Containerize Your Applications with Kubernetes on Azure
How do I ask a question?

If you have a technical or content-related question, please use the Q&A window.

We will address the questions as they come in.

Can I view this presentation after the webinar?

Yes, this presentation is being recorded.

A link to the recorded presentation will be sent to the email address you used to register.
Meet our speaker

Gabe Monroy
Lead PM,
Azure Cloud Native Compute
Containers and Kubernetes momentum

“By 2020, more than 50% of enterprises will run mission-critical, containerized cloud-native applications in production.”

Gartner

The average size of a container deployment has grown 75% in one year.¹

75%

Half of container environment is orchestrated.¹

77% of companies² who use container orchestrators choose Kubernetes.

Larger companies are leading the adoption.¹

Nearly 50% of organizations¹ running 1000 or more hosts have adopted containers.

¹: Datadog report: 8 Surprising Facts About Real Docker Adoption
2: CNCF survey: cloud-native-technologies-scaling-production-applications
Kubernetes: the industry leading orchestrator

- **Portable**
  - Public, private, hybrid, multi-cloud

- **Extensible**
  - Modular, pluggable, hookable, composable

- **Self-healing**
  - Auto-placement, auto-restart, auto-replication, auto-scaling
How Managed Kubernetes on Azure works

- Automated upgrades, patches
- High reliability and availability
- Easy and secure cluster scaling
- Self-healing
- API server monitoring
- Control plane at no charge
**From infrastructure to innovation**

**Managed Kubernetes empowers you to do more**

Focus on your containers and code, not the plumbing of them.

<table>
<thead>
<tr>
<th>Responsibilities</th>
<th>DIY with Kubernetes</th>
<th>Managed Kubernetes on Azure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Containerization</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>Application iteration, debugging</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>CI/CD</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>Cluster hosting</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>Cluster upgrade</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>Patching</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>Scaling</td>
<td></td>
<td>Blue</td>
</tr>
<tr>
<td>Monitoring and logging</td>
<td></td>
<td>Blue</td>
</tr>
</tbody>
</table>

Customer: Microsoft
Azure Kubernetes Service Overview

- **Introduction**
- **Top scenarios**
- **Customer stories**
- **Product demo**
- **Open source culture**
- **Resources**

Azure Kubernetes Service Overview
Azure Kubernetes Service (AKS)

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 momentum

Kubernetes on Azure usage grew 10x
Kubernetes on Azure customers grew 5x

...over the last 12 months
### Azure makes Kubernetes easy

**Deploy and manage Kubernetes with ease**

<table>
<thead>
<tr>
<th>Task</th>
<th>The Old Way</th>
<th>With Azure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a cluster</td>
<td>Provision network and VMs</td>
<td><code>az aks create</code></td>
</tr>
<tr>
<td></td>
<td>Install dozens of system components including etcd</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Create and install certificates</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Register agent nodes with control plane</td>
<td></td>
</tr>
<tr>
<td>Upgrade a cluster</td>
<td>Upgrade your master nodes</td>
<td><code>az aks upgrade</code></td>
</tr>
<tr>
<td></td>
<td>Cordon/drain and upgrade worker nodes individually</td>
<td></td>
</tr>
<tr>
<td>Scale a cluster</td>
<td>Provision new VMs</td>
<td><code>az aks scale</code></td>
</tr>
<tr>
<td></td>
<td>Install system components</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Register nodes with API server</td>
<td></td>
</tr>
</tbody>
</table>
## Azure makes Kubernetes easy

### Accelerate containerized application development

<table>
<thead>
<tr>
<th>Task</th>
<th>The Old Way</th>
<th>With Azure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build a containerized app and deploy to Kubernetes</td>
<td>Build the app resource</td>
<td><code>draft init</code> to configure your environment</td>
</tr>
<tr>
<td></td>
<td>Define a Dockerfile/Helm chart</td>
<td><code>draft create</code> to auto-create Dockerfile/Helm chart</td>
</tr>
<tr>
<td></td>
<td>Build the container image</td>
<td><code>draft up</code> to deploy to Kubernetes</td>
</tr>
<tr>
<td></td>
<td>Push the container to a registry</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Write Kubernetes manifests/Helm chart</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deploy to Kubernetes</td>
<td></td>
</tr>
<tr>
<td>Build and test individual services in a microservices architecture</td>
<td>Set up a local dev environment using Minikube</td>
<td>Use DevSpaces to iterate, test and debug</td>
</tr>
<tr>
<td></td>
<td>Determine the transitive closure of dependencies</td>
<td>Do breakpoint debugging in your IDE</td>
</tr>
<tr>
<td></td>
<td>Identify behavior of dependencies for key test cases</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Stub out dependent services with expected behavior</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Make local changes, check-in, and hope things work</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Validate with application logs</td>
<td></td>
</tr>
<tr>
<td>Expose web apps to the internet with a DNS entry</td>
<td>Deploy an ingress controller</td>
<td>Turn HTTP application routing on in your cluster</td>
</tr>
<tr>
<td></td>
<td>Create a load-balanced IP for it</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Add an ingress resource to your deployment</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Acquire a custom domain</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Create a DNS A-record for your service</td>
<td></td>
</tr>
</tbody>
</table>
### Azure makes Kubernetes easy

**Set up CI/CD in a few clicks**

<table>
<thead>
<tr>
<th>Task</th>
<th>The Old Way</th>
<th>With Azure</th>
</tr>
</thead>
</table>
| Set up a CI/CD pipeline and deploy to Kubernetes | Create git repo  
Create a build pipeline  
Create a container registry  
Create a Kubernetes cluster  
Configure build pipeline to push to container registry  
Configure build pipeline to deploy to Kubernetes | Create an Azure DevOps project with AKS as a target                                    |
| Make container images available for deployment worldwide | Create a container registry in every region  
Configure build pipeline with multiple endpoints  
Loop through all regions and push following build | Create an Azure Container Registry with geo-replication  
Push your image to a single endpoint                                                     |
| Track health with consolidated cluster and application logs | Choose a logging solution  
Deploy log stack in your cluster or provision a service  
Configure and deploy a logging agent onto all nodes | Checkbox “container monitoring” in the Azure portal                                      |
Work how you want with opensource tools and APIs

<table>
<thead>
<tr>
<th>Development</th>
<th>DevOps</th>
<th>Monitoring</th>
<th>Networking</th>
<th>Storage</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>HELM</td>
<td>Jenkins</td>
<td>Prometheus</td>
<td>CNI</td>
<td>MAPR</td>
<td>Twistlock</td>
</tr>
<tr>
<td>Draft</td>
<td>Terraform</td>
<td>fluentd</td>
<td>TIGERA</td>
<td>portworx</td>
<td>heptio</td>
</tr>
<tr>
<td>CODESHIP</td>
<td>BRIGADE</td>
<td>Grafana</td>
<td></td>
<td></td>
<td>HASHICORP</td>
</tr>
<tr>
<td>JFrog</td>
<td>OPENTRACING</td>
<td>BASILISK</td>
<td>JAEGETER</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Take advantage of services and tools in the Kubernetes ecosystem

OR,
Leverage growing Azure support

- VS Code
- VSTS
- ARM
- Azure Monitor
- Azure VNET
- Azure Storage
- Azure Container Registry
- AAD
- Key Vault
Work how you want with open source tools and APIs

Easily connect to SLA-backed Azure services with OSBA
Secure your Kubernetes environment

- Control access through AAD and RBAC
- Safeguard keys and secrets with Key Vault
- Secure network communications with VNET and CNI
- Compliant Kubernetes service with certifications covering SOC, HIPAA, and PCI
Scale and run applications with confidence

- Built-in auto scaling
- Global data center to boost performance and reach
- Elastically burst from AKS cluster using ACI
- Geo-replicated container registry
Top scenarios

- Introduction
- Azure Kubernetes Service Overview
- Top scenarios
- Product demo
- Customer stories
- Open source culture
- Resources
Top scenarios for Kubernetes on Azure

- **Lift and shift to containers**
  - Cost saving
  - without refactoring your app

- **Microservices**
  - Agility
  - Faster application development

- **Machine learning**
  - Performance
  - Low latency processing

- **IoT**
  - Portability
  - Build once, run anywhere
App Modernization without code changes

- Move applications as is to Azure, but with cost savings
- Containers support all frameworks and technology stacks
- Faster app deployment with DevOps tools
Microservices: for faster app development

- Independent deployments
- Improved scale and resource utilization per service
- Smaller, focused teams
Data Scientist in a box

- Quick deployment and high availability
- Low latency data processing
- Consistent environment across test, control and production
Scalable Internet of Things solutions

✓ Portable code, runs anywhere
✓ Elastic scalability and manageability
✓ Quick deployment and high availability
Product deep dive

- Introduction
- Azure Kubernetes Service Overview
- Top scenarios
- Product demo
- Customer stories
- Open source culture
- Resources
Customer stories

- Introduction
- Top scenarios
- Customer stories
- Resources
  - Azure Kubernetes Service Overview
  - Product demo
  - Open source culture
Siemens Health leverages technology to connect medical devices to the cloud through AKS

**Challenge:** Siemens needed to speed up their development process to make the transition from value-added services provider to platform provider.

**Solution:** Siemens adopted Azure Kubernetes Service (AKS) to speed up application development and run their microservices-based apps.

**Outcome:** With AKS, Siemens has driven newfound product development agility. AKS enables them to use an applicant gateway and API management to manage exposure, control, and to meter the access continuously.

> The managed Azure Kubernetes Service puts us really into a position to not only deploy our business logic in Docker containers, including the orchestration, but it's also really easy through application gateway and API management to manage that exposure and control and meter the access continuously.

---

Thomas Gossler, Lead Architect
Digital Ecosystem Platform, Siemens
Energy company electrifies pace of innovation and expansion

**Challenge:** To meet aggressive growth goals, Ambit Energy needed to automate infrastructure provisioning to match their pace of new software creation.

**Solution:** To stand up infrastructure quickly, Ambit used Microsoft Azure services such as Azure Container Service, together with infrastructure as code and open source technologies, to completely automate infrastructure provisioning.

**Outcome:** By implementing Azure, Ambit can move dramatically faster to enhance its services and enter new markets. Infrastructure redundancy is flexible and worry-free. And costs are 22 percent lower, which helps Ambit compete in the crowded electricity market.

*"Azure support for Docker, Kubernetes, Puppet, Terraform, Cassandra, and other open source tools has become very important to us and has really accelerated our move into Azure.*

Robert Rudduck, Director of Architecture and DevOps
Ambit Energy
Altair Engineering democratizes HPC access using the cloud

**Challenge:** Altair needed a specialized HPC architecture containing high-performance graphics processing units to deliver their latest topology optimization and analysis application to customers.

**Solution:** Altair used Kubernetes in Azure Container Service to handle back-end functions and increase the density of services running across compute nodes.

**Outcome:** With Azure, Altair provides customers with a scalable, cost-effective back-end HPC infrastructure, eliminating the need for expensive engineering workstations.

*Customers are limited as to what they can do on workstations, but with Azure we can give them a scalable, cost-effective back-end HPC infrastructure.*

Sam Mahalingam, Chief Technical Officer Cloud Computing and High-Performance Computing Strategy
Altair Engineering
Cancer treatment company streamlines IT, focuses on innovation using container software technology

**Challenge:** Varian needed to provide broader cancer care and enable faster innovation for the benefit of cancer patients.

**Solution:** Varian chose Microsoft Azure as its cloud platform and Azure Kubernetes Service to scale application deployments to thousands of customers, utilizing containers to modernize existing apps and create new ones.

**Outcome:** With AKS, Varian’s developers can deliver features to customers quickly and get their feedback without the overhead of provisioning a group of virtual machines.

"With AKS, developers get a safe place to innovate and to experiment with new technologies and ideas.... It’s the best of open service combined with the best of Azure."

Shivakumar Gopalakrishnan, Senior Manager
Varian Medical Systems
Tech startup creates a “data scientist in a box” with machine learning and Microsoft Azure

**Challenge:** Falkonry needed a solution to scale the deployment of its machine learning application to reach customers in the oil and gas industries.

**Solution:** Falkonry used Azure Kubernetes Service to automate the deployment of Kubernetes clusters to deliver their application globally.

**Outcome:** With Azure Kubernetes Service, Falkonry is able to deploy their solutions in days, compared to months it takes for companies using a more traditional platform approach.

“We’re very happy with the speed of deployment we can offer our customers with Azure. If we had to fly people out to configure and set up hardware and software, we would lose several weeks in the process.”

Sanket Amberkar, Senior Vice President of Marketing
Falkonry
Open source culture

Introduction

Azure Kubernetes Service Overview

Top scenarios

Product demo

Customer stories

Resources

Open source culture
Community culture

- Open source container code contributions
- Numerous open source project builds
- Open source community leadership
- Ongoing partner and customer growth
Azure + Open Source Momentum

Azure is a strong platform for Open Source

Linux VMs are growing at ~2 times Windows VMs today

Microsoft announced GitHub acquisition

1 in 3 VMs on Azure are Linux

~60% of 3rd party Azure Marketplace images are open source

Partnerships

“Microsoft Joins Cloud Native Computing Foundation as Platinum Member”
Microsoft leads open source communities

Two members of the Kubernetes steering committee

Member of the technical board of the Cloud Native Compute Foundation

Board member of the Linux Foundation

Several leads or co-leads of Kubernetes SIGs (special interest groups)
Microsoft contributes open source containers

#2 overall individual contributor to Kubernetes (Brendan Burns)

#4 overall individual contributor to Docker (John Howard)

#1-3 overall individual contributors to Helm

70 Microsoft employees have made contributions to Kubernetes
Microsoft builds open source projects

**Helm** – The de-facto package manager for Kubernetes ([https://helm.sh](https://helm.sh)), Top level CNCF project

**Draft** – A rapid-development environment for new Kubernetes developers ([https://draft.sh](https://draft.sh))

**Brigade** – Easy to use Javascript based workflow definition for Kubernetes ([https://brigade.sh](https://brigade.sh))

Check out resources

Azure Kubernetes Service (AKS)

Containers on Azure pitch deck
https://aka.ms/containerstdmdeck

Smart Hotel 360 Demo
https://aka.ms/containerstdmdeck

Documentation resources
https://docs.microsoft.com/en-us/azure/aks/

Ebook for distributed systems
https://azure.microsoft.com/en-us/resources/designing-distributed-systems/

Distributed system HoL
https://github.com/brendandburns/designing-distributed-systems-labs

AKS HoL
https://aka.ms/aks360hol
Core team
PM: Gabe Monroy, @gabrtv
PM: Sean McKenna
PM: Jason Hansen
PMM: Stella Lin
CDA: Bryan Liston

Community
Brendan Burns, @brendandburns
Michelle Noorali

Partner team
Morgan Pettis
Leon Jones
Dan Sandlin

Connect with us
Thank you for joining us.