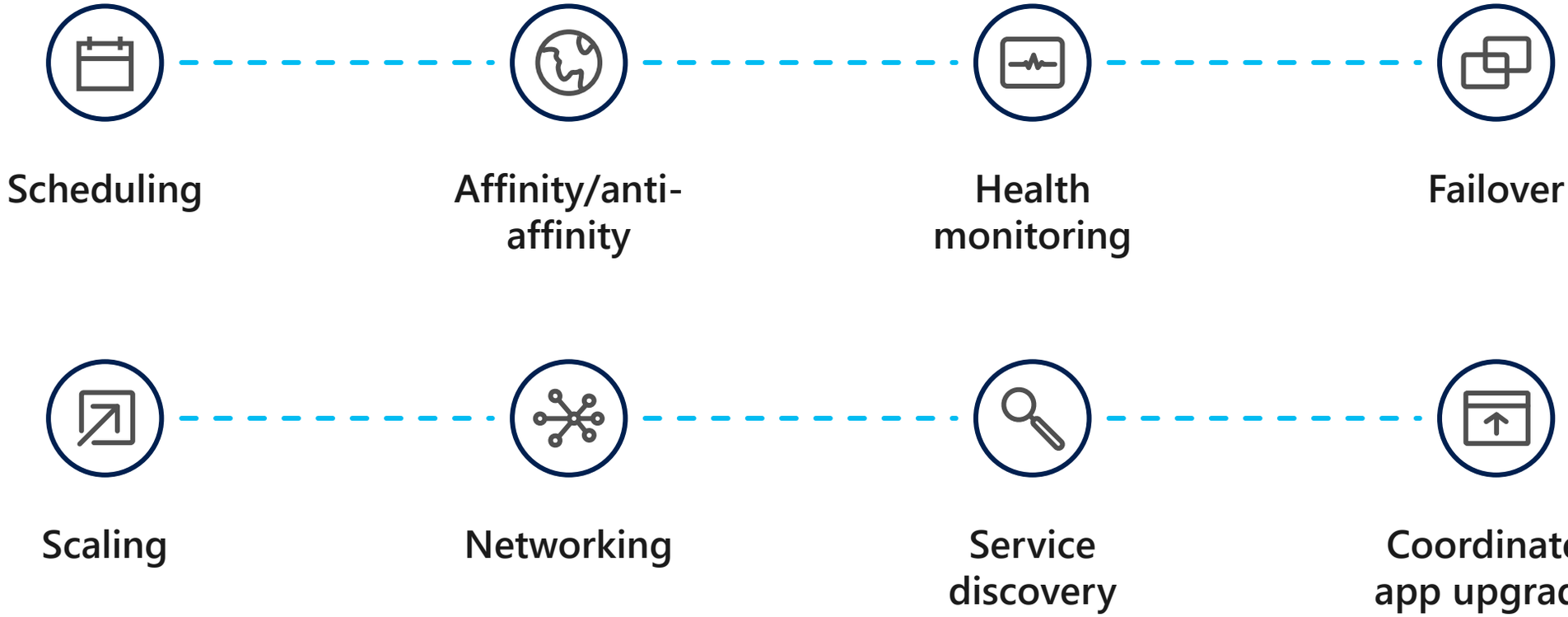
- What Is Orchestration?
- Microsoft Offerings For Containers
- Additional Options For Orchestration

Challenges Of A Containerized World

As application development has moved towards a container-based approach, the need to orchestrate and manage the inter-connected resources becomes important

- Load Balancing
  - Distributing traffic across containers at scale
- Naming and Discovery
  - How do containers or groups find one another?
- Logging and Monitoring
  - Keeping track of what containers are doing
- Debugging
  - Getting inside running containers
- Networking
  - Differentiating container networks from host networks at scale

The Elements Of Orchestration



Clustering Versus Orchestration

Clustering

- Grouping “hosts”—either VMs or bare metal—and networking them together
- A cluster should feel like a single resource rather than a group of disparate machines

Orchestration

- Managing and monitoring of the workloads running in your cluster
- Starting containers on appropriate hosts and connecting them
- May also include support for scaling, automatic failover, and node rebalancing

Microsoft Offerings for Containers

IF YOU'RE LOOKING FOR THIS...	USE THIS
Scale and orchestrate containers using Kubernetes	Kubernetes Service
Easily run containers on Azure with a single command	Container Instances
Store and manage container images across all types of Azure deployments	Container Registry
Develop microservices and orchestrate containers on Windows or Linux	Service Fabric
Deploy web applications on Linux using containers	App Service
Run repetitive compute jobs using containers	Batch

Azure App Service

Easily deploy and run container-based web apps at scale

Accelerated outer loop

Tight integration w/ Docker Hub, Azure Container Registry

Built-in CI/CD w/ Deployment Slots

Intelligent diagnostics & troubleshooting, remote debugging

Fully managed platform

Automatic scaling and load balancing

High availability w/ auto-patching

Backup & recovery

Flexibility & choices

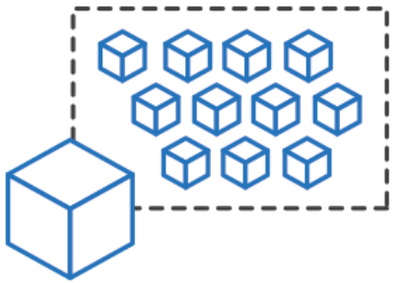
From CLI, portal, or ARM template

Single Docker image, multi container w/ Docker Compose

IntelliJ, , Jenkin, Maven Visual Studio family

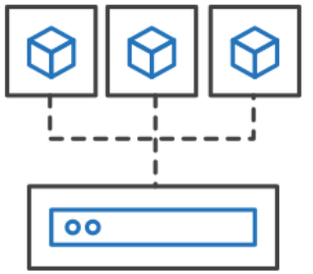
Azure Container Instances (ACI)

Easily run containers on Azure without managing servers



Run containers without managing servers

Increase agility with containers on demand



Secure applications with hypervisor isolation

Azure Container Instances (ACI)

Great for:

- Isolated Windows and Linux containers
- Simple applications
- Task automation
- Build jobs
- Hypervisor-level security
- Custom sizes for CPU cores and memory
- Public IP connectivity
- Persistent storage
- Co-scheduled groups

NOT great for:

- Full container orchestration
- Service discovery across multiple containers
- Automatic scaling
- Coordinated application upgrades

HashiCorp Nomad

- **Docker Support:** Jobs specify the number of container instances needed and Nomad will handle placement and recover from failures automatically
- **Operationally Simple**: Nomad combines features of both resource managers and schedulers into a single system
- **Multi-Datacenter and Multi-Region Aware**
- **Flexible Workloads**: Nomad has extensible support for task drivers, allowing it to run containerized, virtualized, and standalone applications
- **Built for Scale**
- **Synergy with HashiCorp Terraform, Consul and Vault**

Mesosphere DC/OS

Integrate Platform for Data and Containers

- Mesophere DC/OS is built on Apache Mesos. Mesos simplifies administration and maximizes resource utilization by abstracting the datacenter into a single giant computer
  - Run Docker and other container image types with the reliability of Mesos
  - Get the benefit of container network interface (CNI) support and GPU-based acceleration
- Marathon is a framework for container and services orchestration typically used with Mesos and DC/OS
- Kubernetes can also be used for orchestration on top of DC/OS
- Typically used in **large-scale clusters** with node counts exceeding 10,000

https://info.mesosphere.com/The-Docker-vs-Kubernetes-vs-Apache-Mesos-Myth_RegistrationPage.html

Rancher

OSS for delivering Kubernetes-as-a-Service

- Enables you to deploy new clusters from scratch, launch EKS, GKE and AKS clusters, or even import existing Kubernetes clusters
- “One Platform for Everyone who uses Kubernetes”
- Useful for multi-cloud scenarios

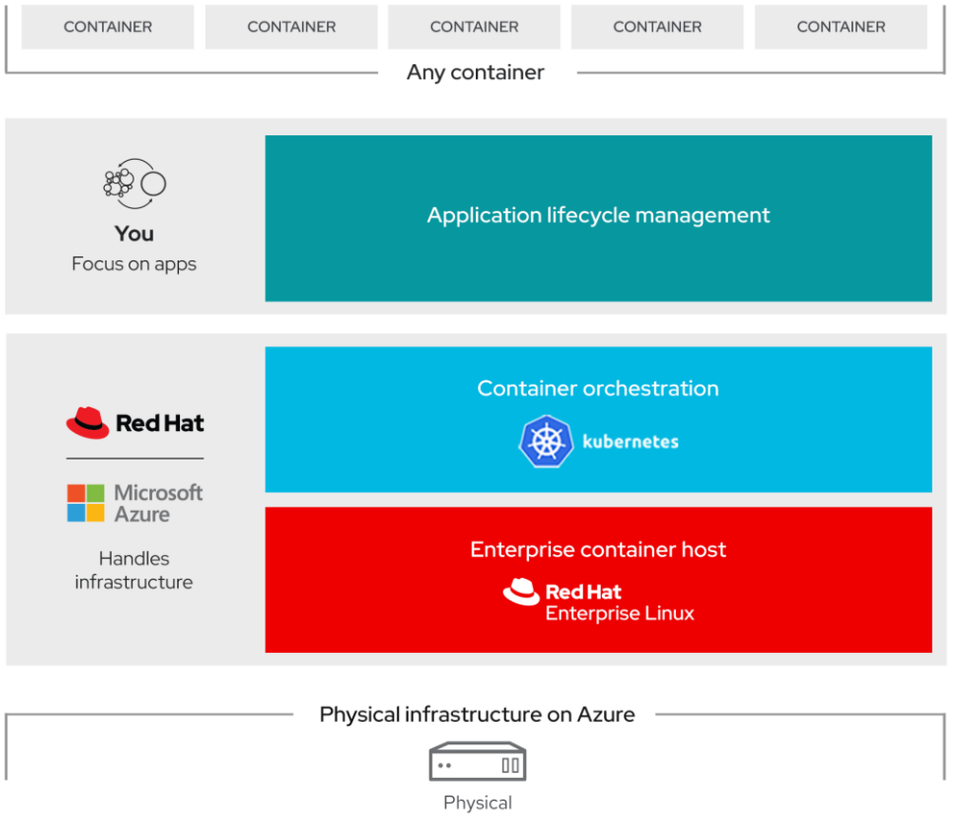
EC2, Azure, GCE, Digital Ocean	Rancher provisions compute instances, installs Kubernetes onto them, and then manages the full lifecycle of all resources. This allows you to benefit from an IaaS platform while running a Kubernetes-managed container cluster.
GKE, EKS, AKS	Rancher provides full management of the cloud resources themselves, including the ability to spin resources up and down. However, instead of learning different interfaces each time you switch clouds or managing accounts and access between them, Rancher provides a common and consistent view of each of these hosted services. It centralizes RBAC and keeps your clusters secure.

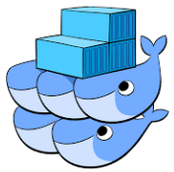
Red Hat OpenShift

- Extends the capabilities available in Kubernetes
  - *"We package Kubernetes and include additional tooling as features that we find important and our users demand"*
- Enhances the enterprise user experience by adding features to enable rapid application development, easy deployment, and lifecycle maintenance
- Can deploy and support anywhere Red Hat Enterprise Linux is deployed and supported (i.e. AWS, Azure, GCP, VMWare)
- Istio available as a technology preview in OpenShift

Microsoft Azure Red Hat OpenShift

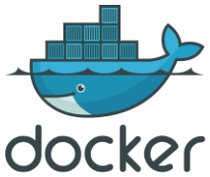
Azure Red Hat OpenShift is a fully managed service of Red Hat OpenShift on Azure, jointly, engineered, operated and supported by Microsoft and Red Hat





Docker Swarm

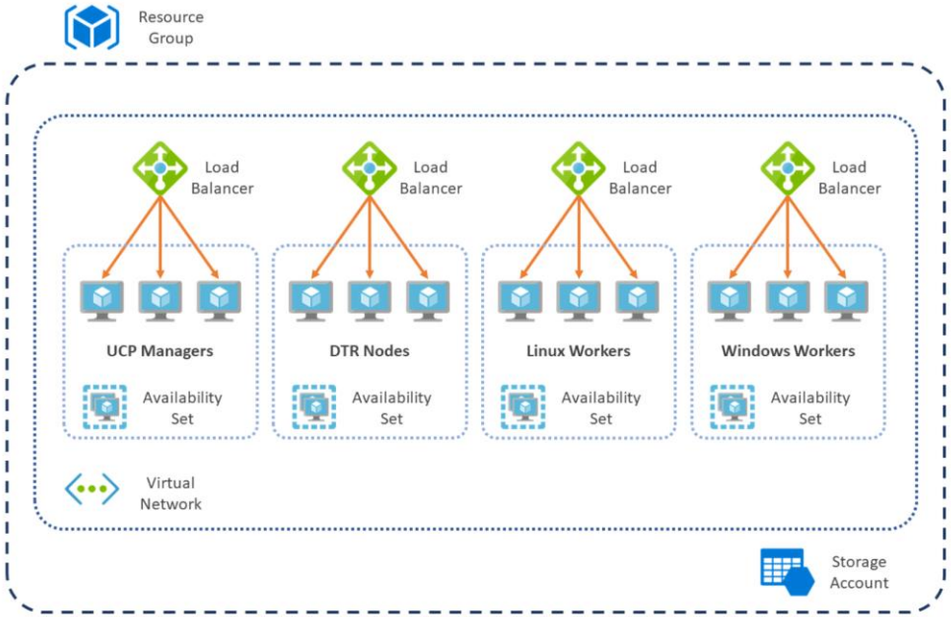
- Docker Swarm is native clustering for Docker using the native Docker API
- Swarm mode is a Docker feature that provides built in container orchestration capabilities, including native clustering of Docker hosts and scheduling of container workloads
- The workflow for managing containers on a Docker Swarm is almost identical to what it would be on a single container host
  - Uses the same artifacts, so you can run your app across 50 containers in a 20-node swarm and the functionality will be the same as when you run it in a single container on your laptop



Docker Certified Infrastructure for Azure

https://success.docker.com/article/certified-infrastructures-azure

Docker Certified Infrastructure is Docker's prescriptive approach to deploying Docker Enterprise on a range of infrastructure choices



Kubernetes

- **Kubernetes** is "an open-source software for automating deployment, scaling, and management of containerized applications"
- **Kubernetes**, in Greek κυβερνήτης, means the Helmsman, or pilot of the ship
- Keeping with the maritime theme of **Docker** containers, **Kubernetes** is the pilot of a ship of containers

Kubernetes

The de-facto orchestrator

Portable

Public

Private

Hybrid

multi-cloud

Extensible

Modular

pluggable

Hookable

composable

Self-healing

Auto-placement

auto-restart

auto-replication

auto-scaling

Kubernetes

Empowering you to do more

Deploy your applications quickly and predictably

Scale your applications on the fly

Roll out new features seamlessly

Limit hardware usage to required resources only

Azure Kubernetes Engine

Open source: https://github.com/Azure/aks-engine

- Easiest way to provision a self-managed Kubernetes cluster on Azure
- Leverages Azure Resource Manager (ARM), to help you create, destroy and maintain clusters provisioned with basic IaaS resources in Azure
- Allows you to customize Deployments
  - Deploying into existing virtual networks
  - Utilizing multiple agent pools

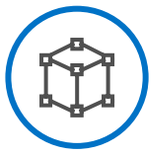
Azure Kubernetes Service

Simplify the deployment, management, and operations of Kubernetes

Deploy and manage Kubernetes with ease

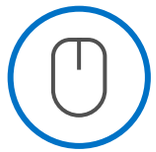
Scale and run applications with confidence

Secure your Kubernetes environment



Accelerate containerized application development

Work how you want with open-source tools & APIs



Set up CI/CD in a few clicks

Azure Kubernetes Service

Windows Support

- Windows support available in Preview on AKS
  => Only Windows Server 2019 agent nodes
- AKS cluster with Windows Server feature will be composed of a Windows node pool and a Linux node pool
- Use **Node Selector** to specify on which nodes the workloads have to be deployed

Demonstration: *Kubernetes Cluster in Azure Kubernetes Service*

Deploy Kubernetes clusters in Azure Kubernetes Service

Deploy NGINX container into Kubernetes cluster