

Versioniert, automatisiert und sicher - Optimale Bereitstellung von Azure- Ressourcen

Marc Müller
Principal Consultant



marc.mueller@4tecture.ch
@muellermarc
www.4tecture.ch

4tecture®
empower your software solutions



About me:

Marc Müller
Principal Consultant
@muellermarc



4tecture[®]
empower your software solutions

Our Products:

Multi-Tenant OpenID Connect Identity Provider



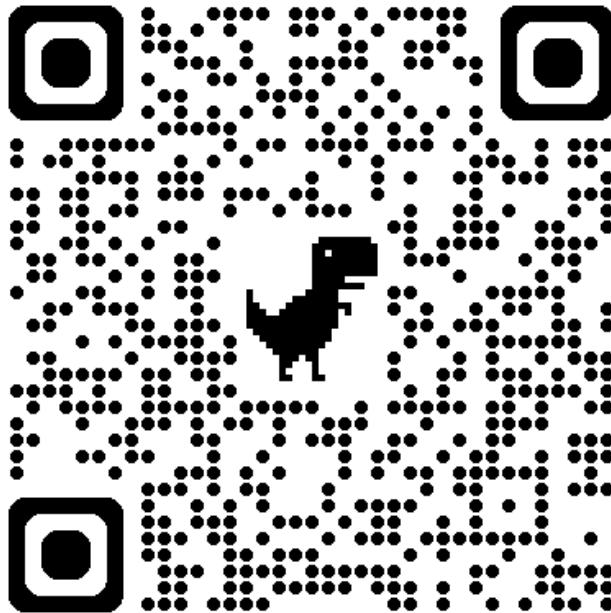
www.proauth.net

Enterprise Application Framework for .NET



www.reafx.net

Slide Download



<https://www.4tecture.ch/events/ddc24azure>

Agenda

- Intro
- Security / Authentication
- Infrastructure as Code / Configuration as Code
- Traditional CI/CD vs GitOps
- Dynamic Resource Deployment
- Conclusion





Azure Deployments

Intro

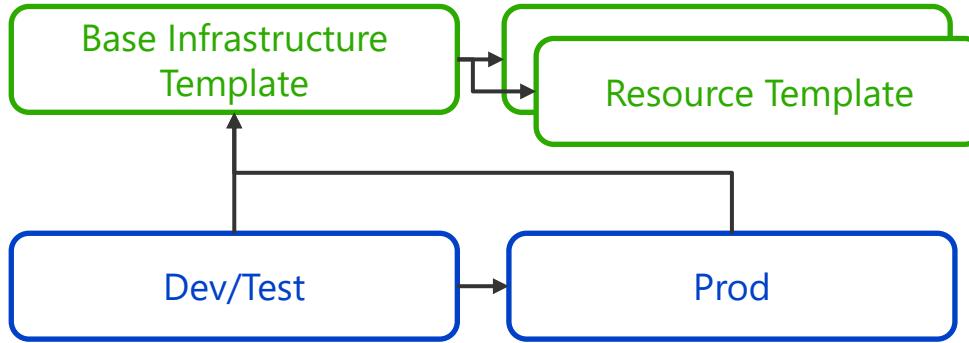
4tecture®
empower your software solutions

Challenges

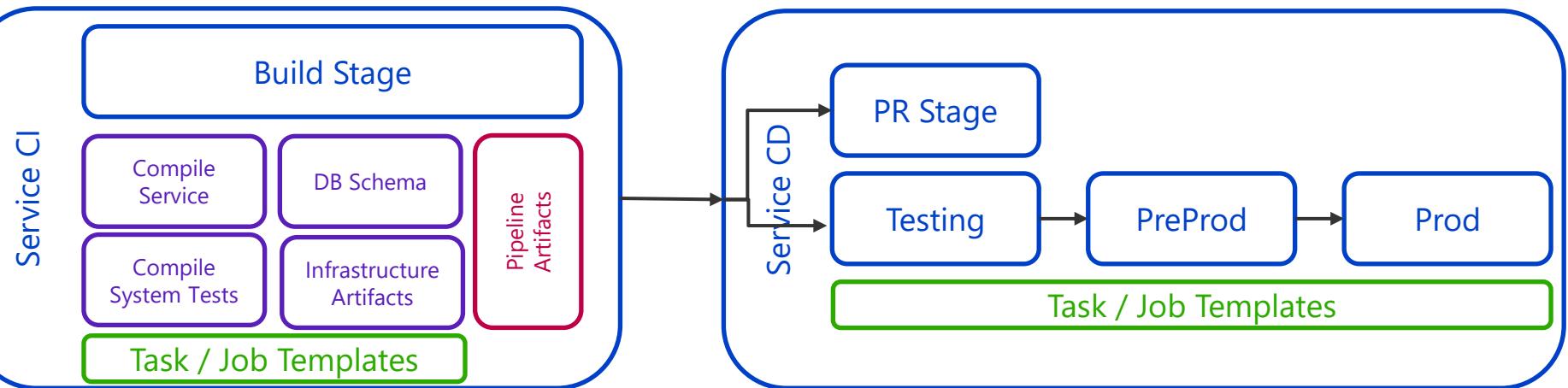
- Versioned – everything is code
- No manual interaction
- Reusability
- Least Privileged
- Network Access

CI/CD Best Practices

Platform CD



Service CI



A close-up, low-angle shot of several rowers in a racing shell. Their legs and torsos are visible as they pull on the oars. The oars have yellow handles and black blades. The water is choppy, creating white spray around the oars.

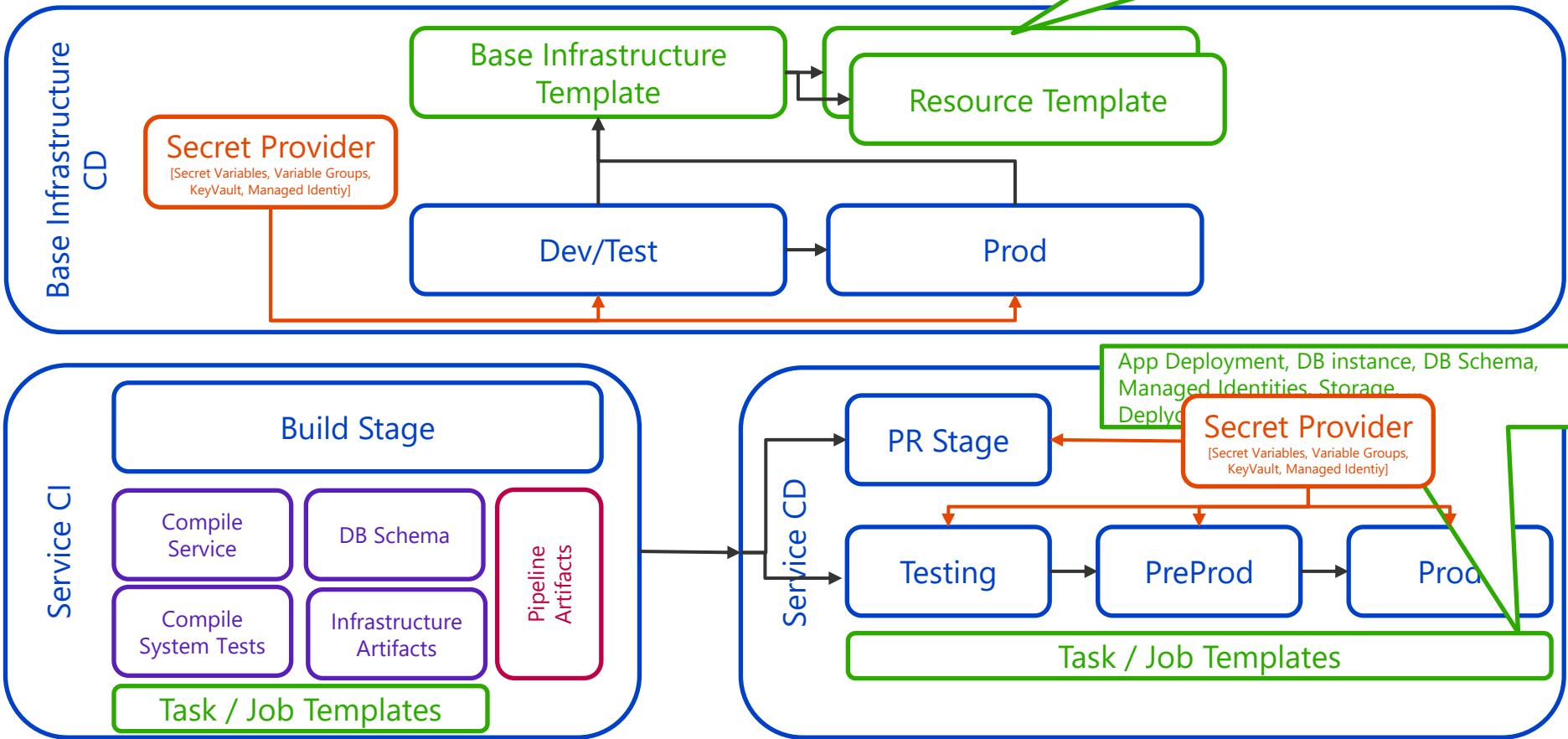
Azure Deployments

Security / Authentication

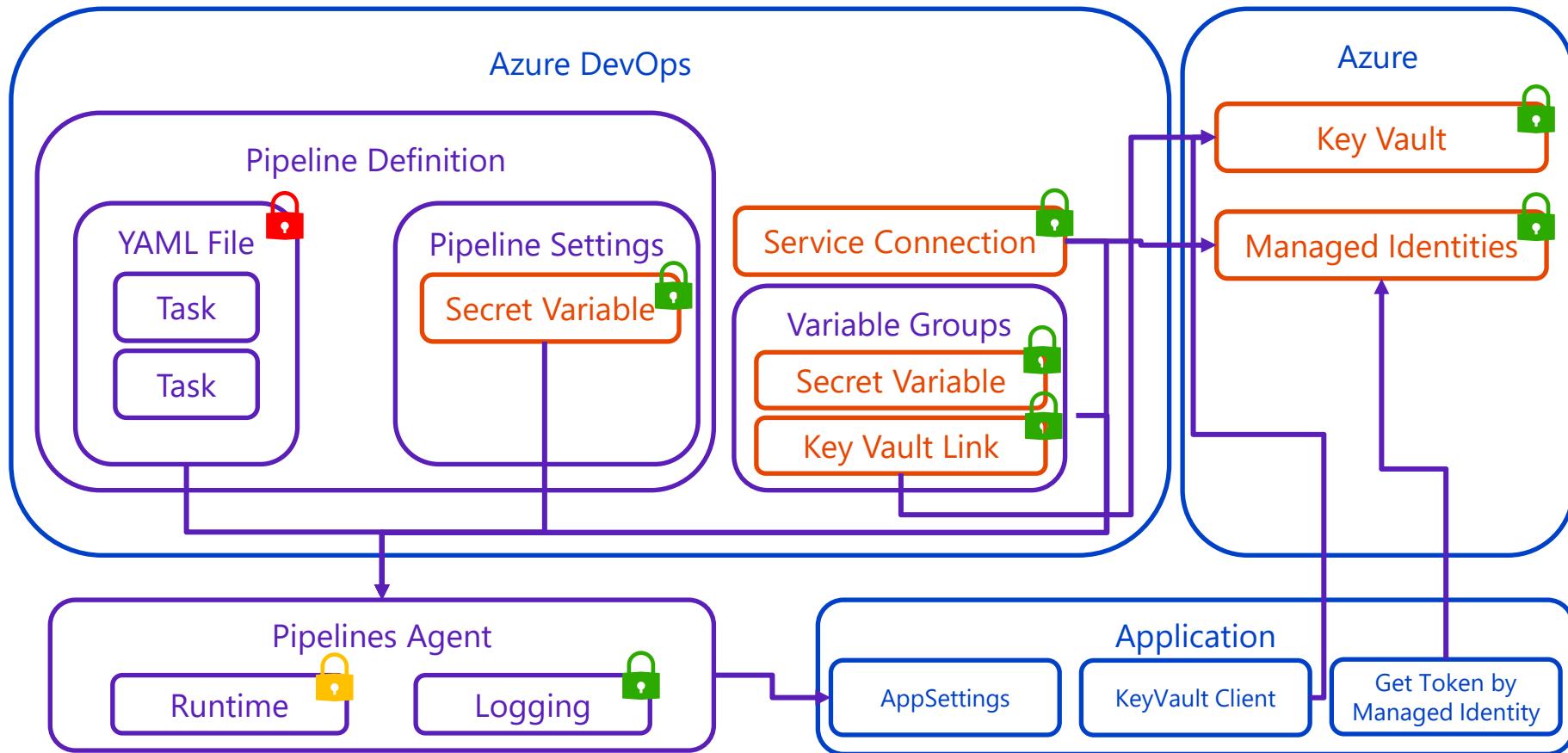
Secrets Challenges

- Configuration as Code
 - Pipeline Definition and Variables are in Source Control
 - Public / Search Index
 - Validate Source Code to not contain any secrets (PR/CI)
 - Just reference secrets in your code / pipelines
- Secret Usage vs. Secret Management
 - Pipeline editors don't see values in secret variables
 - Pipeline editors only have rights to reference secrets / service connection
 - Additional authorization process for service connections
 - Management of secrets belongs only to dedicated administrators

Best Practices



Simplified Overview



DEMO

Variable Groups / Secret Files



4ecture-demo / YAMLPipelinesDemo / Pipelines / Library

Search

☰ ☰ ? 🔎

Library > DemoVariableGroup

Variable group | Save | Clone | Security | Pipeline permissions | Approvals and checks | Help

Properties

Variable group name

DemoVariableGroup

Description

Link secrets from an Azure key vault as variables ⓘ

Variables

Name ↑	Value	🔒
GroupSecret1	*****	
GroupValue1	Value 01	
GroupValue2	Value 02	↗

+ Add

Demo Recap



Library > DemoKeyVaultVariableGroup

Variable group



Properties

Variable group name

DemoKeyVaultVariableGroup

Description

 Link secrets from an Azure key vault as variables ⓘ

Azure subscription * | Manage ↗

Microsoft Azure Sponsorship)



Scoped to subscription 'Microsoft Azure Sponsorship'

Key vault name * | Manage ↗

demovaultmizr6qqogvzw



Variables

Last refreshed: 15.02.2021

Delete	Secret name	Content type	Status	Expiration date
	theSecret		Enabled	Never

+ Add



← 02_SecretVariableGroup

Variables

Run



Show assistant



```
8° main ▾  ♦ SecretsDemo / Pipelines/02_SecretVariableGroup.yml

1 trigger:
2   - main
3
4 pool:
5   - vmImage: ubuntu-latest
6
7 variables:
8   - name: SampleVariable01
9     value: "Some sample content"
10  - group: DemoVariableGroup
11  - group: DemoKeyVaultVariableGroup
12
13 steps:
14   - pwsh: |
15     Write-Host "SampleVariable01: $(SampleVariable01)"
16     displayName: "Show a plain text variable"
17
18   - pwsh: |
19     Write-Host "GroupValue1: $(GroupValue1)"
20     displayName: "Show a plain text variable from group"
21
22   - pwsh: |
23     Write-Host "GroupSecret1: $(GroupSecret1)"
24     displayName: "Show a secret variable from group"
25
26   - pwsh: |
27     Write-Host "theSecret: $(theSecret)"
28     displayName: "Show a secret variable from Key Vault by group"
29
```





Upload a secure file

Upload file

Drag and drop a file here or click browse to select a file.

OK

Cancel

Demo
Recap

DEMO

Service Connection





Project Settings

YamlPipelinesDemo

General

[Overview](#)[Teams](#)[Permissions](#)[Notifications](#)[Service hooks](#)[Dashboards](#)

Boards

[Project configuration](#)[Team configuration](#)[GitHub connections](#)

Pipelines

[Agent pools](#)[Parallel jobs](#)[Settings](#)[Test management](#)[Release retention](#)[Service connections](#)[XAML build services](#)

Repos

[Repositories](#)

Service connections

[Filter by keywords](#)[Microsoft Azure Sponsorship](#)

New service connection

Choose a service or connection type

 Search connection types Azure Classic Azure Repos/Team Foundation Server Azure Resource Manager Azure Service Bus Bitbucket Cloud Chef Docker Host Docker Registry Generic GitHub GitHub Enterprise Server Incoming WebHook Jenkins Jira Kubernetes

Demo Recap

Project Settings

YamlPipelinesDemo

General

Overview

Teams

Permissions

Notifications

Service hooks

Dashboards

Boards

Project configuration

Team configuration

Github connections

Pipelines

Agent pools

Parallel jobs

Settings

Test management

Release retention

Service connections

XAML build services

Repos

Repositories

Artifacts

Storage

Service connections

Convert your existing Azure Resource Manager service connections which use secrets to authenticate to leverage Workload identity federation maintenance.

Filter by keywords

Microsoft Azure Sponsorship

New Azure service connection

Azure Resource Manager

Authentication method

-  Workload Identity federation (automatic) Recommended
-  Workload Identity federation (manual)
-  Service principal (automatic)
-  Service principal (manual)
-  Managed identity
-  Publish Profile

[Need help choosing a connection type?](#)

Back

Next



Project Settings
YamlPipelinesDemo

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards

- Project configuration
- Team configuration
- GitHub connections

Pipelines

- Agent pools
- Parallel jobs
- Settings
- Test management
- Release retention
- Service connections
- XAML build services

Repos

- Repositories

Artifacts

- Storage

← **Security**
Microsoft Azure Sponsorship (8) [YamlPipelinesDemo] (2)

User permissions

Project Organization

+ Add Undo Save Inheritance ▾

User	Role	Access
[YamlPipelinesDemo]\Endpoint Administrators	Administrator ▾	Inherited
Marc Müller	Administrator ▾	Assigned

Pipeline permissions

The following YAML pipelines are allowed to use this resource. YAML pipelines from other projects are not shown in this list. All Classic pipelines can use this resource.

Pipeline
04_HelloWorld_MultiStage Pipeline
05_HelloWorld_Trigger_CD Pipeline
03_KeyVaultDemo.yml Pipeline
02_SecretVariableGroup Pipeline

Project permissions

The projects below have access to this service connection.

Demo
Recap

4tecture-demo / YamIPipelinesDemo / Settings / Service connections

Search

New service connection

Project Settings

YamlPipelinesDemo

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards

- Project configuration
- Team configuration
- Github connections

Pipelines

- Agent pools
- Parallel jobs
- Settings
- Test management
- Release retention
- Service connections
- XAML build services

Repos

- Repositories

Service connections

Filter by keywords

Created by

Microsoft Azure Sponsorship ()



← 03_KeyVaultDemo.yml

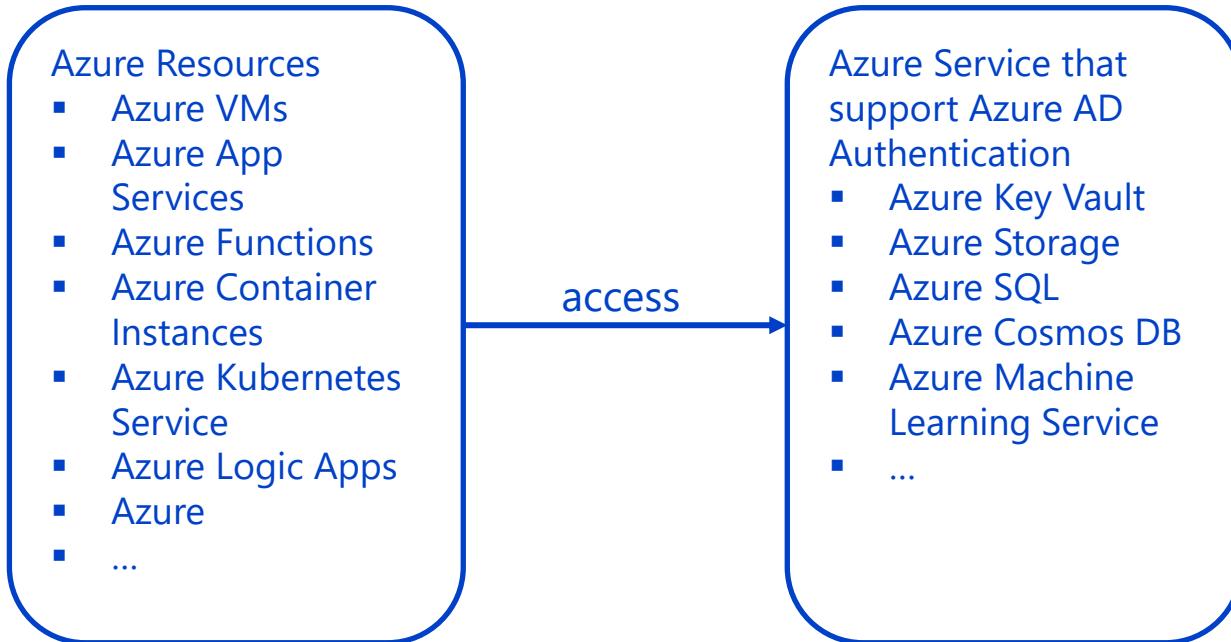
```
1 trigger:
2 - main
3
4 pool:
5 - vmImage: ubuntu-latest
6
7 variables:
8 - AzureConnection: 'Microsoft Azure Sponsorship ('
9   buildConfiguration: 'Release'
10  webArtifactName: 'SecretsWebApp'
11  infrastructureArtifactName: 'Infrastructure'
12 stages:
13 > - stage: build...
14
15 - stage: deploy
16   displayName: 'Deploy the App'
17   jobs:
18     - deployment: deployAzure
19       displayName: 'Deploy Infrastructure and App'
20       environment: secretdemoadzure
21       strategy:
22         runOnce:
23           deploy:
24             steps:
25               - download: current
26                 artifact: '$(infrastructureArtifactName)'
27                 displayName: 'Download pipeline artifacts'
28               - download: current
29                 artifact: '$(webArtifactName)'
30                 displayName: 'Download pipeline artifacts'
31
32 Settings
33   - task: AzureResourceManagerTemplateDeployment@3
34     inputs:
35       deploymentScope: 'Resource Group'
36       azureResourceManagerConnection: $(AzureConnection)
37       action: 'Create Or Update Resource Group'
38       resourceGroupName: 'SecretsDemo'
```



Managed Identities

- **Benefits**
 - No credentials accessible to the user
 - No management of credentials
 - Used to authenticate to any Azure service that supports Azure AD
 - No additional costs
- **Types**
 - System-assigned Managed Identities
 - Managed identity directly on the service instance
 - Is tied to the lifecycle of that service instance
 - User-assigned Managed Identities
 - Created as a standalone Azure resource

Use Managed Identities when...



Azure DevOps 4ecture-demo / YamlPipelinesDemo / Repos / Files / SecretsDemo

Search

YamlPipelinesDemo + SecretsDemo

Overview Boards Repos Files Commits Pushes Branches Tags Pull requests Advanced Security Pipelines Test Plans Artifacts

Program.cs

Contents History Compare Blame

```
1 using Azure.Extensions.AspNetCore.Configuration.Secrets;
2 using Azure.Identity;
3 using Azure.Security.KeyVault.Secrets;
4 using Microsoft.AspNetCore.Hosting;
5 using Microsoft.Extensions.Configuration;
6 using Microsoft.Extensions.Hosting;
7 using System;
8
9 namespace HelloWorldSecrets
10 {
11     public class Program
12     {
13         public static void Main(string[] args)
14         {
15             CreateHostBuilder(args).Build().Run();
16         }
17
18         public static IHostBuilder CreateHostBuilder(string[] args) =>
19             Host.CreateDefaultBuilder(args)
20                 .ConfigureAppConfiguration((context, config) =>
21                 {
22                     var builtConfig = context.GetBuiltInConfiguration();
23                     var secretClient = new SecretClient(new Uri($"https://{{builtConfig["KeyVaultName"]}}.vault.azure.net/"),
24                         new DefaultAzureCredential());
25                     config.AddAzureKeyVault(secretClient,
26                         new AzureKeyVaultConfigurationOptions()
27                         {
28                             ReloadInterval = TimeSpan.FromMinutes(1),
29                             Manager = new KeyVaultSecretManager()
30                         });
31                 })
32                 .ConfigureWebHostDefaults(webBuilder =>
33                 {
34                     webBuilder.UseStartup<Startup>();
35                 });
36             }
37         }
38     }
```

C# Program.cs

C# Startup.cs

HelloWorldSecrets.sln

Infrastructure

DEMO

Managed Identity



demovaultmizr6qqogvzzw | Access policies

Key vault

 Search[+ Create](#)

Refresh



Delete



Edit

⚠️ Quickly protect your Key vault from accidental deletion by turning on soft-delete. Please enable soft-delete in 'Properties' page. Click here to learn more. →

Access policies enable you to have fine grained control over access to vault items. [Learn more](#)

 Search

Permissions : All

Type : All

Showing 1 to 5 of 5 records.

Name	Email	Key Permissions	Secret Permissions	Certificate Permissions
APPLICATION				
<input type="checkbox"/>			Get, List	
USER				
<input type="checkbox"/> secretsdemo-webapp			Get, List	
<input type="checkbox"/> I			All	
<input type="checkbox"/> I			Get, List	
<input type="checkbox"/> I			All	

Events

Objects



Keys



Secrets



Certificates

Settings

Access configuration



Microsoft Defender for Cloud



Properties



Locks

> Monitoring

> Automation

> Help

< Previous

Page

1

▼

of 1

Next >



Azure DevOps 4tecture-demo / YamlPipelinesDemo / Repos / Files / SecretsDemo

Search

YamlPipelinesDemo + SecretsDemo

Overview Boards Repos Files Commits Pushes Branches Tags Pull requests Advanced Security Pipelines Test Plans Artifacts

main / Infrastructure / webapp.bicep

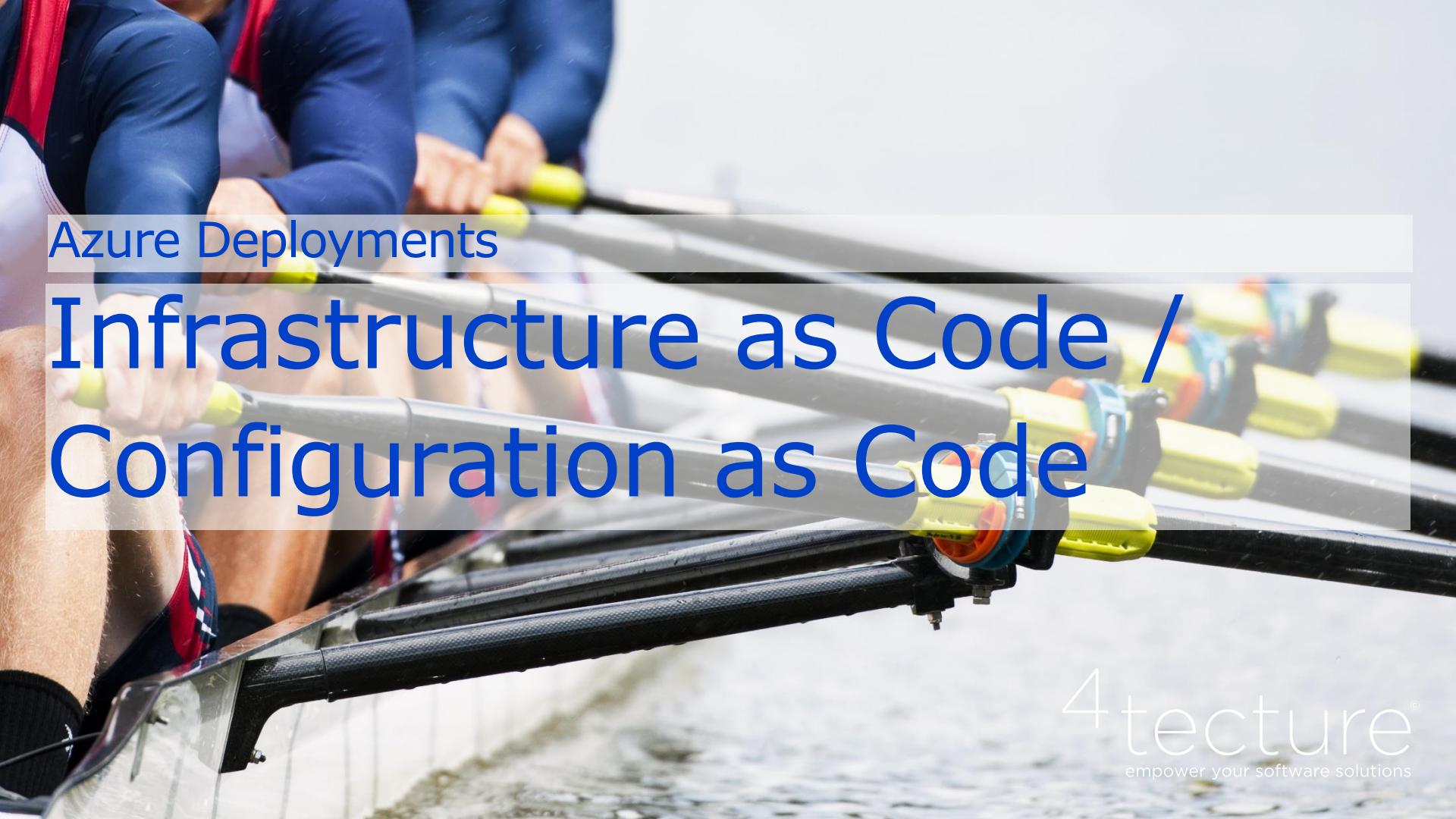
Edit

webapp.bicep

Contents History Compare Blame

```
21 var keyVaultUniqueName = '${keyVaultName}${uniqueString(resourceGroup().id)}'
22
23 resource appServicePlan 'Microsoft.Web/serverfarms@2024-04-01' = {
24   name: appServicePlanName
25   location: location
26   sku: {
27     name: sku
28   }
29   kind: 'linux'
30   properties: {
31     reserved: true
32   }
33 }
34
35 resource webApp 'Microsoft.Web/sites@2024-04-01' = {
36   name: webAppPortalName
37   location: location
38   kind: 'app'
39   identity: {
40     type: 'SystemAssigned'
41   }
42   properties: {
43     serverFarmId: appServicePlan.id
44     siteConfig: {
45       linuxFxVersion: linuxFxVersion
46       webSocketsEnabled: true
47     }
48   }
49 }
50
51 resource keyVault 'Microsoft.KeyVault/vaults/accessPolicies@2023-07-01' = {
52   name: '${keyVaultUniqueName}/add'
53   properties: {
54     accessPolicies: [
55       {
56         tenantId: subscription().tenantId
57         objectId: reference(webApp.id, '2019-08-01', 'full').identity.principalId
58         permissions: {
59           secrets: [
60             'get'
61             'list'
62           ]
63         }
64       ]
65     ]
66   }
67 }
```

Demo Recap

A close-up, low-angle shot of several rowers in a racing shell. Their hands are gripping yellow oars, which are partially submerged in water. The rowers are wearing blue and red athletic gear. The background is blurred, suggesting motion on a body of water.

Azure Deployments

Infrastructure as Code / Configuration as Code

Configuration as Code

- Pipelines, Templates, Variables / Values files stored in Git
- Do not store secrets in Git!
- Mono-Repo / Repo per Service
- Staging is implemented by
 - Feature Branches → PR → PR Deployment with QA → PR Approval / Integration
 - Main Branch → Pre Production → Production (checks and approvals, deployment rings)

DEMO

Simple Deployment



YamlPipelinesDemo +

- Overview
- Boards
- Repos
- Files**
- Commits
- Pushes
- Branches
- Tags
- Pull requests
- Advanced Security

SecretsDemo

main / Infrastructure / webapp.bicep

webapp.bicep

Contents History Compare Blame

```
1 @description('Base name of the resource such as web app name and app service plan ')
2 @minLength(2)
3 param webAppName string = 'secretsdemo'
4
5 @description('The SKU of App Service Plan ')
6 param sku string = 'F1'
7
8 @description('The Runtime stack of current web app')
9 param linuxFxVersion string = 'DOTNETCORE|9.0'
10
11 @description('Location for all resources.')
12 param location string = resourceGroup().location
13
14 @description('Specifies the name of the key vault.')
15 @minLength(1)
16 @maxLength(11)
17 param keyVaultName string
18
19 var webAppPortalName = '${webAppName}-webapp'
20 var appServicePlanName = 'AppServicePlan-${webAppName}'
21 var keyVaultUniqueName = '${keyVaultName}${uniqueString(resourceGroup().id)}'
22
23 resource appServicePlan 'Microsoft.Web/serverfarms@2024-04-01' = {
24   name: appServicePlanName
25   location: location
26   sku: {
27     name: sku
28   }
29   kind: 'linux'
30   properties: {
31     reserved: true
32   }
33 }
34
35 resource webApp 'Microsoft.Web/sites@2024-04-01' = {
36   name: webAppPortalName
37   location: location
38   kind: 'app'
39   identity: {
40     type: 'SystemAssigned'
41   }
42   properties: {
43     serverFarmId: appServicePlan.id
44     siteConfig: {
45       linuxFxVersion: linuxFxVersion
46       webSocketsEnabled: true
47     }
48 }
```



YamlPipelinesDemo +

- Overview
- Boards
- Repos
- Pipelines
- Pipelines
- Environments
- Releases
- Library
- Task groups
- Deployment groups
- Test Plans
- Artifacts

← 03_KeyVaultDemo.yml

Variables

Run

Show assistant

```
48     steps:  
49       - download: current  
50         artifact: '$(infrastructureArtifactName)'  
51         displayName: 'Download pipeline artifacts'  
52       - download: current  
53         artifact: '$(webArtifactName)'  
54         displayName: 'Download pipeline artifacts'  
55     - task: AzureCLI@2  
56       inputs:  
57         azureSubscription: $(AzureConnection)  
58         scriptType: bash  
59         scriptLocation: inlineScript  
60         inlineScript: |  
61           az group create --name $(resourceGroupName) --location $(location)  
62           az deployment group create --resource-group $(resourceGroupName) --template-file $(Pipeline.Workspace)/Infrastructure/storage-and-keyvault.bicep  
63         displayName: "Deploy Storage and KeyVault Infrastructure"  
64     - task: AzureCLI@2  
65       inputs:  
66         azureSubscription: $(AzureConnection)  
67         scriptType: bash  
68         scriptLocation: inlineScript  
69         inlineScript: |  
70           az group create --name $(resourceGroupName) --location $(location)  
71           az deployment group create --resource-group $(resourceGroupName) --template-file $(Pipeline.Workspace)/Infrastructure/webapp.bicep --parameters @  
72         displayName: "Deploy WebApp Infrastructure"  
73     - task: AzureWebApp@1  
74       displayName: 'Deploy app to azure web app'  
75       inputs:  
76         azureSubscription: '$(AzureConnection)'  
77         appType: 'webAppLinux'  
78         appName: 'secretsdemo-webapp'  
79         package: '$(Pipeline.Workspace)/**/HelloWorldSecrets.zip'  
80  
81
```

Demo Recap

DEMO

Complex Deployment



Deploy Cluster Services

- Cluster setup has a lot of Helm chart deployments
 - Configuration as Code
 - Pipeline automation on configuration object
-
- Runtime Parameters / Parameters support objects
 - Define your cluster configuration as object

Azure DevOps 4tecture-demo / k8sDemo / Repos / Files / Infrastructure

Search

k8sDemo

Overview Boards Repos Files Commits Pushes Branches Tags Pull requests Advanced Security Pipelines Test Plans Artifacts

Infrastructure

feature/privateEndpoints / azure / k8senvironment.bicep

k8senvironment.bicep

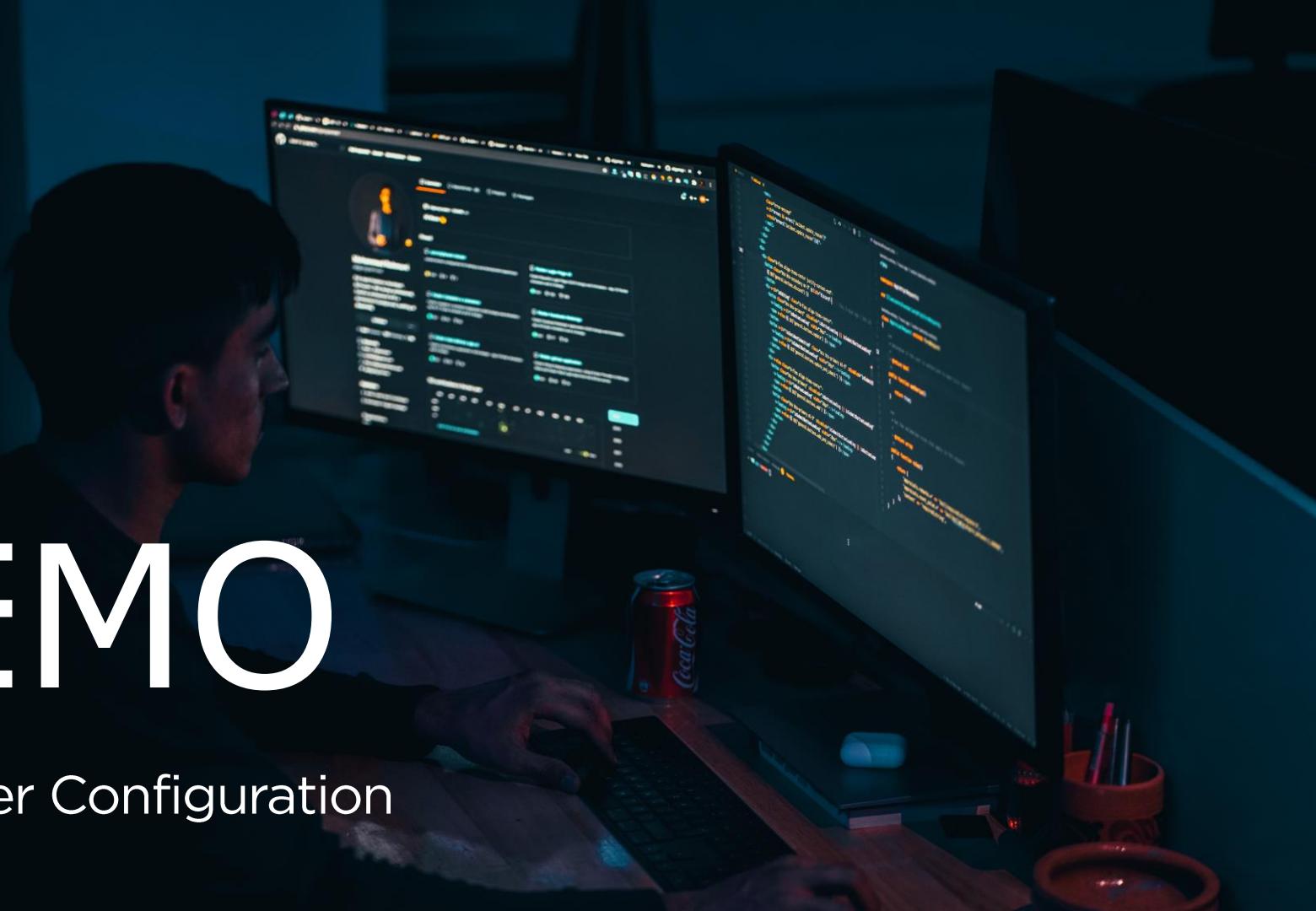
Contents History Compare Blame

64 param agentVmPatToken string
65
66
67 module vnet 'vnet.bicep' = {
68 name: 'vnetDeploy'
69 params: {
70 vnetName: '\${prefix}-vnet-\${clusterName}-\${location}'
71 subnetName: '\${prefix}-snet-\${clusterName}-\${location}'
72 vnetAddressPrefixes: vnetAddressPrefixes
73 subnetAddressPrefix: subnetAddressPrefix
74 tags: tags
75 location: location
76 }
77 scope: resourceGroup()
78 }
79
80 module privateDNS 'privatedns.bicep' = {
81 name: 'privateDNSDeploy'
82 params: {
83 vnetId: vnet.outputs.vnetId
84 tags: tags
85 userAssignedIdentityPrincipalId: aksidentity.outputs.identityPrincipalId
86 }
87 dependsOn: [
88 vnet
89]
90 }
91
92 module aksidentity 'aksidentity.bicep' = {
93 name: 'aksidentityDeploy'
94 params: {
95 clusterName: clusterName
96 prefix: prefix
97 tags: tags
98 location: location
99 }
100 }
101
102 module aks 'aks.bicep' = {
103 name: 'aksDeploy'
104 params: {
105 prefix: prefix
106 clusterName: clusterName
107 subnetId: vnet.outputs.subnetId
108 nodeAdminUsername: nodeAdminUsername
109 adminGroupObjectIDs: adminGroupObjectIDs
110 }

Demo Recap

DEMO

K8s Cluster Configuration



Parameter with all charts

Azure DevOps 4tecture-demo / k8sDemo / Pipelines

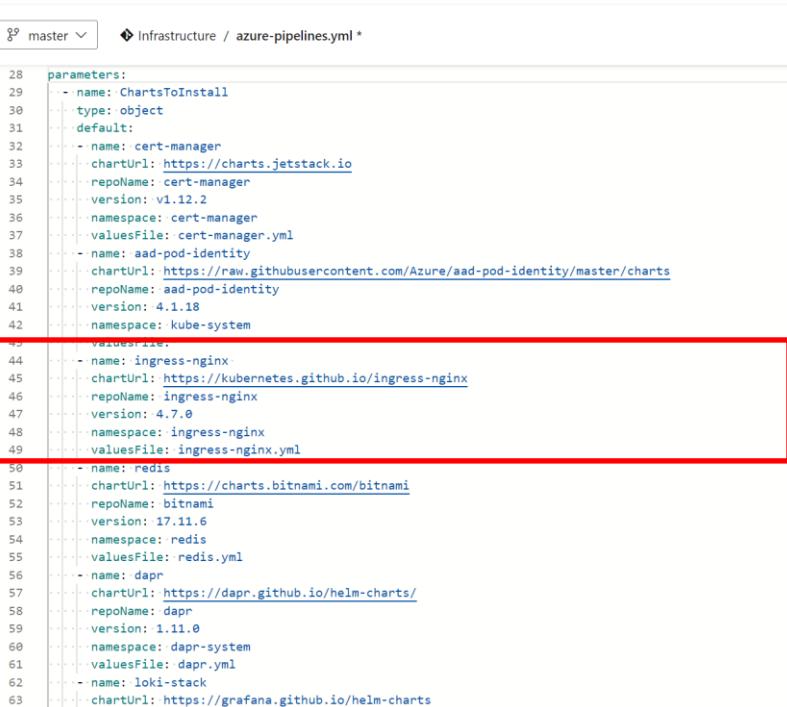
k8sDemo ← AzureK8SInfrastructure

Variables Save ⋮

Show assistant

```
parameters:
  - name: ChartsToInstall
    type: object
    default:
      - name: cert-manager
        chartUrl: https://charts.jetstack.io
        repoName: cert-manager
        version: v1.12.2
        namespace: cert-manager
        valuesFile: cert-manager.yaml
      - name: aad-pod-identity
        chartUrl: https://raw.githubusercontent.com/Azure/aad-pod-identity/master/charts
        repoName: aad-pod-identity
        version: 4.1.18
        namespace: kube-system
        valuesFile: aad-pod-identity.yaml
      - name: ingress-nginx
        chartUrl: https://kubernetes.github.io/ingress-nginx
        repoName: ingress-nginx
        version: 4.7.0
        namespace: ingress-nginx
        valuesFile: ingress-nginx.yaml
      - name: redis
        chartUrl: https://charts.bitnami.com/bitnami
        repoName: bitnami
        version: 17.11.6
        namespace: redis
        valuesFile: redis.yaml
      - name: dapr
        chartUrl: https://dapr.github.io/helm-charts/
        repoName: dapr
        version: 1.11.0
        namespace: dapr-system
        valuesFile: dapr.yaml
      - name: loki-stack
        chartUrl: https://grafana.github.io/helm-charts
```

Project settings <>



Configuration as Code

The screenshot shows the Azure DevOps interface for the 'k8sDemo' repository. The left sidebar navigation bar is visible, with 'Files' selected. The main content area shows the 'Infrastructure' folder structure under 'kubernetes-config'. The 'valuesfiles' folder contains several files, with 'ingress-nginx.yml' being viewed. The code editor displays the YAML configuration for an Ingress controller.

```
controller:
  # Will add custom configuration options to Nginx https://kubernetes.github.io/ingress-nginx/user-guide/nginx-configuration/configuration/
  config:
    proxy-buffers: "8 64k"
    proxy-buffer-size: 64k
    http2-max-field-size: 64k
    http2-max-header-size: 64k
    large-client-header-buffers: "4 64k"

  service:
    annotations:
      | service.beta.kubernetes.io/azure-load-balancer-health-probe-request-path: /healthz

    ## Annotations to be added to controller pods
    ##
    podAnnotations:
      | prometheus.io/scrape: true
      | prometheus.io/port: 10254

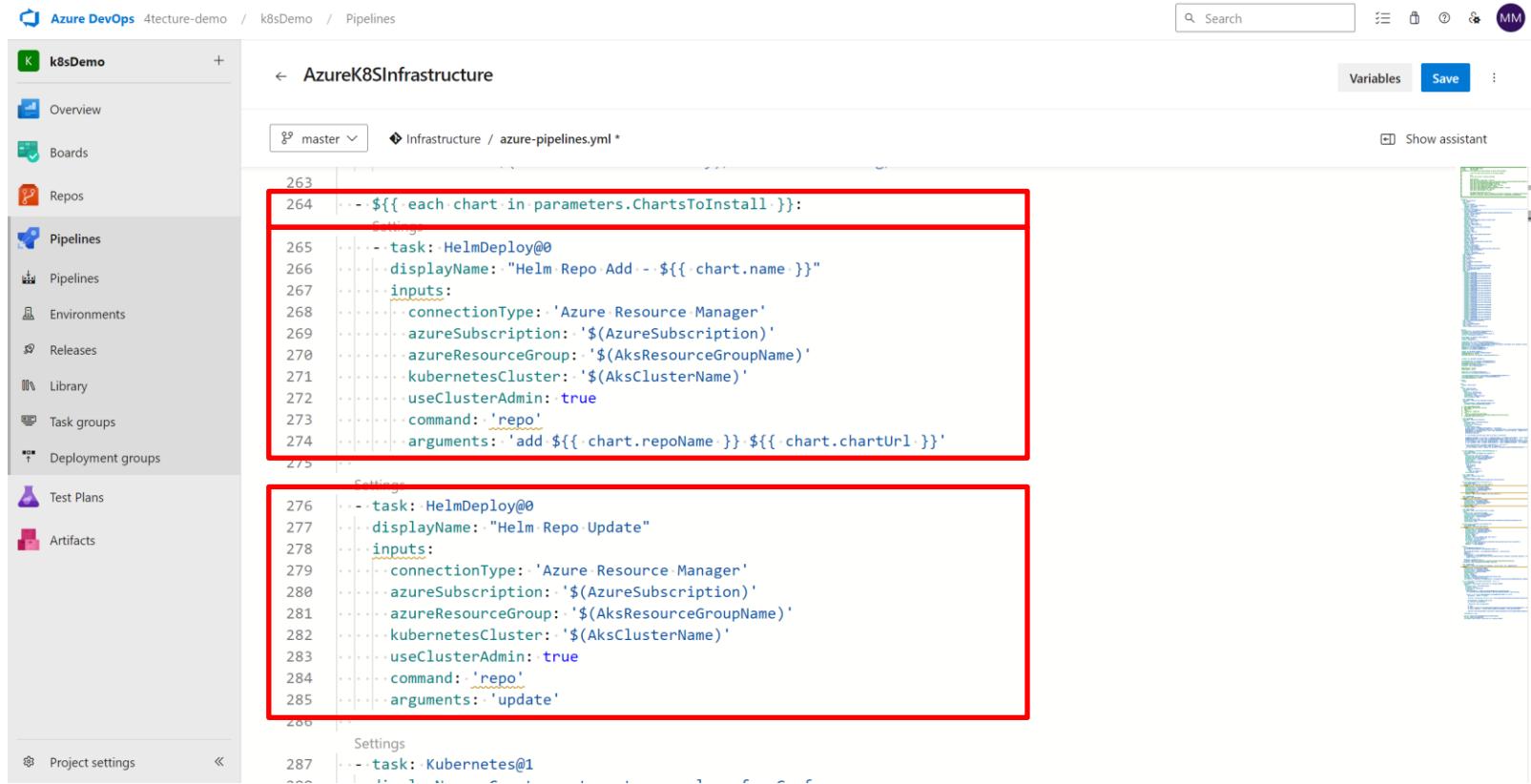
    replicaCount: 2

    ## Node labels for controller pod assignment
    ## Ref: https://kubernetes.io/docs/user-guide/node-selection/
    ##
    nodeSelector:
      | kubernetes.io/os: linux

    admissionWebhooks:
      patch:
        nodeSelector:
          | kubernetes.io/os: linux

    metrics:
      enabled: true
```

Add and update Helm Repos



The screenshot shows the Azure DevOps Pipeline Editor for a project named "k8sDemo". The pipeline file is "Infrastructure / azure-pipelines.yml". Two specific sections of the YAML code are highlighted with red boxes:

```
263
264 - ${{ each chart in parametersChartsToInstall }}:
265   task: HelmDeploy@0
266   displayName: "Helm Repo Add - ${chart.name}"
267   inputs:
268     connectionType: 'Azure Resource Manager'
269     azureSubscription: '${AzureSubscription}'
270     azureResourceGroup: '${AksResourceGroupName}'
271     kubernetesCluster: '${AksClusterName}'
272     useClusterAdmin: true
273     command: 'repo'
274     arguments: 'add ${chart.repoName} ${chart.chartUrl}'
275
276   task: HelmDeploy@0
277   displayName: "Helm Repo Update"
278   inputs:
279     connectionType: 'Azure Resource Manager'
280     azureSubscription: '${AzureSubscription}'
281     azureResourceGroup: '${AksResourceGroupName}'
282     kubernetesCluster: '${AksClusterName}'
283     useClusterAdmin: true
284     command: 'repo'
285     arguments: 'update'
286
287   task: Kubernetes@1
```

Install helm charts

The screenshot shows the Azure DevOps Pipeline Editor interface. On the left, there's a sidebar with project navigation (k8sDemo), a list of pipelines (Pipelines, Pipelines, Environments, Releases, Library, Task groups, Deployment groups), and artifact management (Test Plans, Artifacts). The main area displays the 'azure-pipelines.yml' configuration file for the 'AzureK8SInfrastructure' pipeline. A red box highlights the section of the code responsible for installing Helm charts. The code uses a 'each' loop to iterate over 'parameters.ChartsToInstall'. Each iteration defines a task named 'HelmDeploy@0' with various inputs like connection type, subscription, resource group, cluster name, and chart details. The full code is as follows:

```
299 -> ${{ each chart in parameters.ChartsToInstall }}:
300   task: HelmDeploy@0
301   displayName: "Install-${{ chart.name }}"
302   inputs:
303     connectionType: 'Azure Resource Manager'
304     azureSubscription: '$(AzureSubscription)'
305     azureResourceGroup: '$(AksResourceGroupName)'
306     kubernetesCluster: '$(AksClusterName)'
307     useClusterAdmin: true
308     command: 'upgrade'
309     chartType: 'Name'
310     chartName: '${{ chart.repoName }}/${{ chart.name }}'
311     chartVersion: '${{ chart.version }}'
312     releaseName: '${{ chart.name }}'
313     ${{ if chart.valuesFile }}:
314       valueFile: $(Build.SourcesDirectory)/kubernetes-config/valuesfiles/${{ chart.valuesFile }}
315     namespace: ${{ chart.namespace }}
316     arguments: '--create-namespace'
317
318   pwsh: |
319     $trainingWorkEnvironmentsJson = @"
320     ${{ convertToJson(parameters.TrainingWorkEnvironment) }}
321     @@
322     $trainingWorkEnvironments = $trainingWorkEnvironmentsJson | ConvertFrom-Json
323     $index = 0
324     $setValues = ""
325     foreach($workEnv in $trainingWorkEnvironments){
```

A close-up, low-angle shot of several rowers in a racing shell. Their legs and torsos are visible as they pull on the oars. The water is choppy, creating white spray at the blades. The oars have yellow and orange grips. The background is blurred.

Azure Deployments

Traditional CI/CD vs. GitOps

4tecture®
empower your software solutions



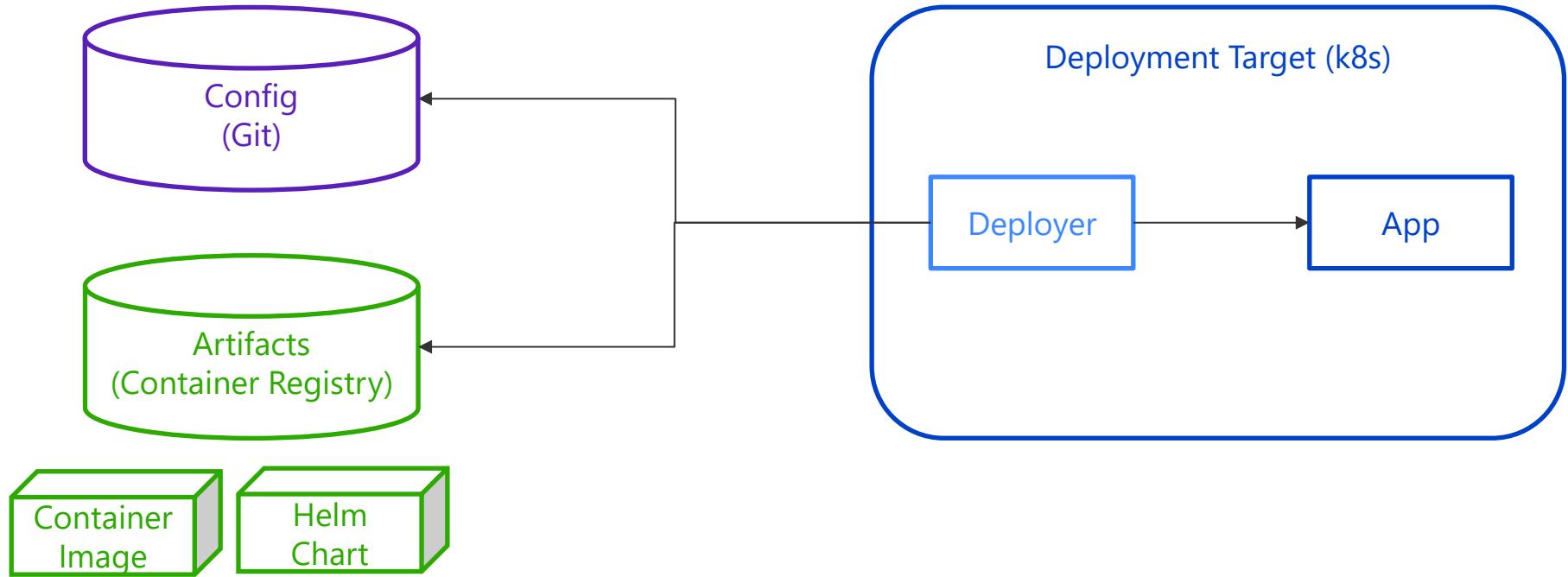
KEEP
CALM
AND

GIT COMMIT
GIT PUSH ORIGIN MAIN

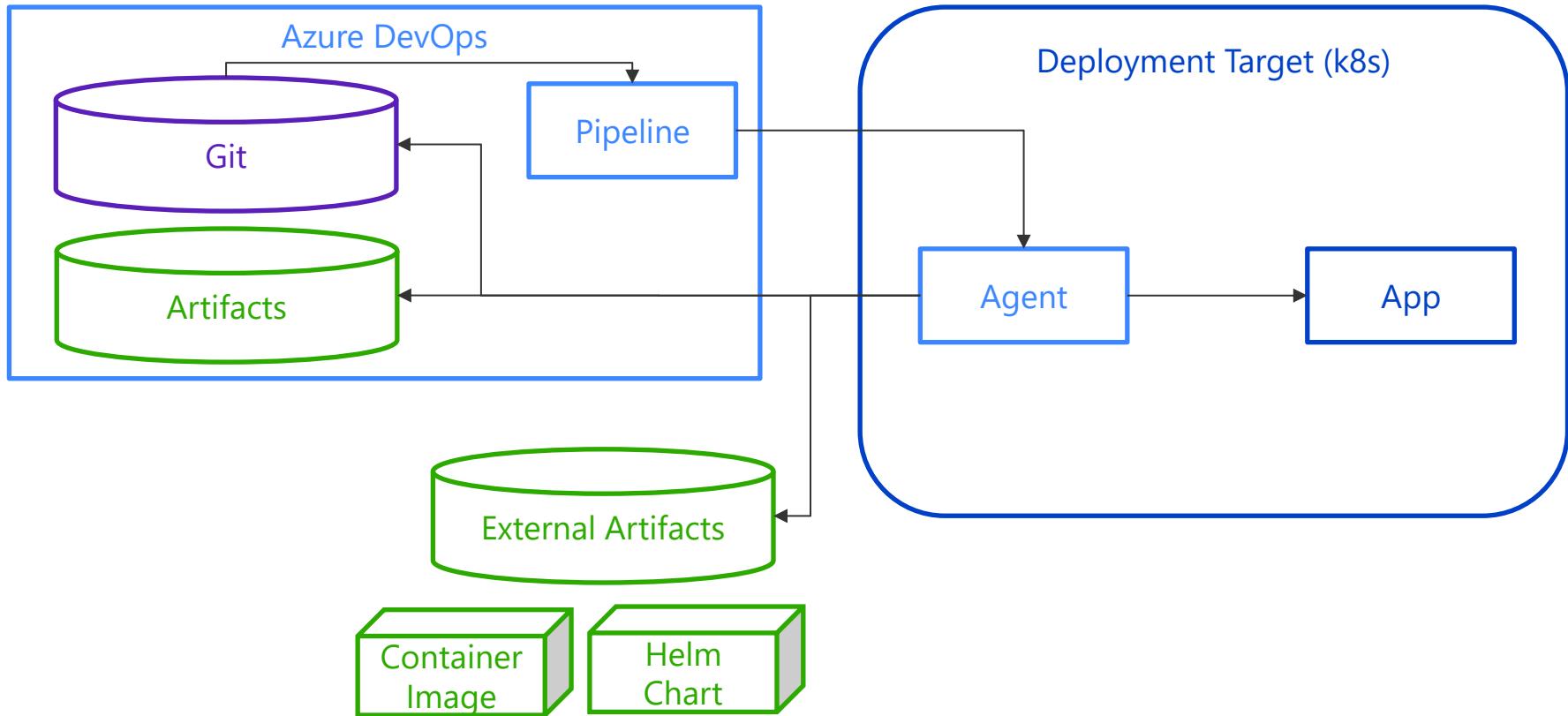
GitOps

By following these principles, GitOps aims to make operations and deployment more consistent, repeatable, and secure.

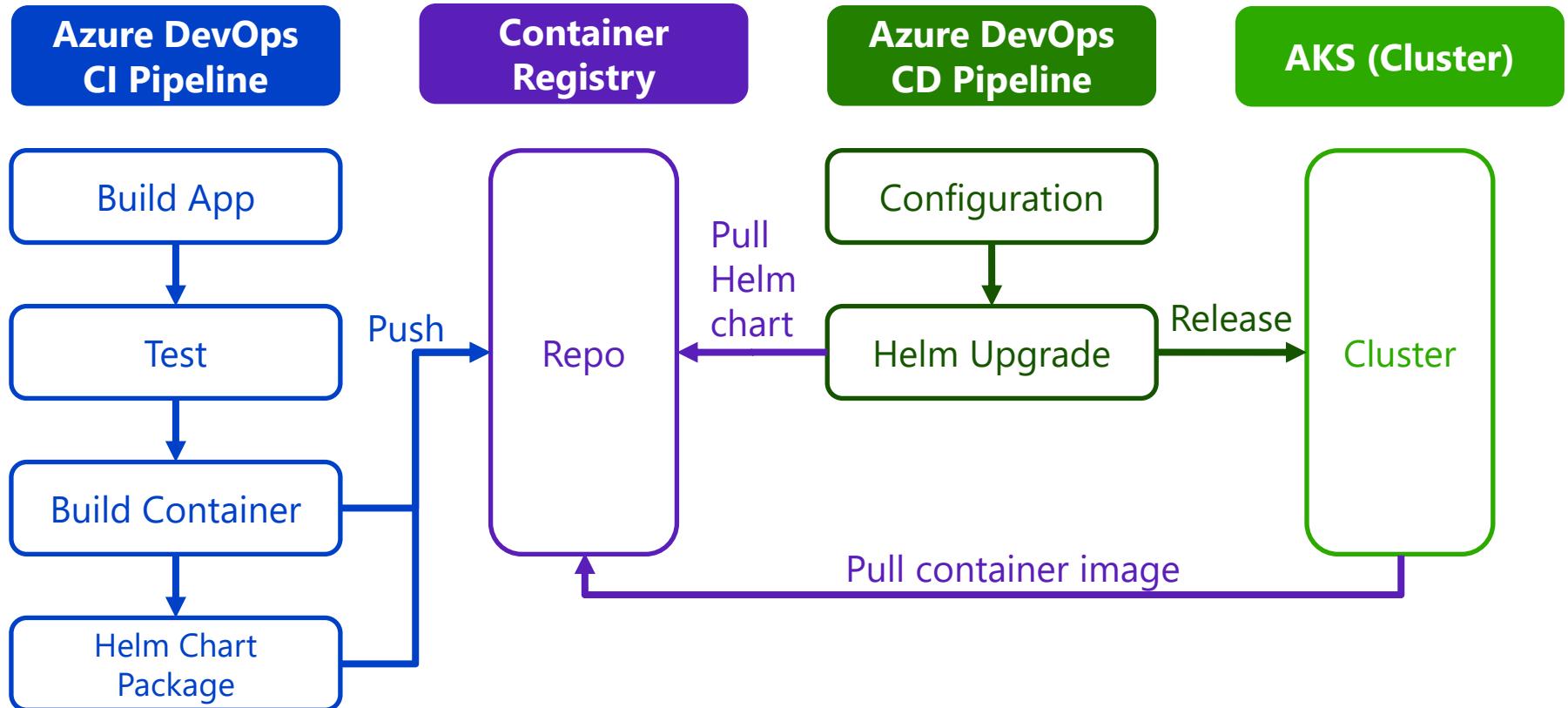
General Setup



Azure DevOps

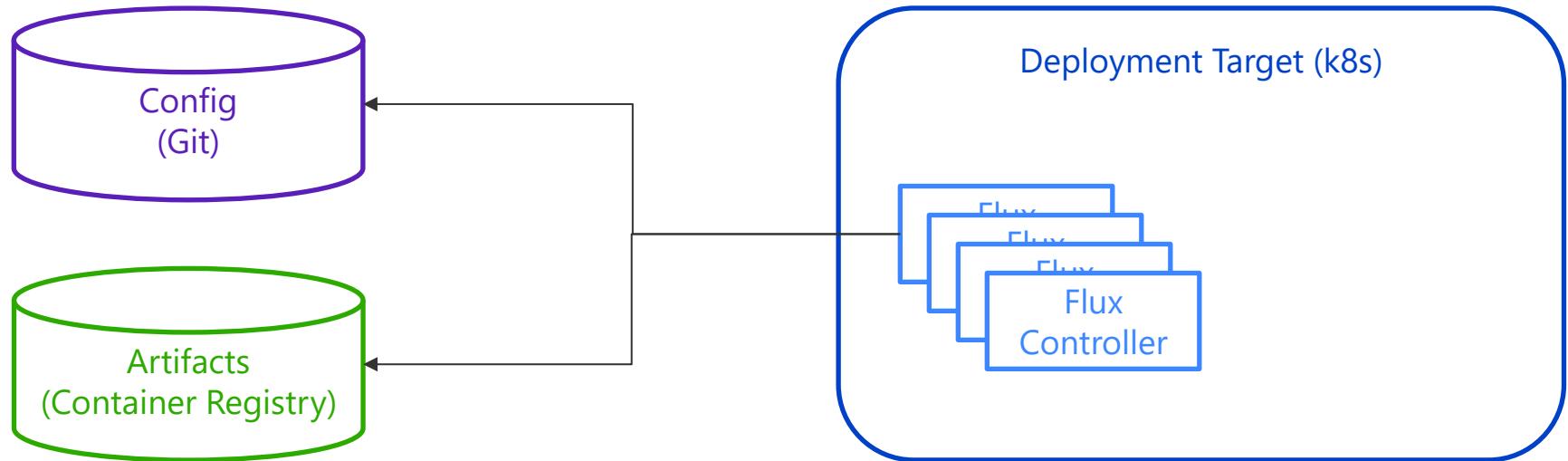


Azure DevOps & AKS CI / CD

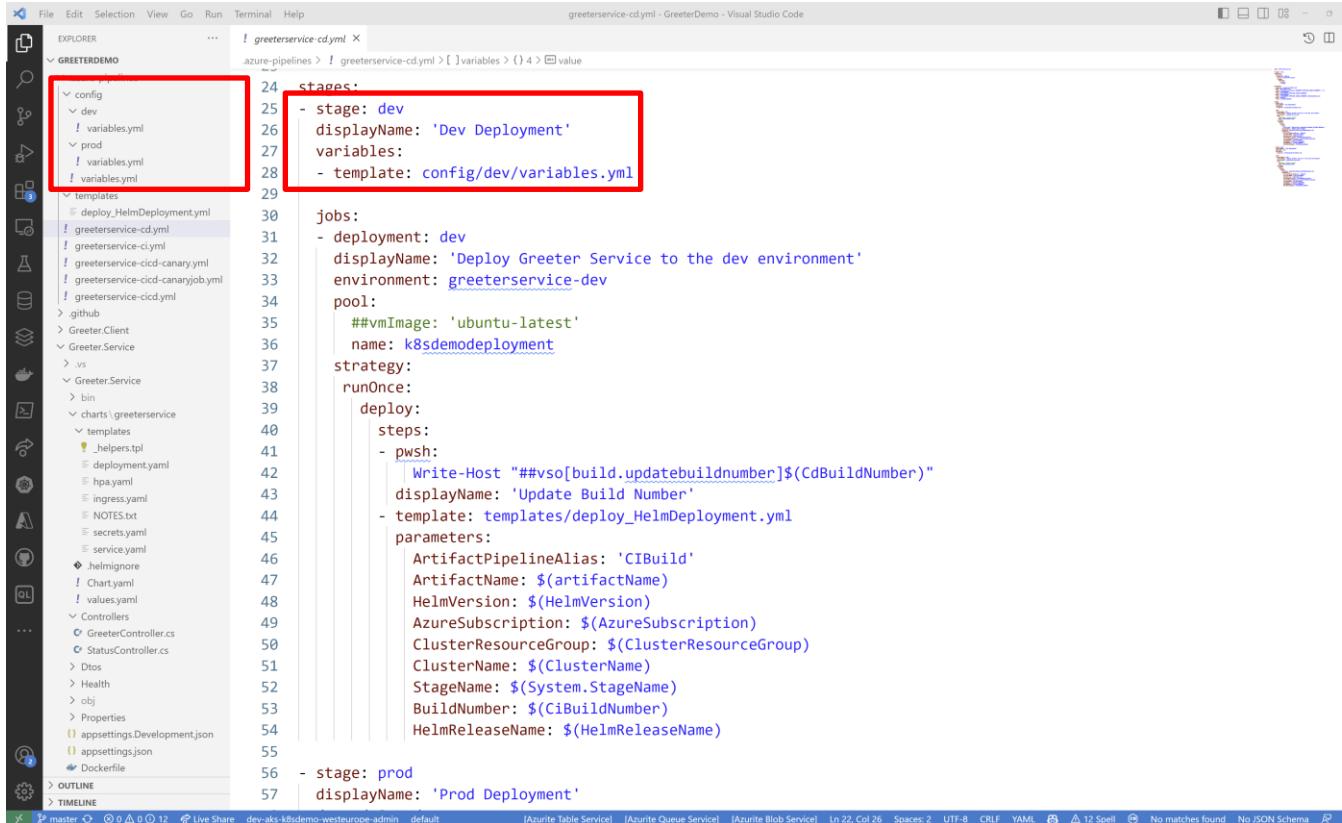


FluxCD

- Install the CLI
- Installation in k8s
 - flux bootstrap
 - flux install
 - Add kustomizations, sources, ...



Dedicated Settings per Stage



```
greeterService-cd.yml - GreeterDemo - Visual Studio Code
greeterService-cd.yml
...
24 stages:
25 - stage: dev
26   displayName: 'Dev Deployment'
27   variables:
28     - template: config/dev/variables.yml
29
30 jobs:
31 - deployment: dev
32   displayName: 'Deploy Greeter Service to the dev environment'
33   environment: greeterService-dev
34   pool:
35     ##vmImage: 'ubuntu-latest'
36     name: k8sdemodeployment
37   strategy:
38     runOnce:
39       deploy:
40         steps:
41           - pwsh:
42             Write-Host "##vso[build.updatebuildnumber]$(CdBuildNumber)"
43             displayName: 'Update Build Number'
44           - template: templates/deploy_HelmDeployment.yml
45         parameters:
46           ArtifactPipelineAlias: 'CIBuild'
47           ArtifacName: $(artifactName)
48           HelmVersion: $(HelmVersion)
49           AzureSubscription: $(AzureSubscription)
50           ClusterResourceGroup: $(ClusterResourceGroup)
51           ClusterName: $(ClusterName)
52           StageName: $(System.StageName)
53           BuildNumber: $(CiBuildNumber)
54           HelmReleaseName: $(HelmReleaseName)
55
56 - stage: prod
57   displayName: 'Prod Deployment'
```

Dedicated Settings per Stage

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under the 'GREETERDEMO' folder:
 - .azure-pipelines (with config and dev subfolders)
 - variables.yml (selected)
 - prod (with variables.yml and variables.yaml files)
 - templates (with deploy.HelmDeployment.yaml, greeterservice-cd.yaml, greeterservice-clym.yaml, greeterservice-cicd-canary.yaml, greeterservice-cicd-canaryjob.yaml, and greeterservice-cicd.yaml)
 - .github
 - Greeter.Client
 - Greeter.Service
 - vs
 - Greeter.Service
 - bin
 - charts/greeterservice
 - templates (_helpers.tpl, deployment.yaml, hpa.yaml, ingress.yaml, NOTES.txt, secrets.yaml, service.yaml)
 - .helmignore
 - Chart.yaml
 - values.yaml
 - Controllers (GreeterController.cs, StatusController.cs)
 - Dtos
 - Health
- Editor (Center):** The 'variables.yml' file is open, showing configuration for the 'dev' stage:

```
1 variables:
2   hpaAverageCpuUtilization: '50'
3   hpaEnabled: 'false'
4   hpaMaxReplicas: '2'
5   hpaMinReplicas: '1'
6   ReplicaCount: '1'
7   ServiceMessage: 'Hello from Azure Pipelines Dev!'
8   EnablePrimeNumberCalculation: 'false'
```
- Bottom Status Bar:** Shows the current branch (master), repository (dev-aks-k8sdemo-westeurope-admin), and default tab (YAML). Other tabs include Azure Table Service, Queue Service, Blob Service, and Spell Check.

Comparison

Pipelines

- Everything in Git
- Versatile
- Full DevOps Lifecycle
- End-to-end Traceability
- Integrated Test Automation

Flux

- Everything in Git
- K8s Deployments
- Focus on Continuous Deployment
- Independent / Various Sources

A close-up, low-angle shot of several rowers in a racing shell. Their legs and the oars they are pulling are visible. The oars have bright yellow handles and black blades. The water is choppy, creating white spray around the oars.

Azure Deployments

Agents

4tecture®
empower your software solutions

The Agent

- Multi-purpose task executer
- Executes any task
- Orchestration by Azure Pipelines

Considerations / Challenges

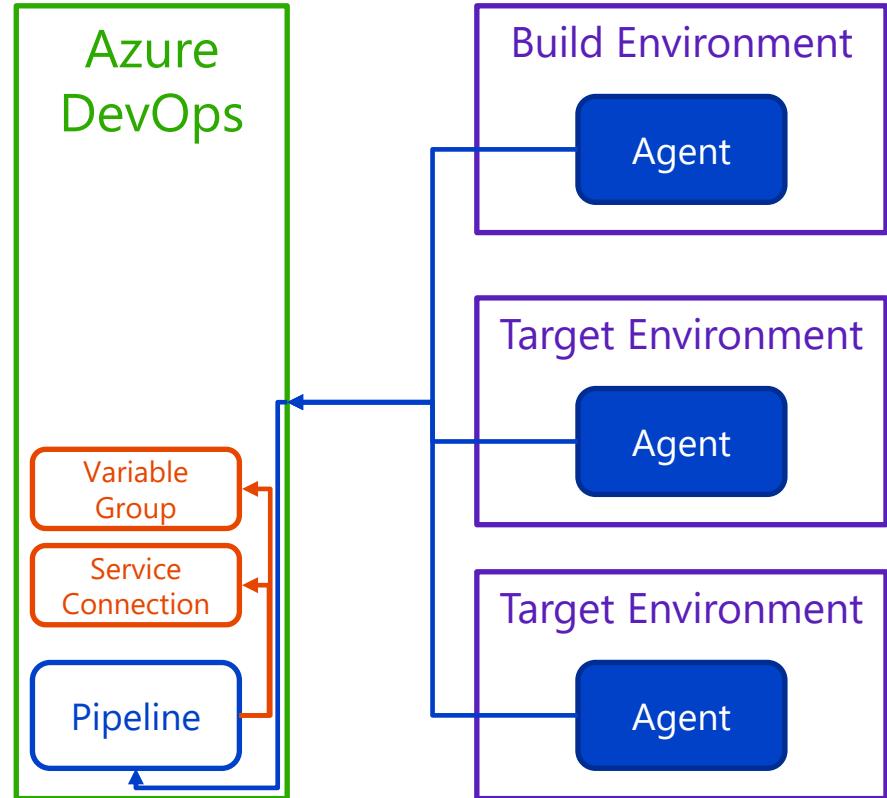
- Tool Dependencies
- Build old and new states of the repo
- Network access / restrictions
- Scaling
- Caching

Agent Hosting Options

Hosted Agent	Virtual Machine Scale Set Pool	Managed DevOps Pool	Private Agent
<ul style="list-style-type: none">• Cloud-Only• Fixed image for Windows, Linux, Mac• Monthly fee per parallel agent instance• Job duration limits	<ul style="list-style-type: none">• Cloud-Only• Run agents in your VMSS resource• Scaled by Azure DevOps Services• You provide VM images• Configurable VM resources	<ul style="list-style-type: none">• Cloud-Only• Hosted on your behalf• Quick-starter images and custom images• Stateful agents• Private networking• Configurable VM resources	<ul style="list-style-type: none">• Runs everywhere• Self-installed• Self-scaled

Agent Deployment

- Agent per Purpose (build, deploy target)
- Agent as near as possible to deployment target
- Connectivity from inside-out, no hole punching, isolate networking
- Least privileged
- Use secret from Azure DevOps, per environment vault instance
- Permit resource usage per pipeline



DEMO

Private Agent in VM



k8sDemo

- Overview
- Boards
- Repos
- Files
- Commits
- Pushes
- Branches
- Tags
- Pull requests
- Advanced Security

Pipelines

Test Plans

Artifacts

Infrastructure

- azure
 - aks.bicep
 - aksIdentity.bicep
 - azdoagentvm.bicep
 - k8senvironment-dev.param
 - k8senvironment.parameters
 - privatedns.bicep
 - rbac.bicep
 - registry.bicep
 - sql.bicep
 - vnet.bicep
- deploymentagent
- kubernetes-config
- linuxworker
- scripts
- .gitignore
- azure-pipelines.yml
- README.md

feature/privateEndpoints

azdoagentvm.bicep

Contents History Compare Blame

```
1 @description('The environment prefix of the Managed Cluster resource e.g. dev, prod, etc.')
2 param prefix string
3 @description('The name of the Managed Cluster resource')
4 param clusterName string
5 @description('Tags for the resources')
6 param tags object
7 @description('Location')
8 param location string = resourceGroup().location
9 param adminUsername string
10 @secure()
11 param adminPassword string
12 param vmSize string = 'Standard_B2s'
13 param subnetId string
14 param agentPool string
15 param agentName string
16 param azdoUrl string
17 @secure()
18 param patToken string
19
20 resource publicIP 'Microsoft.Network/publicIPAddresses@2020-11-01' = {
21   name: '${prefix}-agentvm-${clusterName}-${location}-pip'
22   location: location
23   properties: {
24     publicIPAllocationMethod: 'Dynamic'
25   }
26 }
27
28 resource networkInterface 'Microsoft.Network/networkInterfaces@2020-11-01' = {
29   name: '${prefix}-agentvm-${clusterName}-${location}-nic'
30   location: location
31   properties: {
32     ipConfigurations: [
33       {
34         name: 'ipconfig1'
35         properties: {
36           privateIPAllocationMethod: 'Dynamic'
37           subnet: {
38             id: subnetId
39           }
40           publicIPAddress: {
41             id: publicIP.id
42           }
43         }
44       ]
45     }
46   }
47 }
```



k8sDemo

- Overview
- Boards
- Repos
- Files
- Commits
- Pushes
- Branches
- Tags
- Pull requests
- Advanced Security

Pipelines

Test Plans

Artifacts

Infrastructure

azdoagentvm.bicep

Contents History Compare Blame

```
48
49 resource virtualMachine 'Microsoft.Compute/virtualMachines@2021-04-01' = {
50   name: '${prefix}-agentvm-${clusterName}-${location}'
51   location: location
52   properties: {
53     hardwareProfile: {
54       vmSize: vmSize
55     }
56     osProfile: {
57       computerName: 'agentvm'
58       adminUsername: adminUsername
59       adminPassword: adminPassword
60       linuxConfiguration: {
61         disablePasswordAuthentication: false
62       }
63     }
64     storageProfile: {
65       imageReference: {
66         publisher: 'Canonical'
67         offer: 'UbuntuServer'
68         sku: '18.04-LTS'
69         version: 'latest'
70       }
71       osDisk: {
72         createOption: 'FromImage'
73       }
74     }
75     networkProfile: {
76       networkInterfaces: [
77         {
78           id: networkInterface.id
79         }
80       ]
81     }
82   }
83   tags: tags
84 }
85
86 resource customScriptExtension 'Microsoft.Compute/virtualMachines/extensions@2021-04-01' = {
87   parent: virtualMachine
88   name: 'installAgent'
89   location: location
90   properties: {
91     publisher: 'Microsoft.Azure.Extensions'
92     type: 'CustomScript'
93     typeHandlerVersion: '2.0'
94     autoUpgradeMinorVersion: true
```

Demo Recap

Azure DevOps 4tecture-demo / k8sDemo / Repos / Files / Infrastructure

Search

k8sDemo + Infrastructure

Overview Boards Repos Files Commits Pushes Branches Tags Pull requests Advanced Security Pipelines Test Plans Artifacts

azdoagentvm.bicep

Contents History Compare Blame

```
tags: tags
}
resource customScriptExtension 'Microsoft.Compute/virtualMachines/extensions@2021-04-01' = {
    parent: virtualMachine
    name: 'installAgent'
    location: location
    properties: {
        publisher: 'Microsoft.Azure.Extensions'
        type: 'CustomScript'
        typeHandlerVersion: '2.0'
        autoUpgradeMinorVersion: true
        settings: {
            script: base64(concat('
#!/bin/bash

# Variables
AGENT_POOL=""", agentPool, """
AGENT_NAME=""", agentName, """
AZDO_URL=""", azdoUrl, """
PAT_TOKEN=""", patToken, """

# Update the system
apt-get update
apt-get upgrade -y

# Install prerequisites
apt-get install -y liblttng-ust0 libkrb5-3 zlib1g libicu60 libssl1.0.0 libssl1.0.2 libssl1.1 unzip

# Download the Azure DevOps agent
wget https://vtssagentpackage.azureedge.net/agent/3.227.2/vsts-agent-linux-x64-3.227.2.tar.gz
mkdir myagent && cd myagent
tar zxvf ..vsts-agent-linux-x64-3.227.2.tar.gz

# Allow running as root
export AGENT_ALLOW_RUNASROOT="1"

# Configure the agent
./config.sh --url $AZDO_URL --auth pat --token $PAT_TOKEN --pool $AGENT_POOL --agent $AGENT_NAME --replace --acceptTeeEula --runAsService --unattended

# Install the agent service
./svc.sh install
./svc.sh start
'''))
    }
}
}

# feature/privateEndpoints
```

Demo Recap

DEMO

Managed DevOps Pool



Subscriptions

4ecture Demo Environment

[+ Add](#)[Advanced options](#)

Global administrators can manage all subscriptions in this list by updating their policy setting [here](#).

View list of subscriptions for which you have role-based access control (RBAC) permissions to manage Azure resources. To view subscriptions for which you have billing access, [click here](#)

Showing subscriptions in 4ecture Demo Environment directory. Don't see a subscription? [Switch directories](#)

Subscriptions : **Filtered (1 of 1)**My role == **all**Status == **all**[+ Add filter](#)

Subscription name ↑↓

Microsoft Azure Sponsorship



Microsoft Azure Sponsorship | Resource providers

Subscription

 Search[Register](#)[Unregister](#)[Refresh](#)[Feedback](#)[Overview](#)[Activity log](#)[Access control \(IAM\)](#)[Tags](#)[Diagnose and solve problems](#)[Security](#)[Events](#)[Billing](#)[Settings](#)[Programmatic deployment](#)[Resource groups](#)[Resources](#)[Preview features](#)[Usage + quotas](#)[Policies](#)[My permissions](#)[Resource providers](#)[Deployments](#)[Deployment stacks](#)[Properties](#)[Resource locks](#)[Help](#) devops[X](#)

Status : All

Registration Policy : All

Provider ↑

Status

Registration Policy

Microsoft.DevOpsInfrastructure

...

Registering

RegistrationRequired



Home >

Create a Managed DevOps Pool ...

Managed DevOps Pools that meet your team needs.

[Basics](#)[Scaling](#)[Networking](#)[Storage](#)[Security](#)[Tags](#)[Review + create](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource group * ⓘ

[Create new](#)

Instance details

Dev Center * ⓘ

Dev Center project * ⓘ

[Create new Dev Center and project](#)

Azure DevOps organization * ⓘ

Pool name * ⓘ

ⓘ This pool will be created in every project in the Azure DevOps organization that is selected. If you want to add this pool only to specific projects, use the security tab to limit the projects it will be visible in.

Region * ⓘ

Maximum agents * ⓘ

Agent size * ⓘ

Name	Cores	RAM	
Standard D2v2s v5	2	8	Change size

[Previous](#)[Next](#)[Review + create](#)[Give feedback](#)

Home >

Create a Managed DevOps Pool ...

Managed DevOps Pools that meet your team needs.

[Basics](#) [Scaling](#) [Networking](#) [Storage](#) [Security](#) [Tags](#) [Review](#)

Pool name * ⓘ

DemoMDP

This pool will be created in every project in the Azure DevOps organization that is selected. If you want to add this pool only to specific projects, go to the security tab to limit the projects it will be visible to.

Region * ⓘ

(Europe) West Europe

Maximum agents * ⓘ

2

Agent size * ⓘ

Name	Cores	RAM
 Standard D2ads v5	2	8

OS disk type ⓘ

Standard

Images

Pick one or more from Images with which to create our CI/CD agents.

[+ Add from Image Library](#)

Name	Version	Aliases
------	---------	---------

Ubuntu Server 20.04 LTS - x64 Gen 2

latest

Image Library

[Azure Pipelines Images](#)[Selected marketplace images](#)[Azure Compute Gallery images](#)

These are Quickstarter VM images that contain the most commonly used software. They are the same VM images used by the Microsoft Hosted Azure Pipelines agents. These images can be used to get started quickly but we recommend creating a VM image with exactly the software your CI/CD pipelines need. All of these images are Gen 1(V1) and x64 architecture.

Name & plan	Description
<input type="checkbox"/> Azure Pipelines - Windows Server 2022	VM Image used by Microsoft Hosted Agents. View full list of software.
<input type="checkbox"/> Azure Pipelines - Windows Server 2019	VM Image used by Microsoft Hosted Agents. View full list of software.
<input type="checkbox"/> Azure Pipelines - Ubuntu 22.04	VM Image used by Microsoft Hosted Agents. View full list of software.
<input type="checkbox"/> Azure Pipelines - Ubuntu 20.04	VM Image used by Microsoft Hosted Agents. View full list of software.

[Previous](#)[Next](#)[Review + create](#)[Add](#)[Cancel](#)

Home >

Create a Managed DevOps Pool ...

X

Managed DevOps Pools that meet your team needs.

Basics **Scaling** Networking Storage Security Tags Review + create

Defaults are selected to keep your total cost low. If you want your agents to be more powerful or to be assigned to your pipelines faster, you will have to choose the right option to balance cost and performance.

Agent state ⓘ

Fresh agent every time

Same agent can be reused by multiple builds

Standby agent mode ⓘ

Off

Standby agent mode off.

Manual

If you know your workloads' usage patterns, you can configure standby agents as per the scale conditions.

Automatic

If you don't know your usage patterns, you can rely on automatic forecasting based on past data.

Max time to live * ⓘ

7.00:00:00

Grace period * ⓘ

0.00:15:00



Home >

Create a Managed DevOps Pool ...

Managed DevOps Pools that meet your team needs.

Basics Scaling **Networking** Storage Security Tags Review + create

You can use features in this section to control networking settings of your agents.

Virtual network type

Isolated virtual network

Agents injected into existing virtual network

Subnet resource

Subscription Id:

Resource group:

Virtual network:

Subnet:

[Configure](#)

Subnet resource

Subscription *

Microsoft Azure Sponsorship

Virtual Network *

ServiceCluster-VNet

Subnet *

ServiceCluster-AksSubnet



Home >

Create a Managed DevOps Pool ...

X

Managed DevOps Pools that meet your team needs.

Basics Scaling Networking Storage **Security** Tags Review + create

You can use features in this section to improve security of your agents.

Use pool in multiple organizations

Add pool to all projects Yes
 No

Enable Interactive Mode

Pool Administration Permissions

Choose who will be able to administer the pool that will be created

Pool Administration Permissions Creator only
 Inherit permissions from project
 Specific accounts



DemoMDP
 Managed DevOps Pool

 Search Delete Refresh
Overview
[JSON View](#)

- [Activity log](#)
- [Access control \(IAM\)](#)
- [Tags](#)
- [Diagnose and solve problems](#)
- [Settings](#)
- [Monitoring](#)
- [Automation](#)
- [Help](#)

Essentials

 Resource group ([move](#)) : [DemoMDP](#)

Name : DemoMDP

Location : West Europe

 Azure DevOps organizati... : <https://dev.azure.com/4tecture-demo>

 Subscription ([move](#)) : [Microsoft Azure Sponsorship](#)

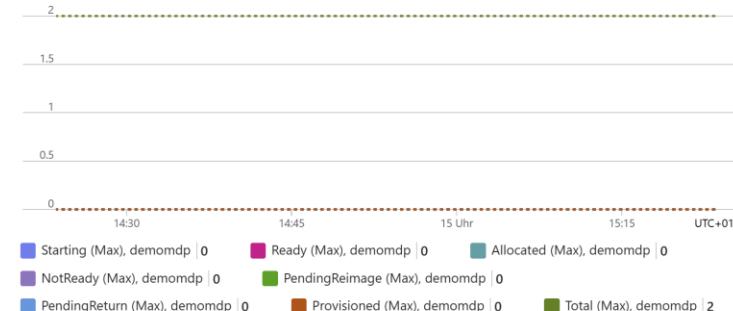
Agent state : Stateful

Subscription ID : 82d445a5-0bf3-454a-804b-2fd0286650c2

Maximum agents : 2

 Tags ([edit](#)) : [Add tags](#)
Metrics

1h 1d 7d 30d

Pool Usage

Pool Provisioning Health

Request Durations

100ms

90ms

80ms

70ms

60ms

50ms

40ms

30ms



- K
- +
- General
- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards
- B
- Boards
- Project configuration
- Team configuration
- GitHub connections
- P
- Agent pools
- Parallel jobs
- Settings
- Test management
- Release retention
- Service connections
- XAML build services
- R
- Repos
- Repositories
- A
- Artifacts
- Storage

DemoMDP

Jobs Details Security Approvals and checks Analytics



No jobs have run on this agent pool

Run a pipeline on this agent pool to see more details

Demo Recap

A dynamic background image showing a team of rowers from behind, wearing blue and red athletic gear. They are in a boat, and their oars are extended forward, creating a sense of motion and teamwork.

Azure Deployments

Dynamic Deployments

4tecture®
empower your software solutions

PR Resources

Why Pull Request Deployments?

- Fail fast - learn fast & fix fast
- Only an integrated change provides clarity if it runs successfully in production
- Pull Request is single point of interaction / status for developers, testers and product owners
- Pipelines with conditions, templates and integration in PR API provide all the tools you need

DEMO

PR Deployment (Resources)



-  k8sDemo
-  Overview
-  Boards
-  Repos
-  Pipelines
-  Pipelines
-  Environments
-  Releases
-  Library
-  Task groups
-  Deployment groups
-  Test Plans

← DevFunApi-CD

Variables Save

 Show assistant

```
70 - deployment: PullRequestDeployment
71   displayName: 'Deploy DevFun API to the PR environment'
72   environment: devfun-api-pr
73   pool:
74     vmImage: 'ubuntu-latest'
75     name: Default
76   container: worker
77   strategy:
78     runOnce:
79       deploy:
80         steps:
81           - download: CIBuild
82             displayName: 'Download artifacts'
```

```
  - task: Bash@3
    inputs:
      targetType: 'inline'
      script: |
        chmod +x $(Pipeline.Workspace)/CIBuild/TestDataInitializer/linux/DevFun.DataInitializer
        $(Pipeline.Workspace)/CIBuild/TestDataInitializer/linux/DevFun.DataInitializer --service \$IngressHostName
    displayName: 'Run DataInitializer'
  - job: ApiTests
```

Deploy service

Initialize test data

Calculate PR specific variables

Setup PR
environment
(k8s namespace,
database,
identities)

Demo
Recap

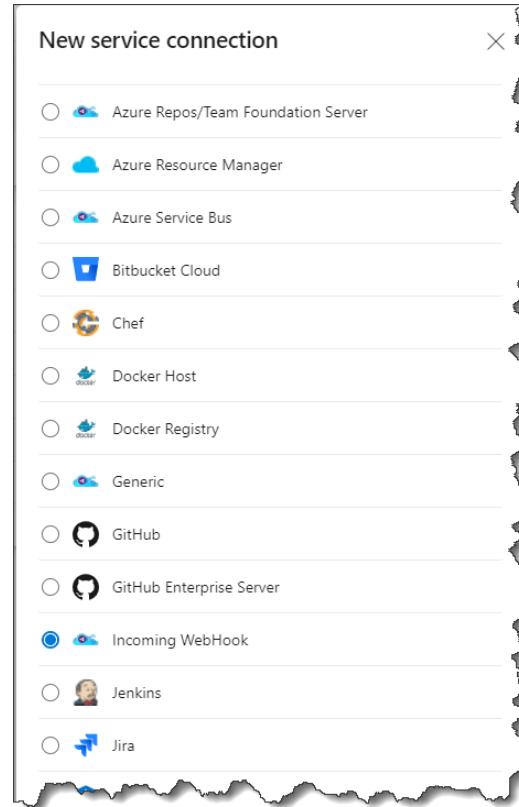
Cleanup

Trigger

- Main triggers are repository / code related
- Many automation scenarios trigger from other events
 - Work Item Update
 - Pull Request Update

Incoming Webhook Service Connection

- Incoming webhook can be defined as service connection
- Generic webhook trigger in pipelines
- Payload can be filtered



Pipeline Resource

- Azure DevOps creates webhook endpoint based on alias
- Pipeline resource triggers pipeline
- Filters can be applied to trigger

```
1  trigger: none
2
3  pool:
4    - vmImage: ubuntu-latest
5
6  resources:
7    - # https://dev.azure.com/4tecture-demo/_apis/public/distributedtask/webhooks/prupdated?api-version=6.0-preview
8    - webhooks:
9      - webhook: PRUpdated..... ## Webhook alias
10     connection: PREventsConnection ... ## Incoming webhook service connection
11     filters:
12       - path: eventType
13       - value: git.pullrequest.updated
14       - path: publisherId
15       - value: tfs
16       - path: resource.repository.name
17       - value: DevFun
18
19  variables:
20    - template: templates/common_variables.yml
21    View template
22
23  triggers:
24    - type: webhook
25      url: https://dev.azure.com/4tecture-demo/_apis/public/distributedtask/webhooks/prupdated?api-version=6.0-preview
26      variables:
27        - name: variable1
28        - name: variable2
29
30  steps:
31    - task: PowerShell@2
32      inputs:
33        targetType: 'inline'
34        script: |
35          Write-Host "Hello from step 1"
36
```

Service Hook & Incoming Webhook

- Service hooks can trigger webhooks for many Azure DevOps Events

- Build completed
- Code pushed
- Elastic agent pool resized
- Pull request commented on
- Pull request created
- Pull request merge attempted
- Pull request updated
- Release abandoned
- Release created
- Release deployment approval completed
- Release deployment approval pending
- Release deployment completed
- Release deployment started
- Run stage approval completed
- Run stage state changed
- Run stage waiting for approval
- Run state changed
- Work item commented on
- Work item created
- Work item deleted
- Work item restored
- Work item updated

The screenshot shows the 'Edit Service Hooks Subscription' dialog box divided into two main sections: 'FILTERS' on the left and 'SETTINGS' on the right.

Trigger on this type of event: Pull request updated

FILTERS:

- Repository: DevFun
- Target branch: [Any]
- Change: Status changed
- Requested by a member of group: [Any]
- Reviewer includes group: [Any]

SETTINGS:

- Action:** Post via HTTP
- This action posts a JSON object representation of the event to the specified URL.
- URL:** /_apis/public/distributedtask/webhooks/prupdated?api-version
- Accept untrusted SSL certificates:**
- Basic authentication username:**
- Basic authentication password:**

At the bottom are buttons for Previous, Next, Test, Finish, and Cancel.



Project Settings

k8sDemo

General

[Overview](#)[Teams](#)[Permissions](#)[Notifications](#)[Service hooks](#)[Dashboards](#)

Boards

[Project configuration](#)[Team configuration](#)[GitHub connections](#)

Pipelines

[Agent pools](#)[Parallel jobs](#)[Settings](#)[Test management](#)[Release retention](#)[Service connections](#)[XAML build services](#)

Repos

[Repositories](#)

Service connections

[Filter by keywords](#)[4taksDemoAcr](#)[4tecture \(MPN\)\(3\)](#)

(8)

[4tectureregistry](#)[aksDemoCanary-dev-aksDemo-aks-dev-1606772543573](#)[Azure DevOps 4tecture-demo](#)[Microsoft Azure Sponsorship](#)[PREventsConnection](#)[ProductsFeed](#)

New service connection

Choose a service or connection type

 Search connection types Azure Classic Azure Repos/Team Foundation Server Azure Resource Manager Azure Service Bus Bitbucket Cloud Chef Docker Host Docker Registry Generic GitHub GitHub Enterprise Server Incoming WebHook Jenkins Jira Kubernetes



Project Settings

k8sDemo

General

Overview

Teams

Permissions

Notifications

Service hooks

Dashboards

Boards

Project configuration

Team configuration

GitHub connections

Pipelines

Agent pools

Parallel jobs

Settings

Test management

Release retention

Service connections

XAML build services

Repos

Repositories

Service connections

Filter by keywords

4tectureDemoAcr

4tecture (MPN) (3)

(8)

4tectureregistry

aksDemoCanary-dev-aksDemo-aks-dev-1606772543573

Azure DevOps 4tecture-demo

Microsoft Azure Sponsorship

PREventsConnection

ProductsFeed

New Incoming WebHook service connection

Authentication

WebHook Name

PRUpdated

Name of the WebHook

Secret (optional)

Optional secret for the webhook. WebHook service will use this secret to calculate the payload checksum

Http Header (optional)

Http header name on which checksum will be sent

Details

Service connection name

PREventsConnection

Description (optional)

Security

 Grant access permission to all pipelines[Learn more](#)[Troubleshoot](#)[Back](#)[Save](#)

Project Settings

k8sDemo

General

Overview

Teams

Permissions

Notifications

Service hooks

Dashboards

Boards

Project configuration

Team configuration

GitHub connections

Pipelines

Agent pools

Parallel jobs

Settings

Test management

Release retention

Service connections

XAML build services

Repos

Repositories

Service Hooks

Integrate with your favorite services by notifying them when events happen in your project.

EDIT SERVICE HOOKS SUBSCRIPTION

Trigger

Select an event to trigger on and configure any filters.

Trigger on this type of event

Pull request updated

Remember that selected events are visible to users of the target service, even if they don't have permission to view the related artifact.

FILTERS

Repository optional

DevFun

Target branch optional

[Any]

Change optional

Status changed

Requested by a member of group optional

[Any]

Previous Next Test Finish Cancel

Owner	7 Day Status	State
Marc Müller	0 attempted	Enabled

4ecture-demo / k8sDemo / Settings / Service hooks

Search

Project Settings

k8sDemo

General

Overview

Teams

Permissions

Notifications

Boards

Project configuration

Team configuration

Github connections

Pipelines

Agent pools

Parallel jobs

Settings

Test management

Release retention

Service connections

XAML build services

Repos

Repositories

Service Hooks

Integrate with your favorite services by notifying them when events happen in your project.

EDIT SERVICE HOOKS SUBSCRIPTION

Action

Select and configure the action to perform.

Perform this action

Post via HTTP

Owner: Marc Müller

7 Day Status: 0 attempted

State: Enabled

https://dev.azure.com/4ecture-demo/_apis/public/distributedtask/webhooks/prupdated?api-version=6.0-preview

including any authentication headers
[more about Webhooks](#)

SETTINGS

URL: https://dev.azure.com/4ecture-demo/_apis/public/distributedtask/webhooks/prupdated?api-version=6.0-preview ✓

Accept untrusted SSL certificates

Basic authentication username (optional)

Basic authentication password (optional)

Previous Next Test Finish Cancel

The screenshot shows the 'Service Hooks' section of the 'Project Settings' for a project named 'k8sDemo'. The left sidebar lists various project settings like General, Boards, Pipelines, and Repos. The 'Service hooks' option is selected and highlighted in blue. A modal window titled 'EDIT SERVICE HOOKS SUBSCRIPTION' is open, showing configuration options for a subscription to 'https://dev.azure.com/4lecture-demo/_apis/public/distributeddt'. The modal includes fields for accepting untrusted SSL certificates, basic authentication credentials, HTTP headers, resource details, messages to send, and detailed messages. It also specifies a 'RESOURCE VERSION' of '1.0'. At the bottom of the modal are buttons for 'Previous', 'Next', 'Test', 'Finish', and 'Cancel'.

Project Settings

k8sDemo

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards

- Project configuration
- Team configuration
- GitHub connections

Pipelines

- Agent pools
- Parallel jobs
- Settings
- Test management
- Release retention
- Service connections
- XAML build services

Repos

- Repositories

Service Hooks

Integrate with your favorite services by notifying them when events happen in your project.

EDIT SERVICE HOOKS SUBSCRIPTION

https://dev.azure.com/4lecture-demo/_apis/public/distributeddt ✓

Accept untrusted SSL certificates ⓘ

Basic authentication username ⓘ optional

Basic authentication password ⓘ optional

HTTP headers ⓘ optional

Resource details to send ⓘ optional

All

Messages to send ⓘ optional

All

Detailed messages to send ⓘ optional

All

RESOURCE VERSION

1.0

Previous Next Test Finish Cancel

k8sDemo

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Releases

Library

Task groups

Deployment groups

Test Plans

Artifacts

← DevFun PR Cleanup

Variables

Run



master DevFun / .azure-pipelines/devfun-prcleanup.yml

Show assistant

```
1 trigger: none
2
3 pool:
4   vmImage: ubuntu-latest
5
6 resources:
7   # https://dev.azure.com/4ecture-demo/_apis/public/distributedtask/webhooks/prupdated?api-version=6.0-preview
8   webhooks:
9     - webhook: PRUpdated ..... ## Webhook alias
10       connection: PREventsConnection ..... ## Incoming webhook service connection
11       filters:
12         - path: eventType
13           value: git.pullrequest.updated
14         - path: publisherId
15           value: tfs
16         - path: resource.repository.name
17           value: DevFun
18
19 variables:
20   Validate
21   - template: templates/common_variables.yml
22   - template: templates/common_variables-cd.yml
23
24 stages:
25   - stage: CleanupPr
26     jobs:
27       - deployment: CleanupPrEnvironment
28         displayName: 'Cleanup PR environment'
29         environment: devfuncleanuppr
30         condition: or(eq('${{ parameters.PRUpdated.resource.status }}', 'completed'), eq('${{ parameters.PRUpdated.resource.status }}', 'abandoned'))
31         strategy:
32           runOnce:
33             deploy:
34               steps:
35                 - download: none
```

Incoming
WebHook as
resource with
trigger

Additional conditions for
dynamic payload evaluation

k8sDemo

← DevFun PR Cleanup

Variables

Run



master DevFun / .azure-pipelines/devfun-prcleanup.yml

Show assistant

```
  strategy:
    runOnce:
      deploy:
        steps:
          - download: none
          - pwsh:
              $prId = "${{ parameters.PRUpdated.resource.pullRequestId }}"
              $dbname = "devfun-pr$($prId)"
              Write-Host "##vso[task.setvariable variable=dbname;]$dbname"
              Write-Host "##vso[task.setvariable variable=k8sNamespace;]pr-$($prId)"
              Write-Host "##vso[task.setvariable variable=managedIdentityApi;]devfunapi-pr-$($prId)"
              Write-Host "##vso[task.setvariable variable=deploymentManagedIdentityApi;]devfunapi-deployment-pr-$($prId)"
              displayName: "Calculate PR related variables"
```

Calculate PR specific variables

```
  validate:
    - template: templates/pr_DeleteK8sPrEnvironment.yml
      parameters:
        azureSubscription: '$(AzureSubscription)'
        clusterResourceGroup: '$(AzureResourceGroup)'
        clusterName: '$(KubernetesCluster)'
        clusterNamespace: '$(k8sNamespace)'
        continueOnError: true
```

Delete deployment

```
  validate:
    - template: templates/sql_DeleteAzureSqlDatabase.yml
      parameters:
        AzureSubscription: '$(AzureSubscription)'
        ResourceGroup: '$(AzureResourceGroup)'
        AzureSqlName: '$(dbservername)'
        DbName: '$(dbname)'
        continueOnError: true
```

Delete database

```
  validate:
    - template: templates/az_DeleteManagedIdentity.yml
      parameters:
        AzureSubscription: '$(AzureSubscription)'
        ManagedIdentity: '$(managedIdentityApi)'
        ManagedIdentityResourceGroup: '$(ManagedIdentityResourceGroup)'
        continueOnError: true
```

Delete Identities





Azure Deployments

Conclusion

4tecture®
empower your software solutions

Conclusion

- Everything is Code – IaC, CaC
- Put Secrets in KeyVault / Secure Files
- Use Managed Identities → no credentials!
- Deployment agents in target environment



Azure Deployments

Q & A

4tecture®
empower your software solutions

Thank you for your attention!

If you have any questions do not hesitate to contact us:

4ecture GmbH
Industriestrasse 25
CH-8604 Volketswil
www.4ecture.ch

Marc Müller
Principal Consultant

www.powerofdevops.com

A close-up photograph of several hands reaching towards a central wooden puzzle piece. The puzzle piece is light-colored wood with a dark green and red section. The hands belong to different people, suggesting collaboration. The background is blurred.

4 tecture[©]
empower your software solutions