Deploying to Azure Kubernetes Service (AKS)

Craig Golightly
SENIOR SOFTWARE CONSULTANT

@seethatgo www.seethatgo.com

Overview

Open an Azure account

Deploy and update demo app on AKS

- Step-by-step
- Commands

Azure Portal

- Monitoring
- Billing

Delete cluster

Opening a Microsoft Azure Account

azure.microsoft.com

Microsoft account (Skype, GitHub)

Name and address

Credit card information

Initialize Tooling

Install Azure CLI

Python 3

gcc

Register namespaces

Create resource group

```
az login
az provider register --namespace Microsoft.Network
az provider register --namespace Microsoft.Compute
az provider register --namespace Microsoft.OperationsManagement
az account list locations
az group create --name kube-demo --location westus
```

Initialize Tooling

Login

Namespace registration

Choose a location

Create resource group

```
az acr create
--resource-group
--location
--name
--sku
```

```
az acr create -g kube-demo -l uswest -n mydemoregistry --sku Basic
```

Create Registry

Resource group name

Location

Registry name

Sku

```
<registry-name>.azurecr.io/<namespace>/<image-name>:<tag>
docker build -t demoregistry.azurecr.io/examples/demo:1.0 .
docker tag 8e2036e25863 demoregistry.azurecr.io/examples/demo:1.0
az acr login --name demoregistry
docker push demoregistry.azurecr.io/examples/demo:1.0
```

<registry-name> - name of your registry to push to

<namespace> - optional to organize repository

<image-name> - name of the image

<tag> - tag for the image

```
az aks create

--resource-group

--name

--node-vm-size

az aks create -g kube-demo -n demo-cluster --node-vm-size Standard_D1
```

Create Cluster

Resource group name

Cluster name

What size of VM to use for nodes

```
CLIENT_ID=$(az aks show ...)

ACR_ID=$(az acr show ...)

az role assignment create \
--assignee $CLIENT_ID \
--role acrpull \
--scope $ACR_ID
```

- Get the id for your cluster
- Get the id for your registry
- Allow cluster pull access from registry

```
az aks get-credentials \
–g kube-demo –n demo-cluster
```

- Get credentials for kubectl to connect to your cluster

```
az aks enable-addons -a monitoring \
-g kube-demo -n demo-cluster
```

- Enable monitoring for your cluster

```
kubectl create deployment demo-app \
--image=demoregistry.azurecr.io/examples/demo:1.0

kubectl expose deployment demo-app \
--type=LoadBalancer --port 5000 --target-port 5000
```

Create Deployment and Service

kubectl to create and expose deployment

```
kubectl scale deployment demo-app --replicas=3

az aks scale -g kube-demo -n demo-cluster -c 5
```

Scale Pods and Nodes

kubectl to scale pods

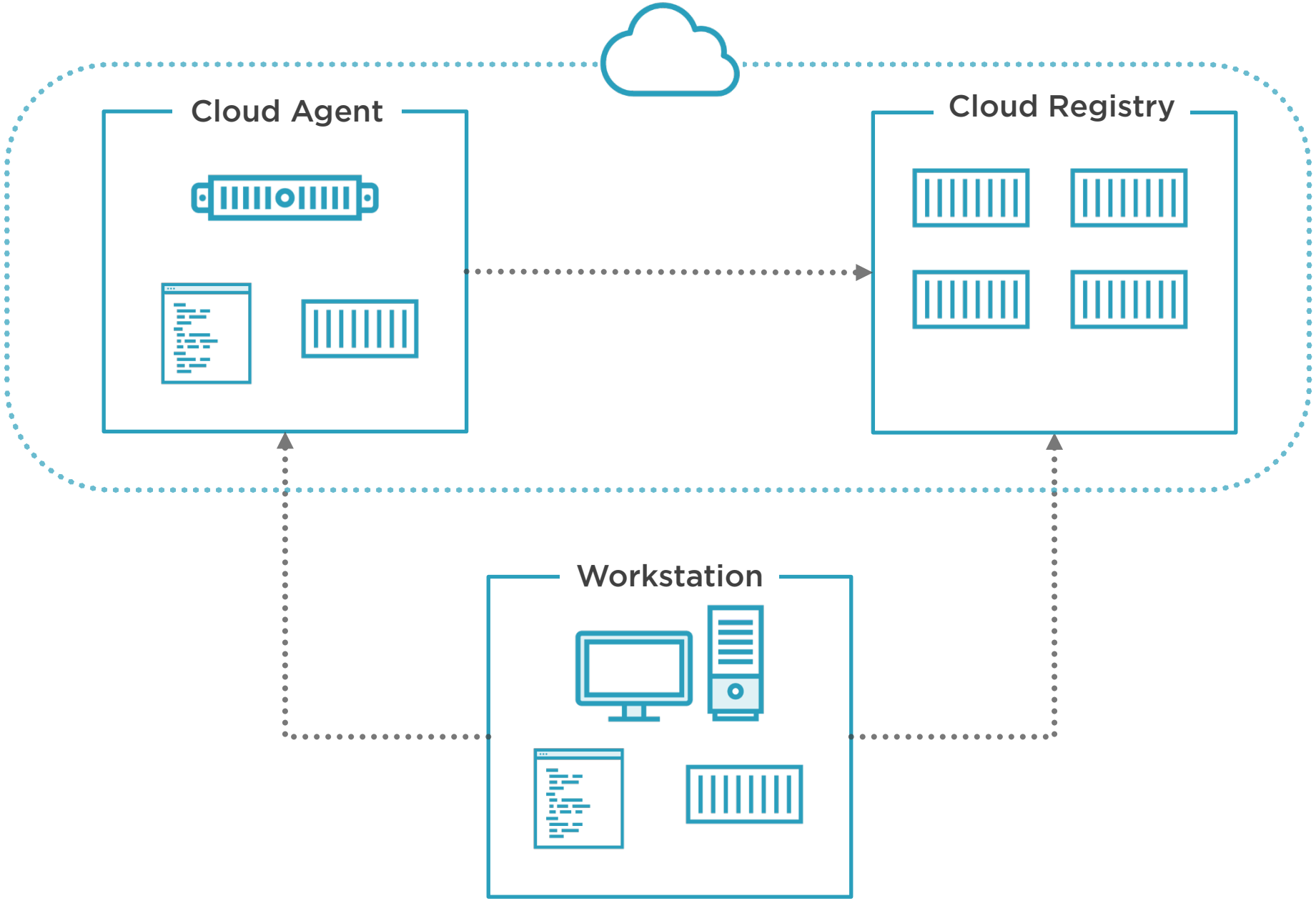
az to scale nodes

```
docker build -t demoregistry.azurecr.io/examples/demo:2.0 .

docker push demoregistry.azurecr.io/examples/demo:2.0

kubectl set image deployment/demo-app \
demo=demoregistry.azurecr.io/examples/demo:2.0
```

Update Application

docker build, tag, and push new image

kubectl set image


Cloud Agent
Cloud Registry
Workstation

```
az acr build
--image
--registry
--file
az acr build -t examples/demo:2.0 -r demoregisty -f dockerfile .
```

Cloud Agent Build

Namespace, name and tag of image

Name of registry to push image

Name of dockerfile

Directory with code for image

Azure Portal

- Monitoring
- Cluster
- Registry
- Billing

```
kubectl delete service demo-app

az aks delete -g kube-demo -n demo-cluster

az acr repository delete -n demoregistry --image examples/demo:2.0

az group delete -n kube-demo
```

Cleanup

Delete service – kubectl

Delete cluster – az

Delete images – az

Summary

Use AKS service

Entire application lifecycle

- Create
- Scale
- Update
- Delete

Try it with your app

Check out all 3 cloud options