Elevator Pitch

Let's see, you can run containers in these Azure services: App Service, Functions, ACI, ACA, and AKS. So which one should you use? In this session, we try to find what workload type best runs in each service, and see how to move to a different service if you outgrow the current one.

Description

Let's see, you can run containers in these Azure services: App Service, Functions, Container Instances (ACI), Container Apps (ACA), and Azure Kubernetes Service (AKS). So which one should you use? Like tools in a toolbox, each service can run different workload types. In this session, we try to find what workload type best runs in each service, and see how to move to a different service if you outgrow the current one.

Containers on Azure
Why so many choices?

Gold Sponsors

neo

SOLVERA
Part of Accenture

NORTHFIELD

online business systems

Microsoft

LOTLINX®

Johnston group

Community Supporters

iQmetrix®

RICHARDSON

Agenda

- Why containers?
- Look at the Azure services that can run containers
- Look at the use cases / workloads
- Conclusion

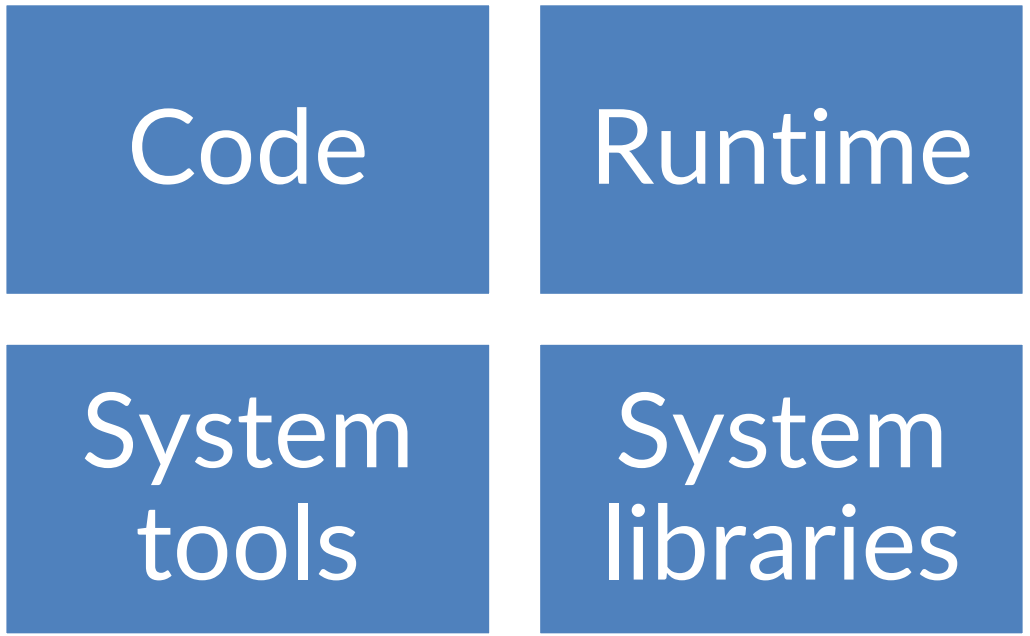
Who am I?

- Guy Barrette
- Dev/Coach/Trainer
- Based in Montreal, Canada
- @GuyBarrette
- linkedin.com/in/guybarrette
- guybarrette.com

Why Containers?

What is a container?

A unit of software/deployment



Containers = Kubernetes

Containers on Azure

Azure | Explore | Products | Solutions | Pricing | Partners | Resources | Search | Learn | Support | Contact Sales | Free account | Sign in

Search all products

View all products (200+)

Results will display instantly

- Popular
- AI + machine learning
- Analytics
- Compute
- Containers
- Databases
- DevOps
- Developer tools
- Hybrid + multicloud
- Identity
- Integration
- Internet of Things
- Management and governance
- Media

Containers

Develop and manage your containerized applications faster with integrated tools

Azure Kubernetes Service (AKS)
Deploy and scale containers on managed Kubernetes

Azure Red Hat OpenShift
Deploy and scale containers on managed Red Hat OpenShift

Azure Container Apps
Build and deploy modern apps and microservices using serverless containers

Azure Functions
Execute event-driven serverless code functions with an end-to-end development experience

Web App for Containers
Run containerized web apps on Windows and Linux

Azure Container Instances
Launch containers with hypervisor isolation

Azure Service Fabric
Deploy and operate always-on, scalable, distributed apps

Azure Container Registry
Build, store, secure, and replicate container images and artifacts

Chat with Sales

Containers

Develop and manage your containerized applications faster with integrated tools

Azure Kubernetes Service (AKS)

Deploy and scale containers on managed Kubernetes

Azure Red Hat OpenShift

Deploy and scale containers on managed Red Hat OpenShift

Azure Container Apps

Build and deploy modern apps and microservices using serverless containers

Azure Functions

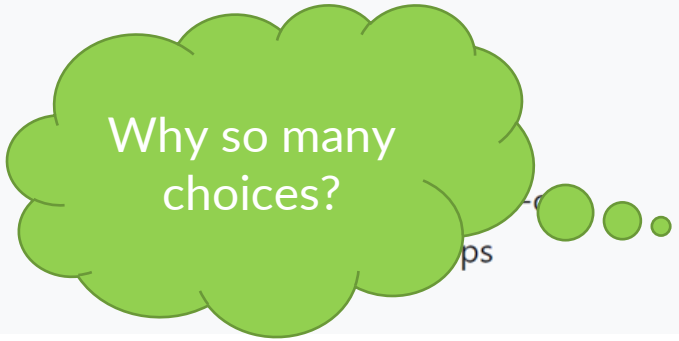
Execute event-driven serverless code functions with an end-to-end development experience

Web App for Containers

Run containerized web apps on Windows and Linux

Azure Container Instances

Launch containers with hypervisor isolation



Azure Container Registry

The right tool for the right job

Azure Container Registry

What are Container Registries?

- Central repositories for container images
- Private or/or public
- Docker Hub
  - hub.docker.com
- Microsoft
  - Azure Container Registry
  - Microsoft Container Registry (public images)
    - mcr.microsoft.com
- Elsewhere
  - Amazon Elastic Container Registry
  - Google Container Registry

Azure Container Registry

- Private or public (Premium) registries
- Your images are stored near the services that will run them
  - No hop over the Internet
- You can secure access using Azure AD, RBAC roles, policies
- You can also store Helm charts
- Task base compute for building containers
  - Linux, Windows, and ARM
  - Triggers: public or private Git repository in GitHub or Azure DevOps

What use cases?

Container Registry - Use Cases

- Private
- Near the compute ressources
- Security
- Build

Azure App Service

App Service

=

Web Apps

Web Apps
For Containers

App Service Benefits

- Windows or Linux
- Automatic OS patching
- High availability
- Automated scale out/in
- Built-in load balancing
- Compliances: ISO, SOC, PCI, etc

Web App for Containers

- All App Service benefits plus…
- Integration with
  - Docker Hub, Azure Container Registry
  - Azure Storage, Key Vault
- Single Docker image
- Multi containers
  - Docker compose
- Windows containers
  - Install drivers, tools, COM objects

What workload?

App Service - Workloads

Azure Functions

Azure Functions

- Event driven apps
- Consumption (serverless)
  - Code
- App Service Elastic Premium plans
  - Code & containers
  - Avoid cold starts with warm instances
  - Virtual network connectivity
  - Unlimited execution duration, with 60 minutes guaranteed

Azure Functions

- Instead of deploying your compiled code, you package it in a container
- The base image must include the Functions runtime
- Can run in Kubernetes
  - KEDA must be installed in the cluster
  - Supported triggers in KEDA
    - Azure Storage Queues
    - Azure Service Bus Queues
    - Azure Event / IoT Hubs
    - Apache Kafka
    - RabbitMQ Queue

What workload?

Azure Functions - Workloads

Blocked by something with Code Functions

Run Functions in Kubernetes

Service Fabric

Service Fabric

- Orchestrate microservices and containers
- Windows and Linux
- Stateless and stateful services
- Application platform
- Powers many Microsoft services

What workload?

Service Fabric - Workloads

Azure Container Instances

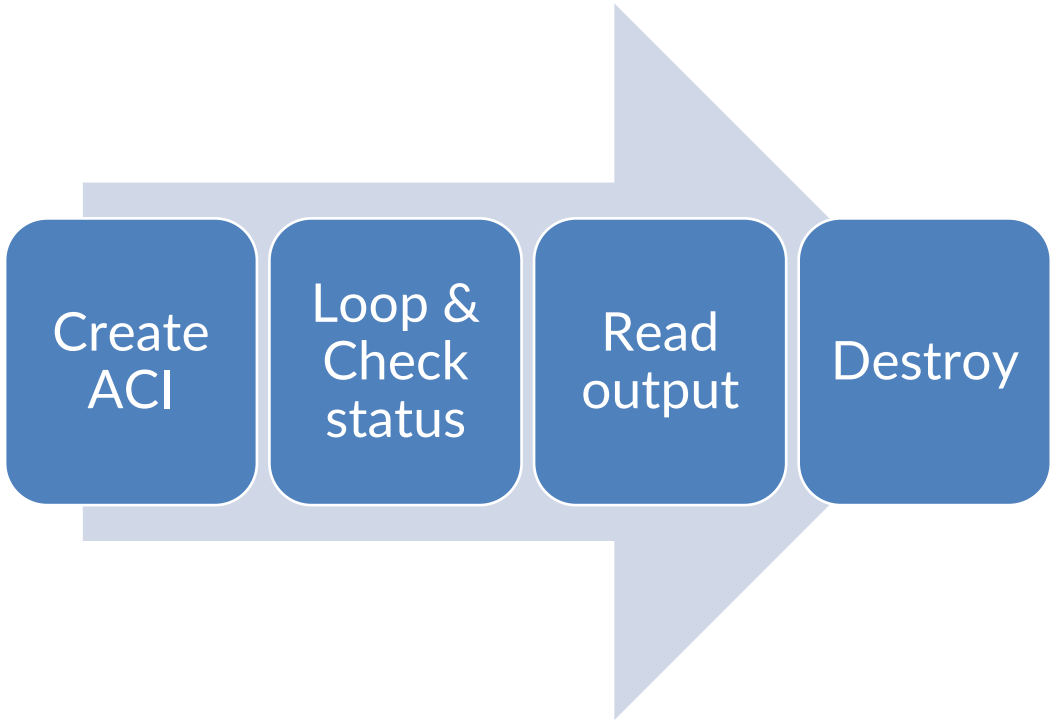
Azure Container Instances

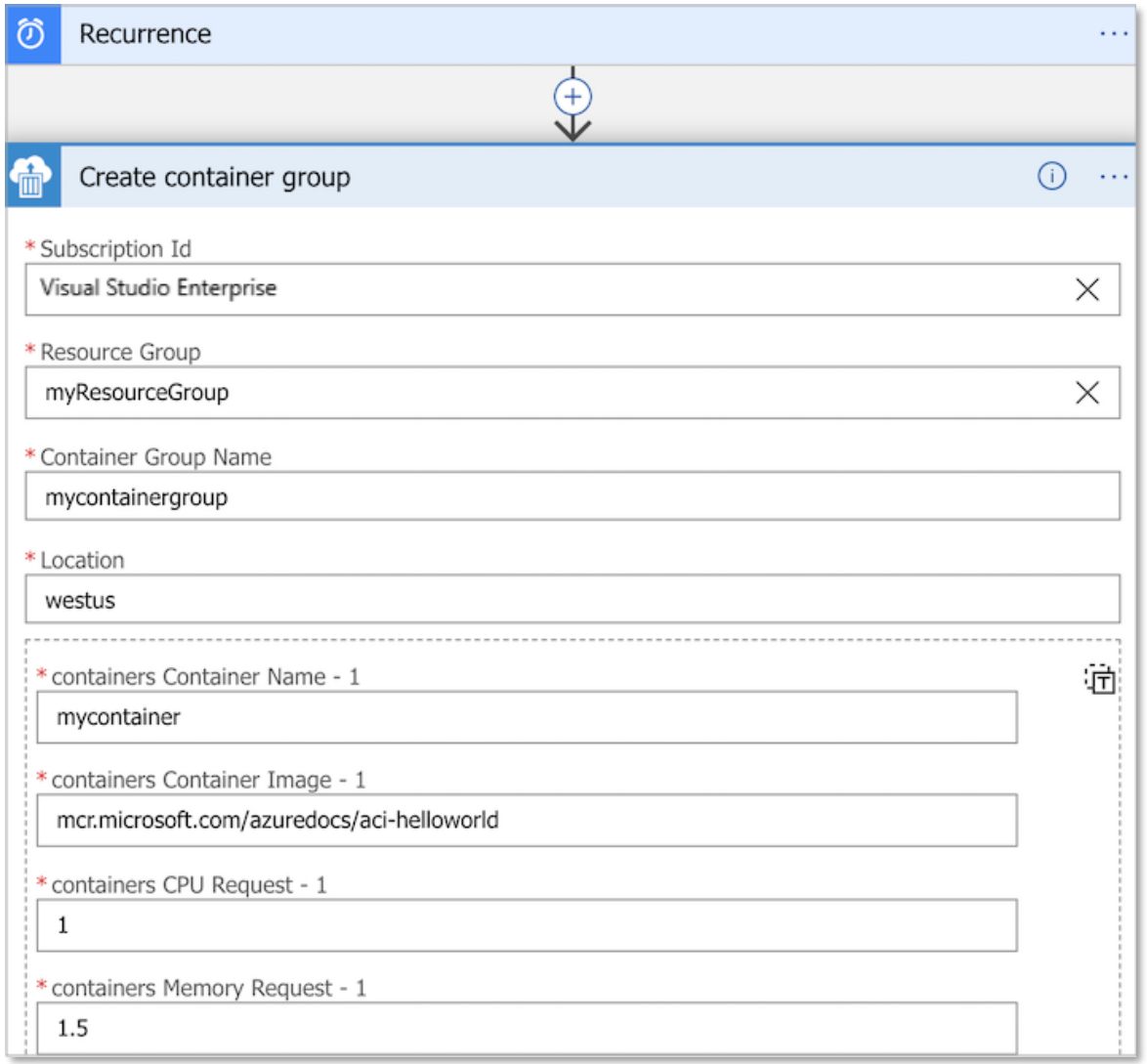
- Per-second billing
- Linux and Windows
- Created and destroyed on demand
- Fast startup time
- Public IP and DNS name
- Supports virtual networks & persistent storage

Azure Container Instances

- Save costs in hosting & maintenance for temporary workloads
- Can act as Kubernetes pods to provide elastic bursting
- Can be created using
  - Azure Portal
  - Docker CLI
  - C# code
  - Logic Apps

Logic Apps


Create ACI
Loop & Check status
Read output
Destroy


Recurrence
Create container group
* Subscription Id
Visual Studio Enterprise
* Resource Group
myResourceGroup
* Container Group Name
mycontainergroup
* Location
westus
* containers Container Name - 1
mycontainer
* containers Container Image - 1
mcr.microsoft.com/azuredocs/aci-helloworld
* containers CPU Request - 1
1
* containers Memory Request - 1
1.5

C#

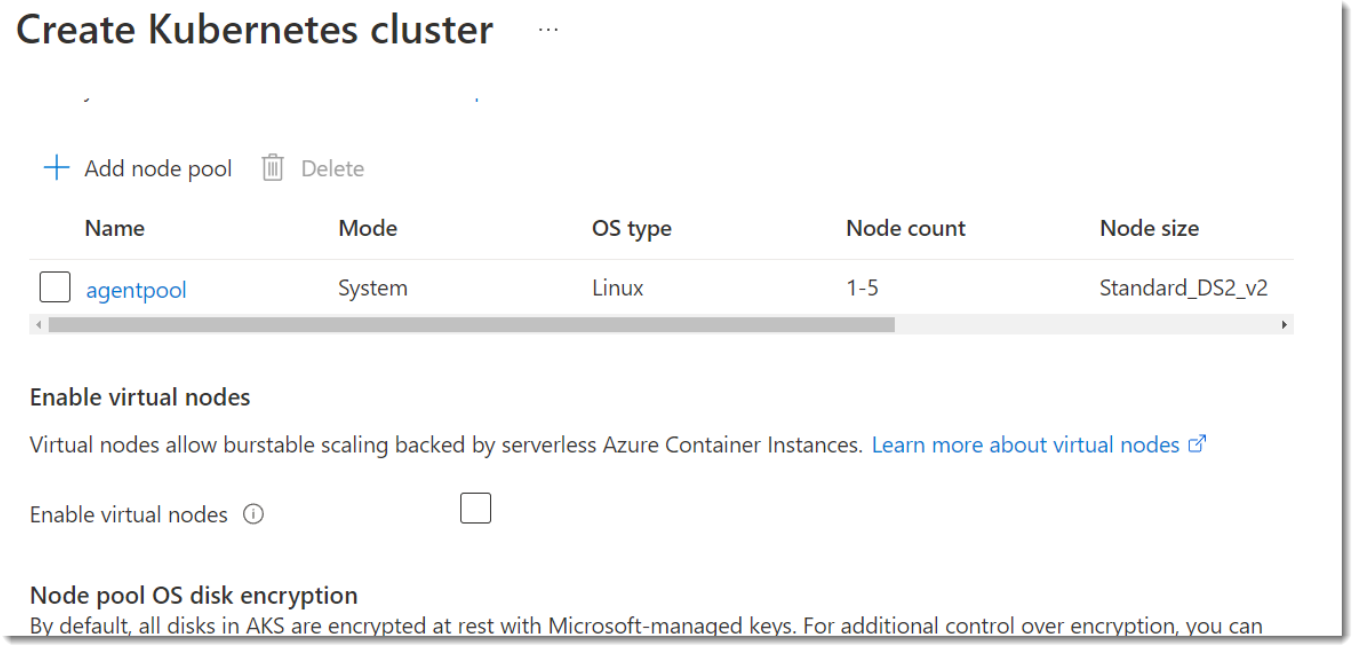
```
using Microsoft.Azure.Management.ContainerInstance.Fluent;
using Microsoft.Azure.Management.ContainerInstance.Fluent.Models;
using Microsoft.Azure.Management.Fluent;

IContainerGroup containerGroup = azure.ContainerGroups.Define(aciName)
  .WithRegion(region)
  .WithNewResourceGroup(rgName)
  .WithLinux()
  .WithPublicImageRegistryOnly()
  .WithoutVolume()
  .DefineContainerInstance(aciName + "-1")
      .WithImage(containerImageName1)
      .WithExternalTcpPort(80)
      .WithCpuCoreCount(.5)
      .WithMemorySizeInGB(1)
      .Attach()
.WithRestartPolicy(ContainerGroupRestartPolicy.Never)
  .WithDnsPrefix(aciName)
  .Create();

SdkContext.DelayProvider.Delay(20000);
containerGroup = azure.ContainerGroups.GetByResourceGroup(rgName, aciName);
string logContent = containerGroup.GetLogContent(aciName + "-1");
Utilities.Log($"Logs for container instance: {aciName}-1\n{logContent}");

azure.ContainerGroups.DeleteById(containerGroup.Id);
```

Kubernetes Virtual Nodes




```
kubectl get nodes

NAME                          STATUS    ROLES     AGE       VERSION
virtual-node-aci-linux        Ready     agent     28m       v1.11.2
aks-agentpool-14693408-0      Ready     agent     32m       v1.11.2
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: aci-helloworld
spec:
  replicas: 1
  selector:
    matchLabels:
      app: aci-helloworld
  template:
    metadata:
      labels:
        app: aci-helloworld
    spec:
      containers:
      - name: aci-helloworld
        image: aci-helloworld
        ports:
        - containerPort: 80
      nodeSelector:
        kubernetes.io/role: agent
        beta.kubernetes.io/os: linux
        type: virtual-kubelet
      tolerations:
      - key: virtual-kubelet.io/provider
        operator: Exists
```

What workload?

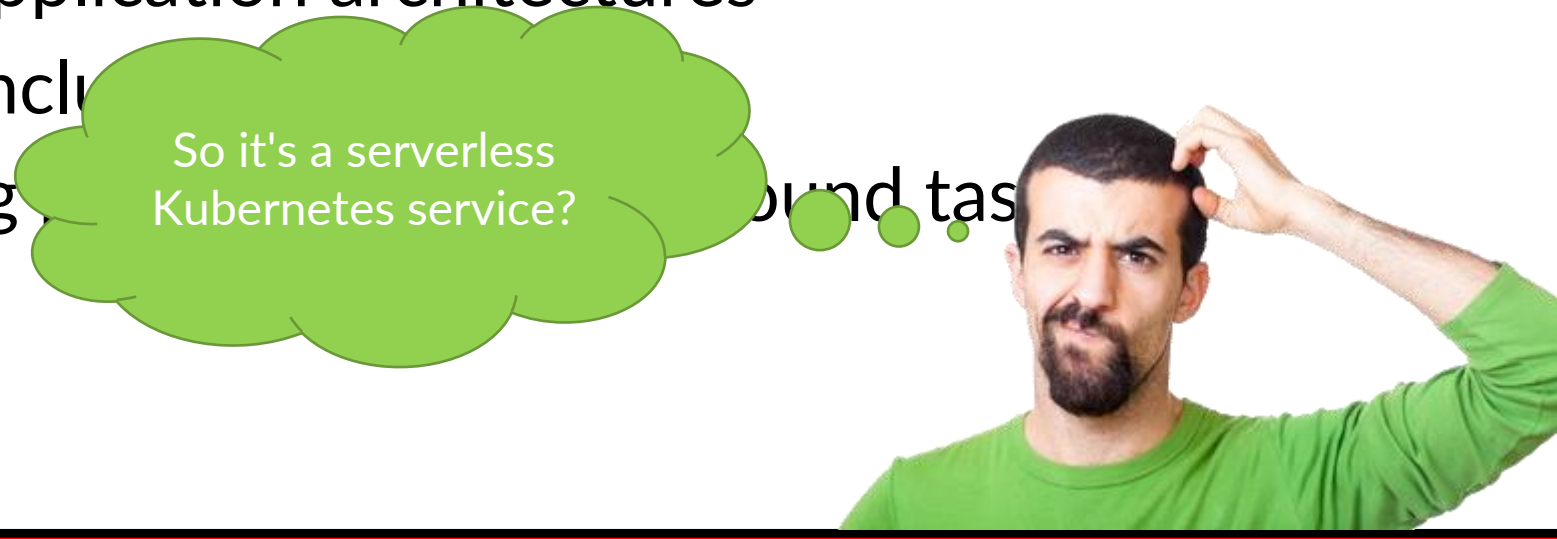
Container Instances - Workloads

Task automation	Agents
Small-scale batch processing	Bursting out

Azure Container Apps

Container Apps

- Serverless container platform powered by Kubernetes
  - General availability in June 2022
- Optimized for running general purpose containers
- Supports Kubernetes-style apps and microservices
- Enables event-driven application architectures
- Scale based on traffic incl
- Support of long running



Container Apps - Features

- HTTP, HTTPS, WebSocket, gRPC
- Visibility
  - External or internal only
- Auto scaling
  - Scaling to zero incur no charges
  - Supports Keda event-driven autoscaling
- Multi containers
  - While the multi container pod pattern is supported, the preferred method is to deploy containers individually
- Health Probes

Container Apps - Features

- Linux-based x86-64 (linux/amd64) container image only
- Revisions
  - Traffic split
- Secrets
- Darp integration
- Support for Managed Identities
- Easy Auth
- Publish revision using GitHub Actions

What workload?

Container Apps - Workloads

Web Apps	API Apps	Background Processes	Event-Driven Processes
• HTTP/S • Scaling by concurrent HTTP requests	• WebSocket, gRPC • Scaling by CPU or memory load	• Continuously running • Scaling by CPU or memory load	• Continuously running • Event-driven scaling • Scaling by Keda scalers

Azure Batch

Azure Batch

- Runs large-scale parallel and high-performance computing (HPC) batch jobs
  - Creates and manages a pool of compute nodes
  - Installs applications
  - Schedule jobs
- Tasks can be Docker-compatible containers

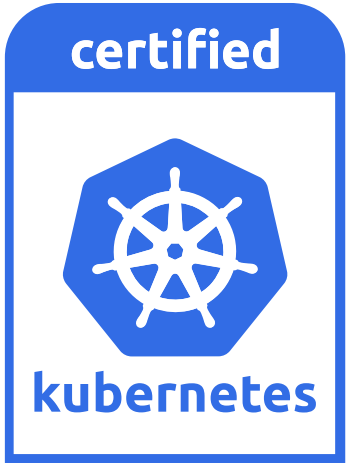
What workload?

Azure Batch - Workloads

Large-scale batch jobs

Azure Kubernetes Service

Azure Kubernetes Service



- Kubernetes as a service
- CNCF certified as Kubernetes conformant
- Not a hacked version
- Spin a cluster in a few minutes
- Azure takes care of the control-plane (master node)
- You pay for the nodes
  - Linux or Windows

Use what you want

	Development	DevOps	Monitoring	Networking	Storage	Security
Take advantage of services and tools in the Kubernetes ecosystem	HELM DRAFT	Jenkins Terraform BRIGADE JFrog CODESHIP HASHICORP	Prometheus fluentd Grafana OPENTRACING DATADOG JAEGER	CNI Networking TIGERA	MAPR portworx	Twistlock aqua heptio RBAC
OR, Leverage growing Azure support	VS Code	VSTS ARM	Azure Monitor	Azure VNET	Azure Storage	Azure Container Registry AAD Key Vault

What workload?

AKS - Workloads

Manage many containers	Leverage K8s skillset	Use K8s native tools
Active Directory integration	Integration with other Azure services	Corporate features & integration

RedHat OpenShift

RedHat OpenShift

- OpenShift is an Enterprise-ready Kubernetes container platform
- OpenShift extends Kubernetes
  - Provides added-value features to complement Kubernetes
- Managed OpenShift clusters
- Jointly engineered, operated, and supported by Red Hat and Microsoft

What workload?

OpenShift - Workloads

Complete Solution

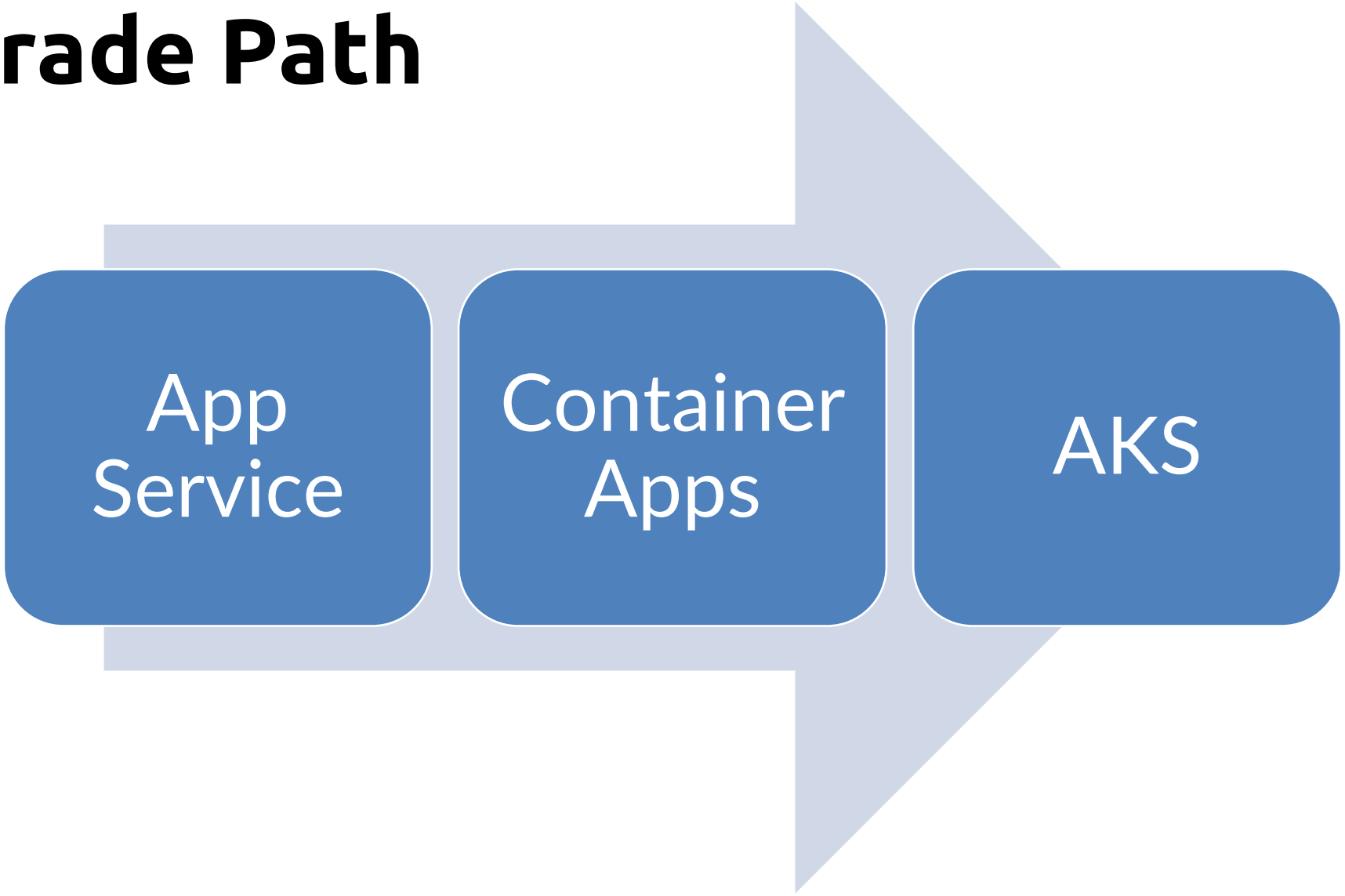
Support

Demo

Conclusion

The right tool for the right job

Upgrade Path



END OF LINE

Thank You!