



# Azure DevOps Pipelines vs GitHub Actions

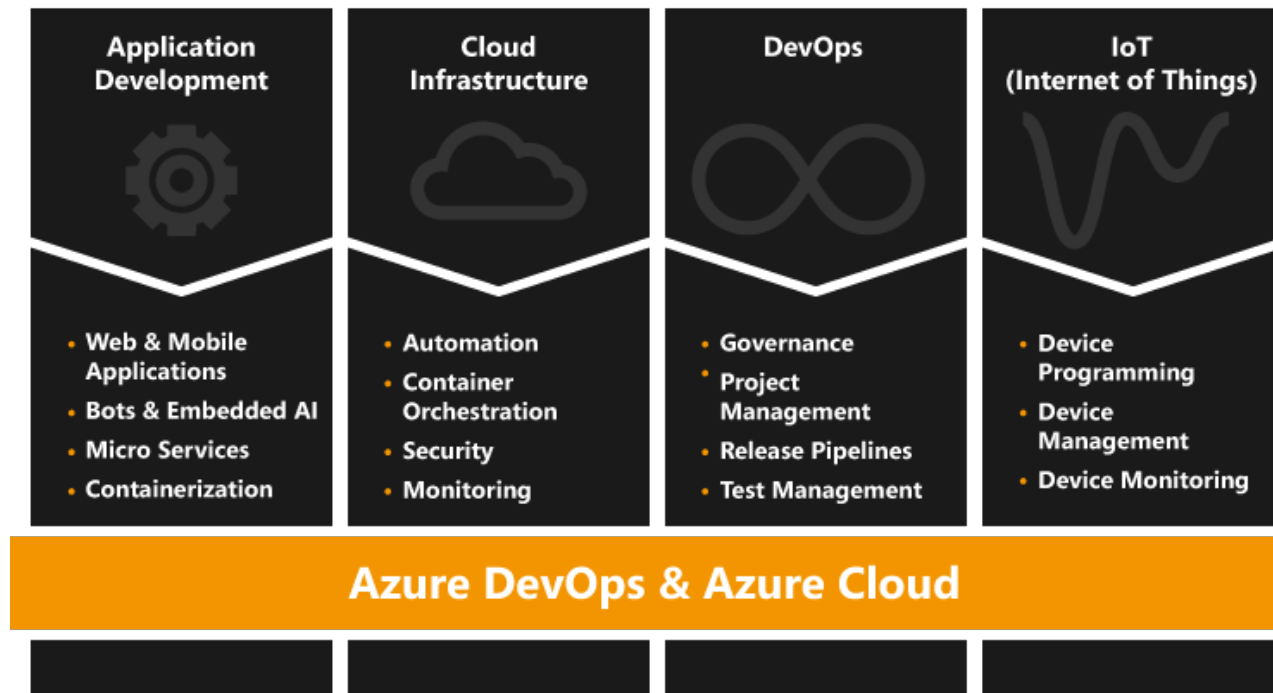
## Wo geht die Reise hin?

09.12.2019



# Über white duck...

- Dienstleister für Cloud- und Software-Engineering für Unternehmen aus allen Branchen
- Spezialisiert auf Application Development, Cloud Infrastructure, DevOps und IoT
- Konzeption, Entwicklung und Betrieb von individuellen Cloud Anwendungen (SaaS-, Web-, Mobile- und IoT)
- Gegründet 2012 mit Sitz in Rosenheim, derzeit 20 Mitarbeiter
- Erfahrung aus mehr als 15 Jahren Softwareentwicklung
- Technologie-Fokus: Microsoft Azure Cloud, Azure DevOps, .NET C#, .NET CORE, REST, Angular, TypeScript



Unsere Cloud Partner:



Gold DevOps  
Gold Cloud Platform



CLOUD  
**ECOSYSTEM**  
Mitglied



# Ressourcen zum Meetup



Folien und Ressourcen zum Meetup werden auf GitHub bereitgestellt:  
<https://github.com/whiteducksoftware/azure-meetup-rosenheim>

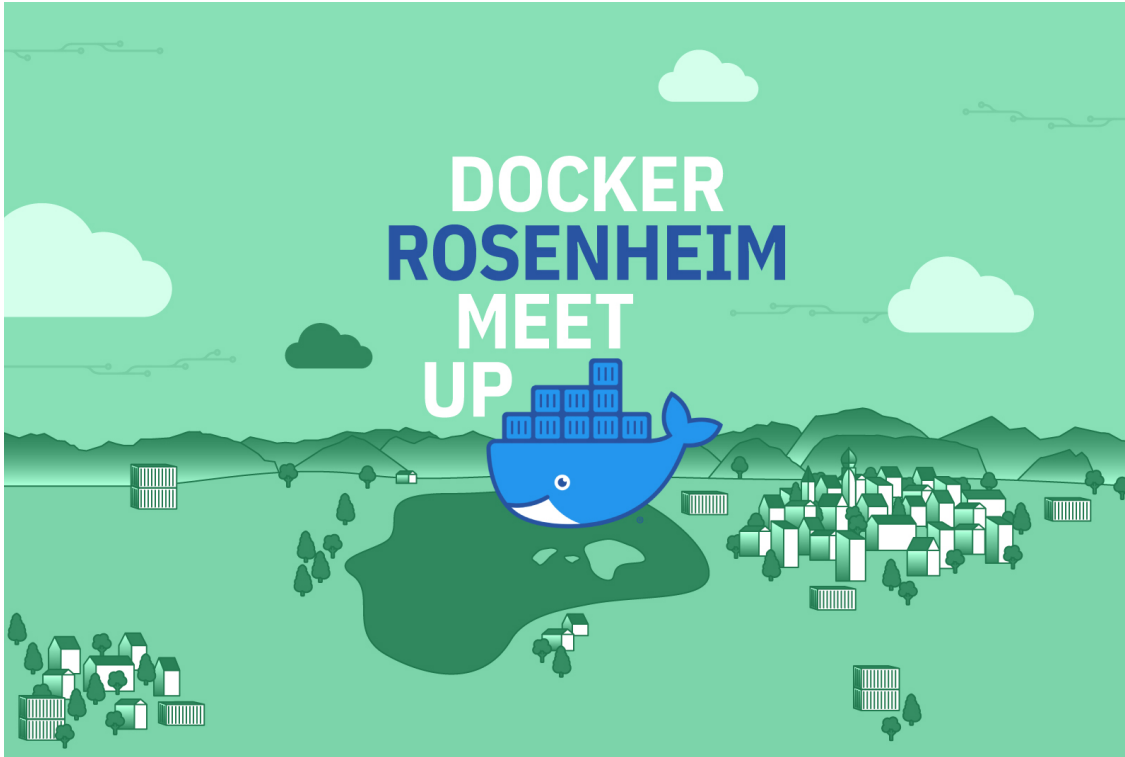


[@AzureMeetup](https://twitter.com/AzureMeetup)



<https://whiteducksoftware.com/azure-rosenheim-meetup>

# Docker Meetup Rosenheim



Hands-on Workshop  
Introduction to Docker for  
Developers

16.12.2019

TH Rosenheim

# **GITHUB ACTIONS 101**

# GitHub Actions

- “fast CI/CD for any OS, any language, and any cloud”
- long private beta, GA since November 13<sup>th</sup>
- Documentation
  - <https://help.github.com/en/actions>
- Forum
  - <https://github.community/t5/GitHub-Actions/bd-p/actions>
- Examples
  - <https://github.com/actions/starter-workflows/tree/master/ci>

# Terminology

- Actions
  - product name
- Step
  - the smallest instance. A command or Action
- Action
  - predefined step which can be used and shared
  - can contain multiple steps
- Job
  - combines multi steps running on the same OS
  - can be started in parallel or in sequence
- Workflow
  - one or multiple jobs including a trigger definition

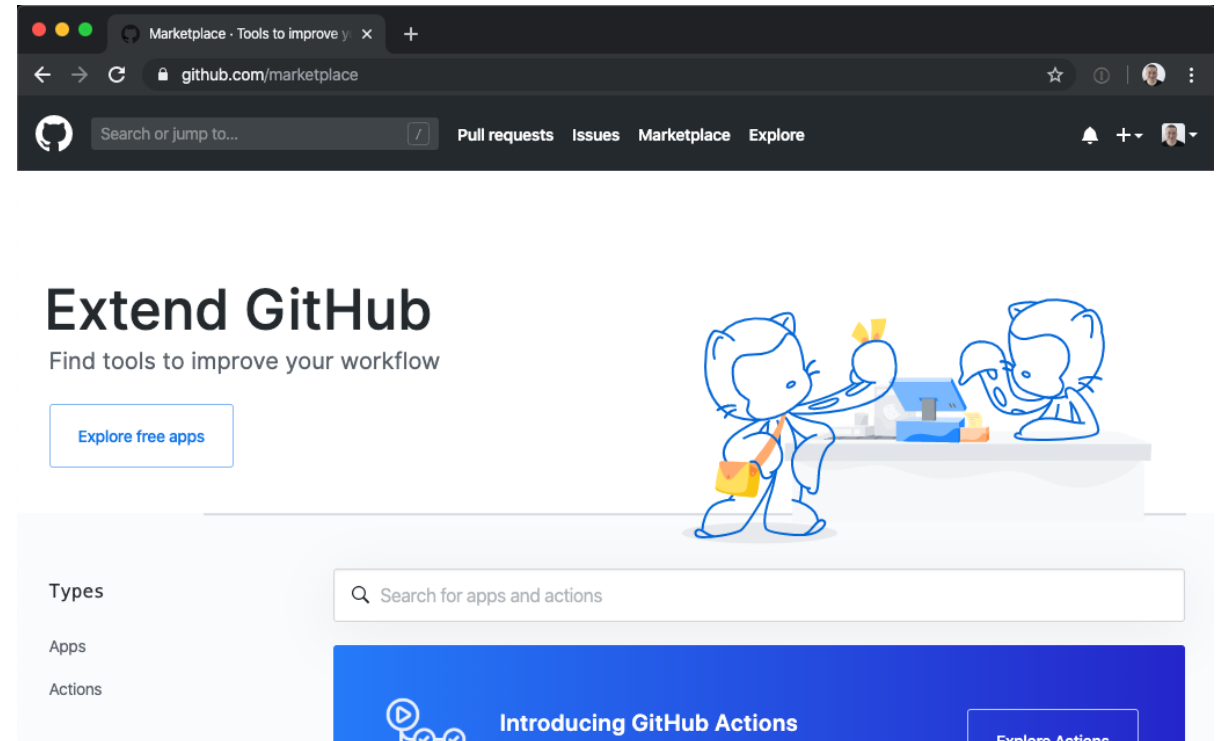
# Workflows

- YAML based manifest
  - code only
  - `.github/workflows/*.yml|yaml`
- can be triggered by **any** GitHub event
  - Wiki updated
  - Label added
  - Milestone closed
  - Pull Request edited
  - match in Commit message
  - ...
- notifications based on mails and/or webhooks



# GitHub Marketplace

- Actions
  - ~ 1500
- Apps
  - integrate third-party apps
  - ~ 150
- <https://github.com/marketplace>



# Usage limits

- 20 concurrent workflows per repository
- 1000 API request per repository and hour
- 6h runtime limits for workflows per repository

GitHub plan	concurrent jobs	concurrent macOS jobs
Free	20	5
Pro	40	5
Team	60	5
Enterprise	180	15

# GitHub-hosted Runner

- Standard\_DS2\_v2 VMs on Azure (2 cores, 7 GB mem)
  - Ubuntu 16.04 LTS
  - Ubuntu 18.04 LTS
  - Windows Server 2016 (removed on December 3)
  - Windows Server 2019
  - macOS 10.15
- list of installed software
  - <https://help.github.com/en/actions/automating-your-workflow-with-github-actions/software-installed-on-github-hosted-runners>

# Self-hosted Runner

- still beta
- no workload/time limitations
- can be added to repositories
  - Organizations will follow
- support for multiple OS
  - Linux: RHEL, CentOS, Fedora, Ubuntu, Mint, openSUSE, SLES
  - Windows: 7, 8.1, 10, 2012, 2016, 2019
  - macOS

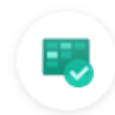
# Connect with Azure DevOps (Apps)

- Azure Pipelines
  - connect DevOps pipelines with GitHub Actions
- Azure Boards
  - Connect DevOps boards with GitHub Actions



Azure Pipelines 

Continuously build, test, and deploy to any platform and cloud



Azure Boards 

Connects Azure Boards with GitHub to plan, track, and discuss work across your teams

# Connect with Azure (Actions)

- by Azure
  - Azure Login
  - Azure WebApp
  - Azure App Service Settings
  - Azure Functions Action
  - Azure SQL Deploy
  - Azure MySQL Deploy
  - Azure Kubernetes set context
  - Azure Pipeline Action
  - Azure CLI Action
  - ...
- by Community
  - Azure Key Vault Action
  - Azure Cosmos DB Emulator
  - Azure Blob Storage Upload
  - Create Azure Blueprint
  - ...

# A Workflow

```
1  on: [push]
2
3  # Environment variables available to all jobs and steps in this workflow
4  env:
5    REGISTRY_NAME: my-registry-name
6    CLUSTER_NAME: my-cluster-name
7    CLUSTER_RESOURCE_GROUP: my-cluster-resource-group
8    NAMESPACE: default
9
10 jobs:
11   build:
12     runs-on: ubuntu-latest
13     steps:
14       - uses: actions/checkout@master
15
16       # Connect to Azure Container registry (ACR)
17       - uses: azure/docker-login@v1
18         with:
19           login-server: ${ env.REGISTRY_NAME }.azurecr.io
20           username: ${ secrets.REGISTRY_USERNAME }
21           password: ${ secrets.REGISTRY_PASSWORD }
22
23       # Container build and push to a Azure Container registry (ACR)
24       - run: |
25         docker build . -t ${ env.REGISTRY_NAME }.azurecr.io/myimage:${ github.sha }
26         docker push ${ env.REGISTRY_NAME }.azurecr.io/myimage:${ github.sha }
```

```
28   # Set the target Azure Kubernetes Service (AKS) cluster.
29   - uses: azure/aks-set-context@v1
30     with:
31       creds: '${ secrets.AZURE_CREDENTIALS }'
32       cluster-name: ${ env.CLUSTER_NAME }
33       resource-group: ${ env.CLUSTER_RESOURCE_GROUP }
34
35   # Create namespace if doesn't exist
36   - run: |
37     kubectl create namespace ${ env.NAMESPACE } --dry-run -o json | kubectl apply -f -
38
39   # Create imagepullsecret for Azure Container registry (ACR)
40   - uses: azure/k8s-create-secret@v1
41     with:
42       container-registry-url: ${ env.REGISTRY_NAME }.azurecr.io
43       container-registry-username: ${ secrets.REGISTRY_USERNAME }
44       container-registry-password: ${ secrets.REGISTRY_PASSWORD }
45       secret-name: ${ env.REGISTRY_NAME }-registry-connection
46       namespace: ${ env.NAMESPACE }
47
48   # Deploy app to AKS
49   - uses: azure/k8s-deploy@v1
50     with:
51       manifests: |
52         manifests/deployment.yml
53         manifests/service.yml
54       images: |
55         ${ env.REGISTRY_NAME }.azurecr.io/myimage:${ github.sha }
56       imagepullsecrets: |
57         ${ env.REGISTRY_NAME }-registry-connection
58       namespace: ${ env.NAMESPACE }
```

# more features

- Matrix builds
  - build and test multiple versions in parallel
  - example: .NET Core 2.2 and 3.0 on Windows and Linux
- live logs
- integrates with GitHub Package Registry
- suggested workflows to lower barriers (GitHub UI)



# Something missing? Build your own Action

- can be private
  - .github/actions/\*.yaml|yml
- or shared
  - GitHub Marketplace
- can be based on
  - Docker Image (Linux only)
  - Node.js
- more details
  - <https://help.github.com/en/actions/automating-your-workflow-with-github-actions/about-actions>

GitHub Actions in action 😊

# **GITHUB ACTIONS DEMO**

# Connect to Azure

## 1. Create a Service Principal

```
az ad sp create-for-rbac --name "myApp" --role contributor --scopes  
"/subscriptions/{id}/resourceGroups/{rg}" --sdk-auth
```

## 2. Store the output to a GitHub Secret variable (AZURE\_CREDENTIALS)

## 3. Login using the Azure Login Action

```
- uses: azure/login@v1
```

```
  with:
```

```
    creds: ${ secrets.AZURE_CREDENTIALS }
```

# GitHub Action to trigger a run in Azure pipelines

```
- uses: Azure/pipelines@v1
```

```
  with:
```

```
    azure-devops-project-url: 'https://dev.azure.com/org/project'
```

```
    azure-pipeline-name: 'pipeline-name'
```

```
    azure-devops-token: '${{ secrets.AZURE_DEVOPS_TOKEN }}'
```

more details: <https://github.com/Azure/pipelines>

# **AZURE DEVOPS PIPELINES**

# **ACTIONS VS DEVOPS PIPELINES**