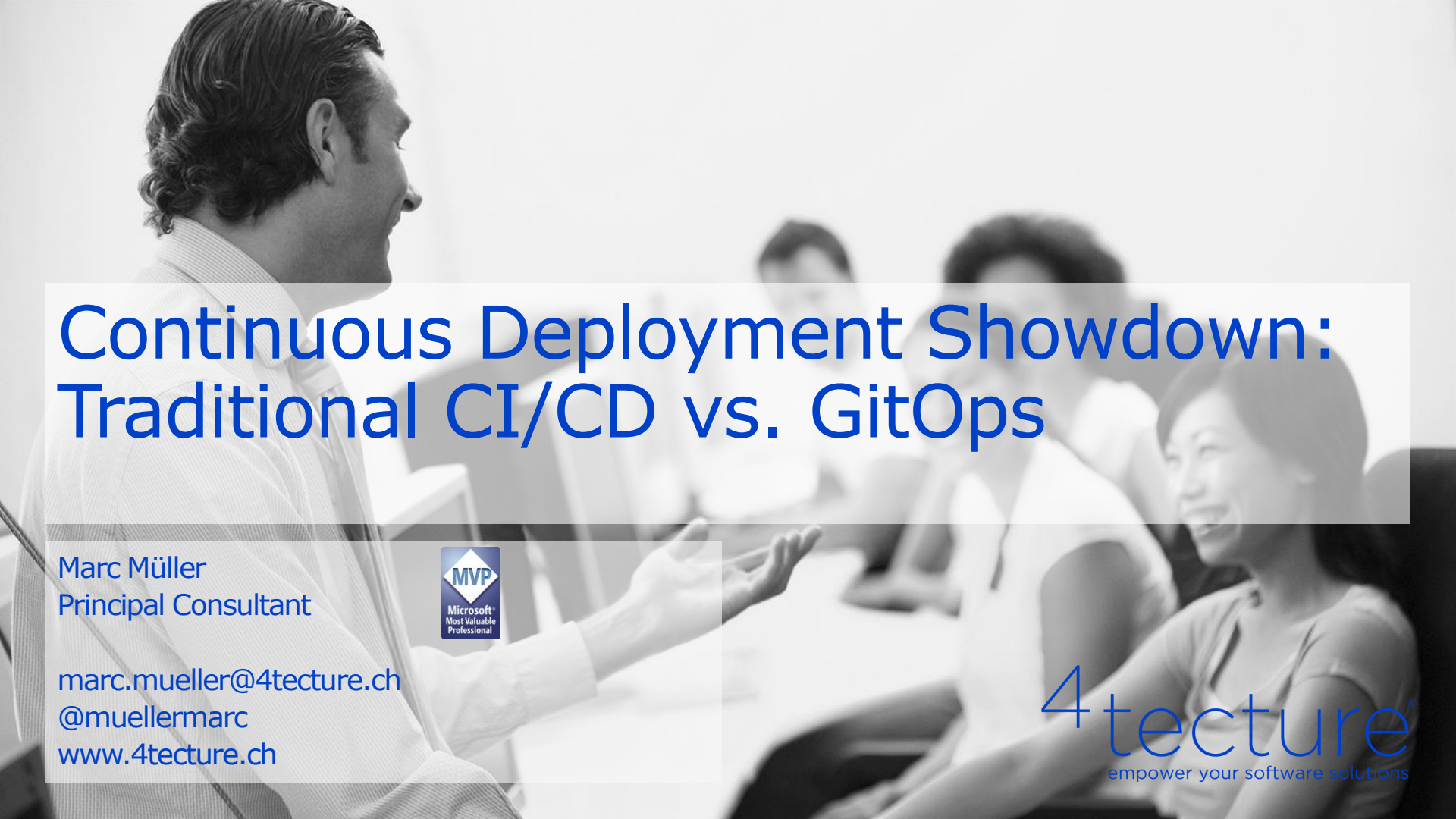


Continuous Deployment Showdown: Traditional CI/CD vs. GitOps

Marc Müller
Principal Consultant

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

MVP Microsoft Most Valuable Professional

4tecture
empower your software solutions

About me:

Marc Müller
Principal Consultant
@muellermarc

Microsoft Most Valuable Professional

4tecture©
empower your software solutions

Our Products:

Multi-Tenant OpenID Connect Identity Provider

ProAuth

www.proauth.net

Enterprise Application Framework for .NET

ReaFx

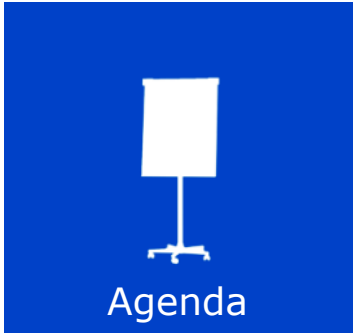
www.reafx.net

Slide Download

https://www.4tecture.ch/events/dwx24cicdgitops

Agenda

- Intro
- Setup
- Implementing the Deployment
- Staging
- Automatic Updates
- Cluster Setup with Azure Pipelines
- Conclusion



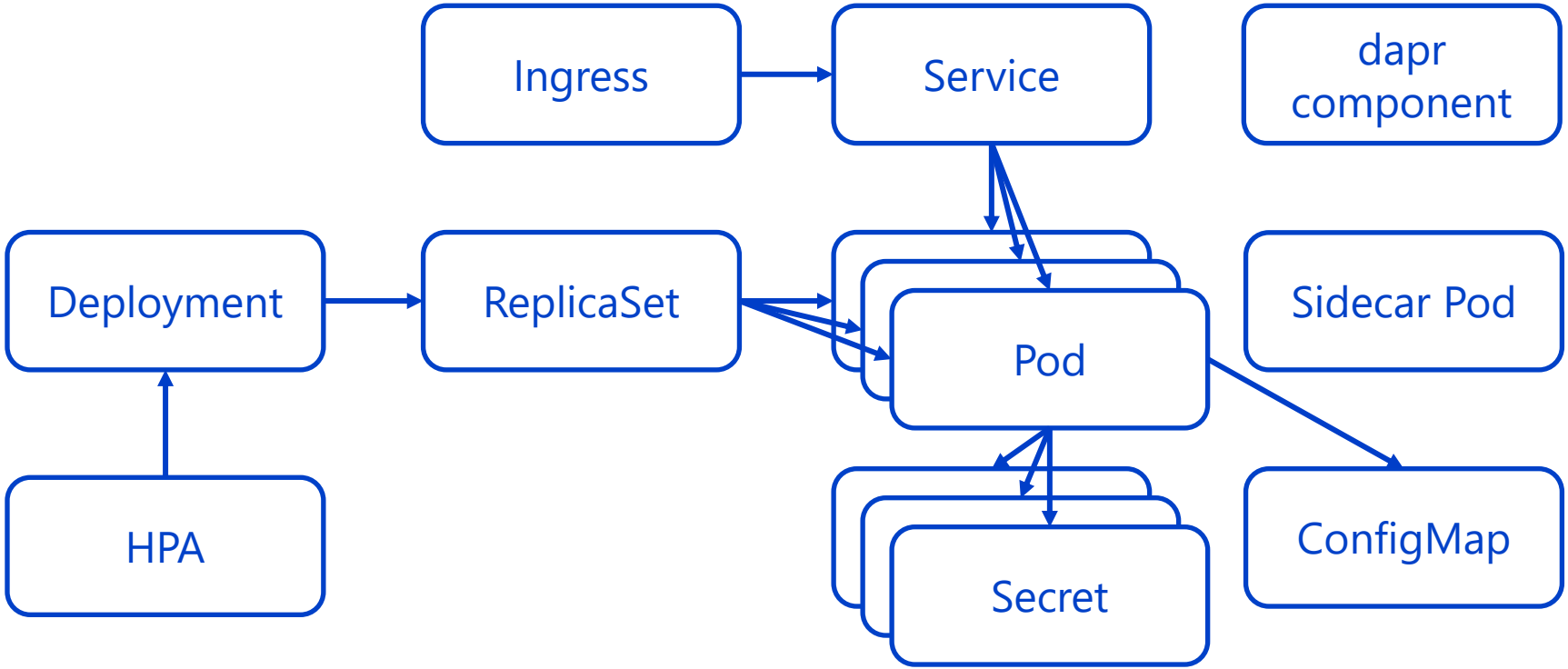
Intro

4tecture
empower your software solutions

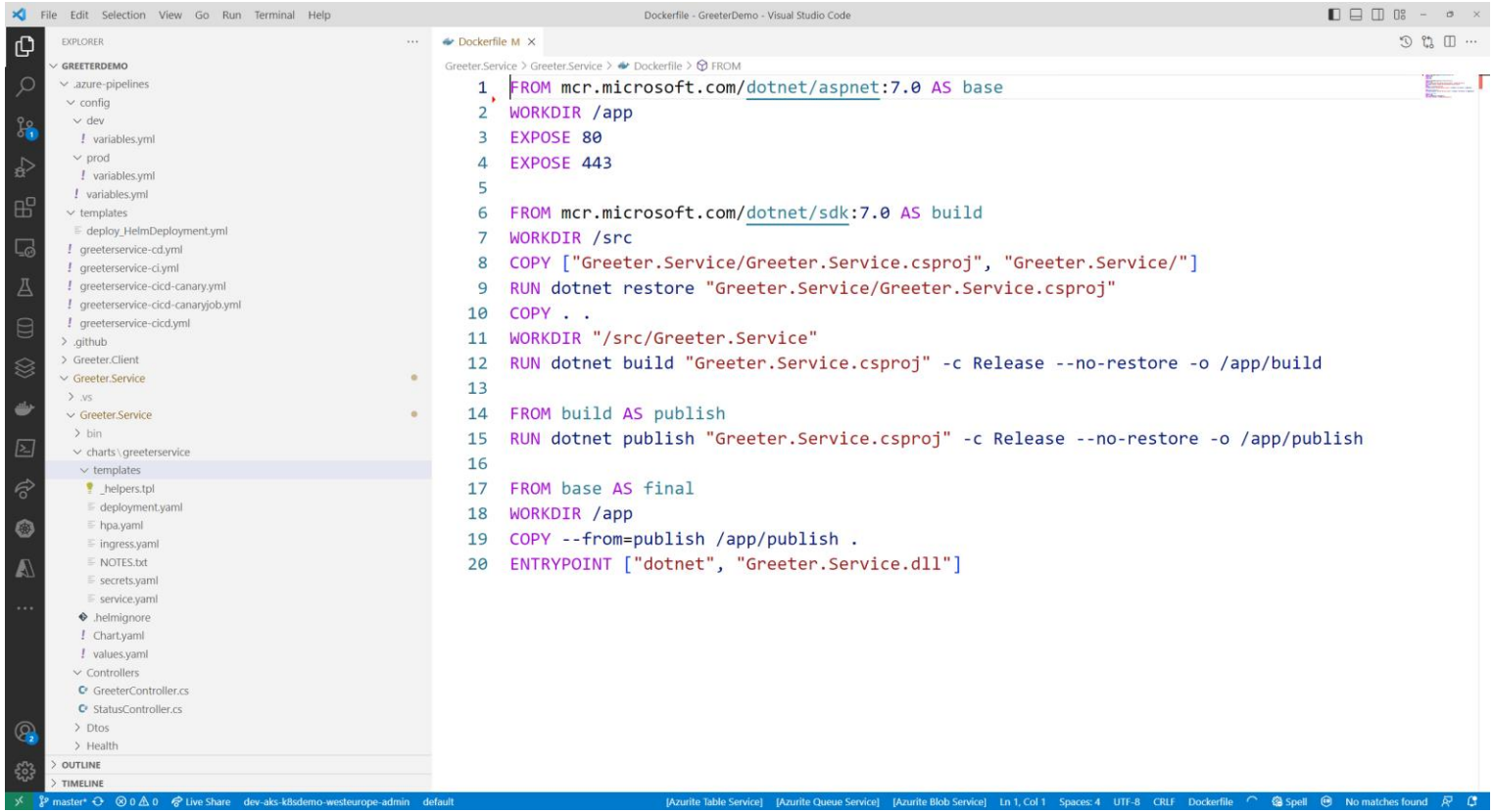
Our playground…

- ASP.NET Core Application
- Dockerized
- Kubernetes deployment environment

Kubernetes Deployment



We focus on deployment ...this is the build



Kustomize

- Standalone tool
  - for customizing Kubernetes objects
  - by using a declarative configuration language
- Built-in in kubectl since 1.14
- kustomization.yaml → processes resources
- Bases and overlays
- Patches, ConfigMaps and Secrets, Variable substitution

Helm

- Package Manger for Kubernetes
- Define, install and upgrade Kubernetes applications
- Configuration based on «Charts»

Source: https://helm.sh/

Helm Features

- Manage Complexity
  - Charts describe even the most complex apps
  - Provide repeatable application installation
  - Serve as a single point of authority
- Easy Updates
  - Take the pain out of updates
  - In-place upgrades
  - Custom hooks.
- Simple Sharing
  - Versioning
  - Easy to share, and host on public or private servers
- Rollbacks
  - Use helm rollback to roll back to an older version of a release with ease.

Source: https://helm.sh/

KEEP
CALM
AND

GIT COMMIT
GIT PUSH ORIGIN MAIN

Source: https://imgflip.com/i/7xal0f

GitOps

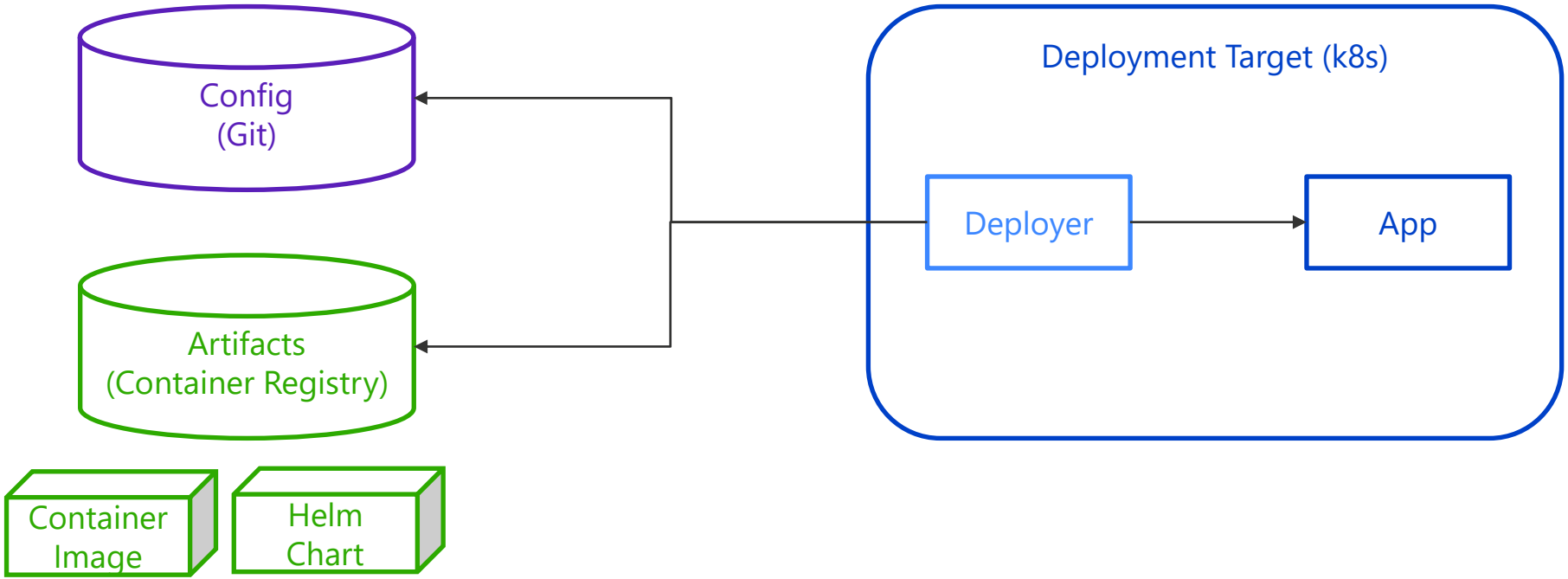
- Declarative Configuration:
  - System state is described in a declarative manner, usually using files like YAML.
- Version-Controlled System State:
  - Desired system state is stored in a version control system, typically Git.
  - Git serves as the single source of truth.
- Automated Delivery:
  - Changes in the Git repository trigger automatic system updates.
- Software Agents Ensure System Convergence:
  - Tools continuously monitor and ensure the actual system state matches the desired state in the Git repository.
  - Any discrepancies lead the system to self-correct to match the desired state.
- Infrastructure as Code (IaC):
  - Infrastructure setup and configuration are defined and version-controlled as code.
- Enhanced Visibility and Traceability:
  - All changes are tracked, auditable, and can be rolled back using Git's capabilities.
- Reduced Manual Intervention:
  - Direct changes to production are discouraged.
  - Changes are made via pull requests or commits to the repository.

GitOps

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

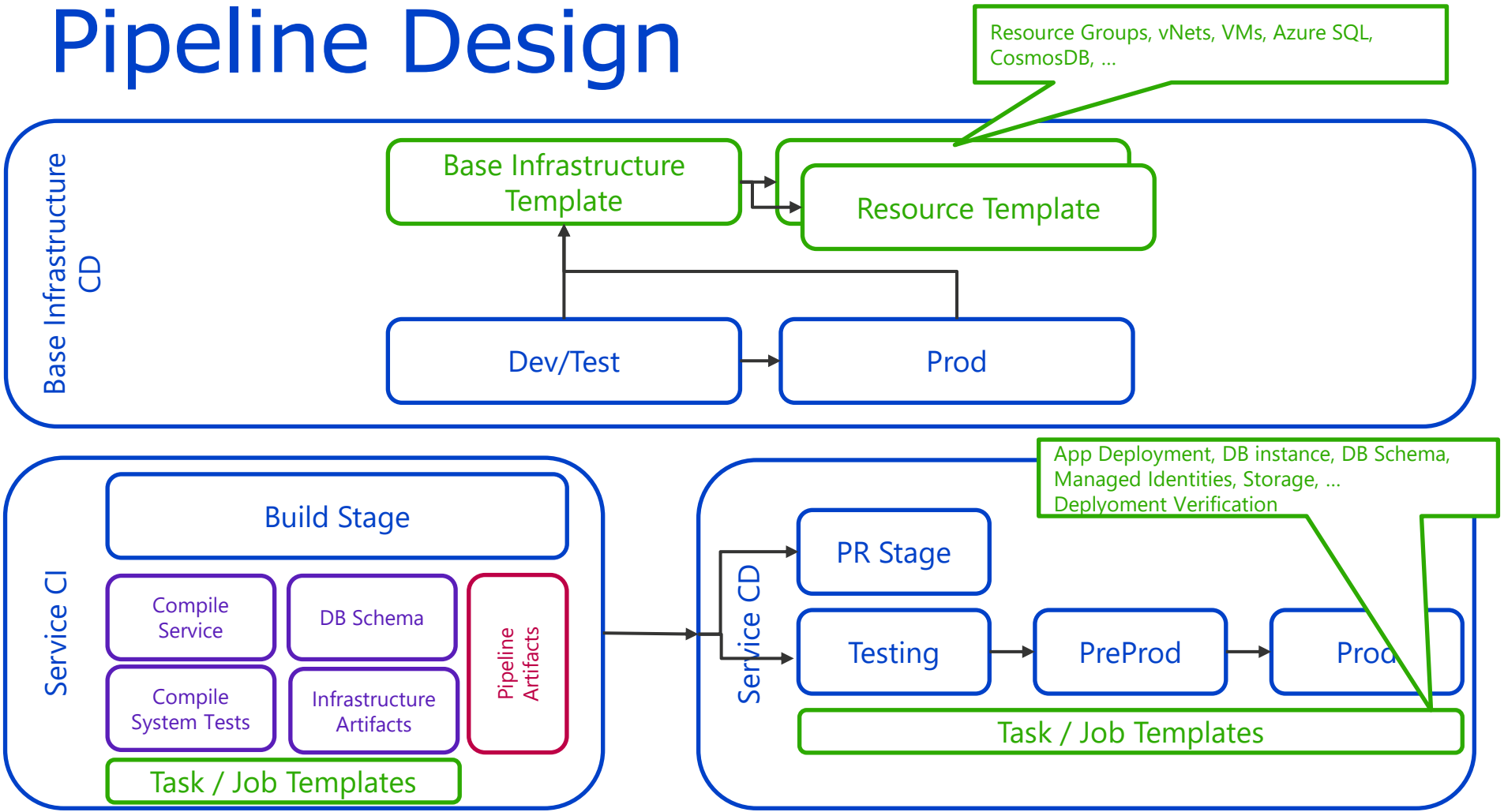

Setup
4tecture
empower your software solutions

General Setup

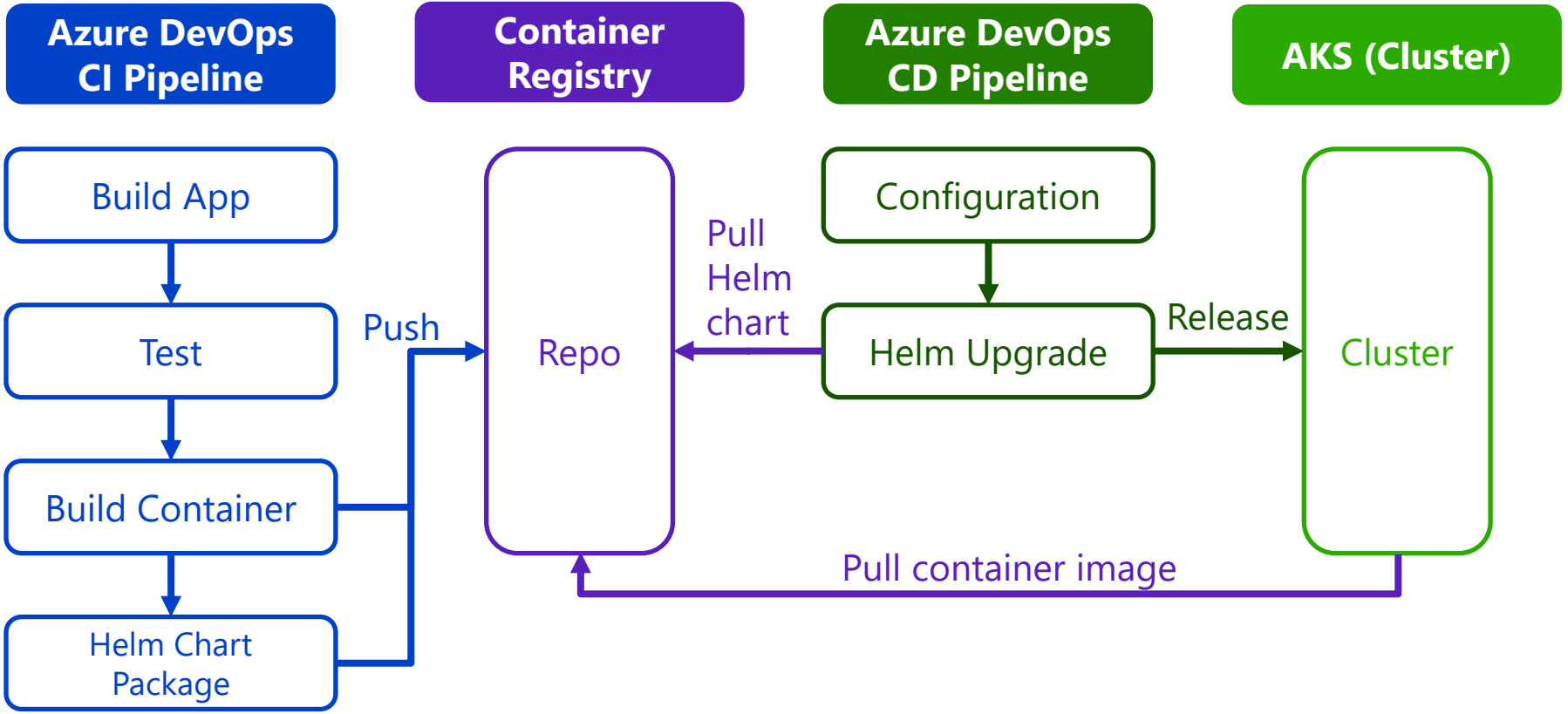


Azure Pipelines

Pipeline Design

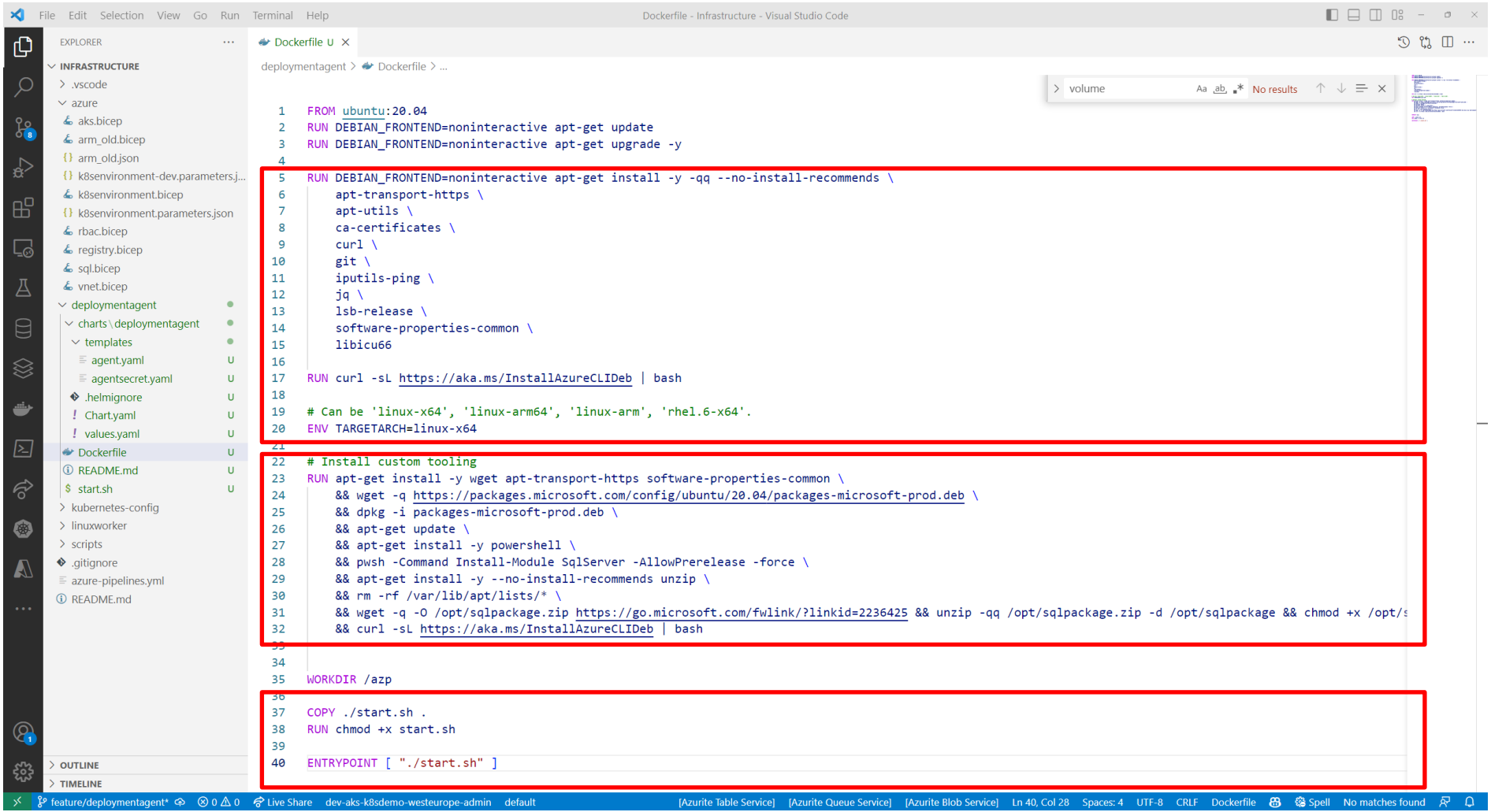


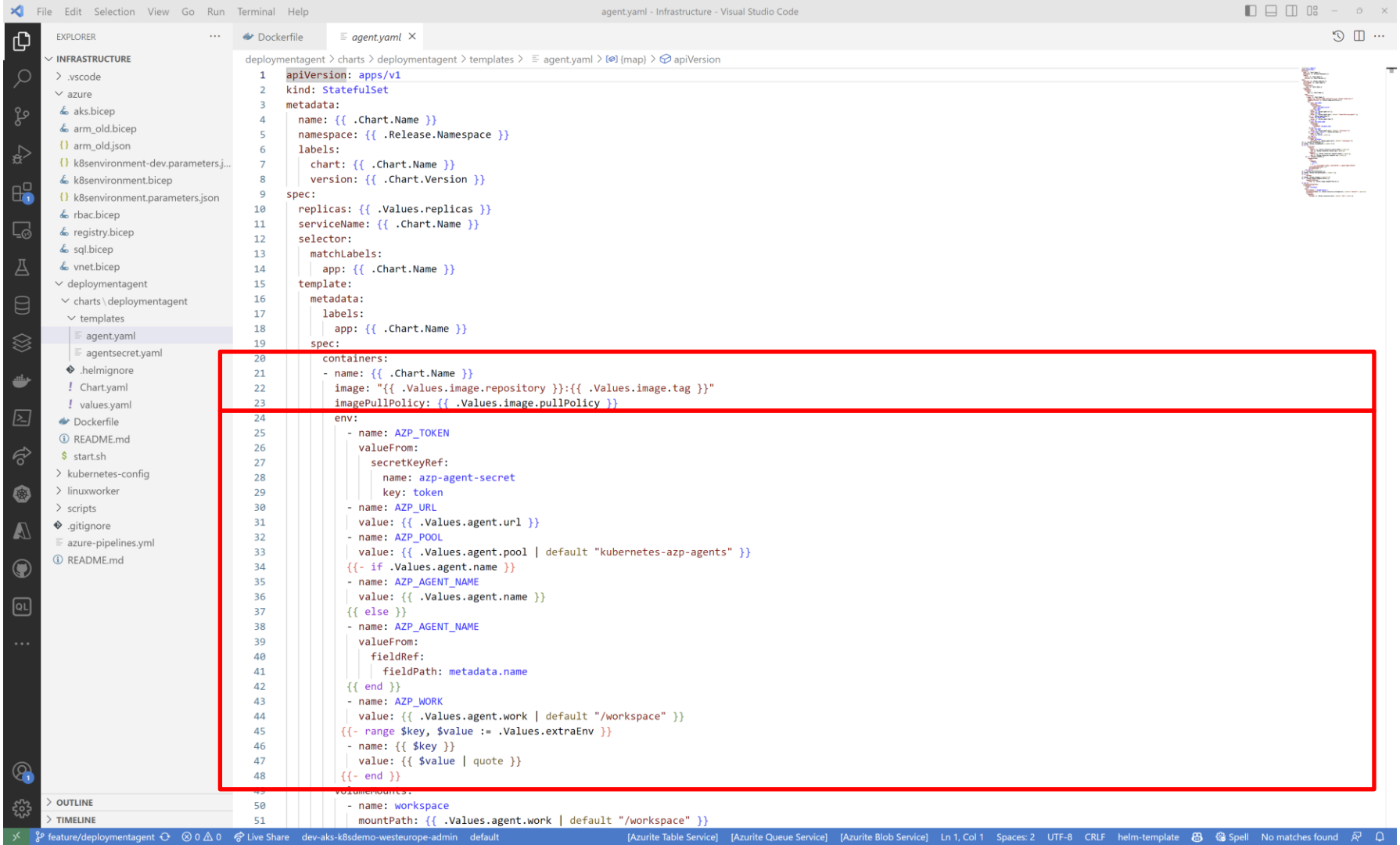
Azure DevOps & AKS CI / CD

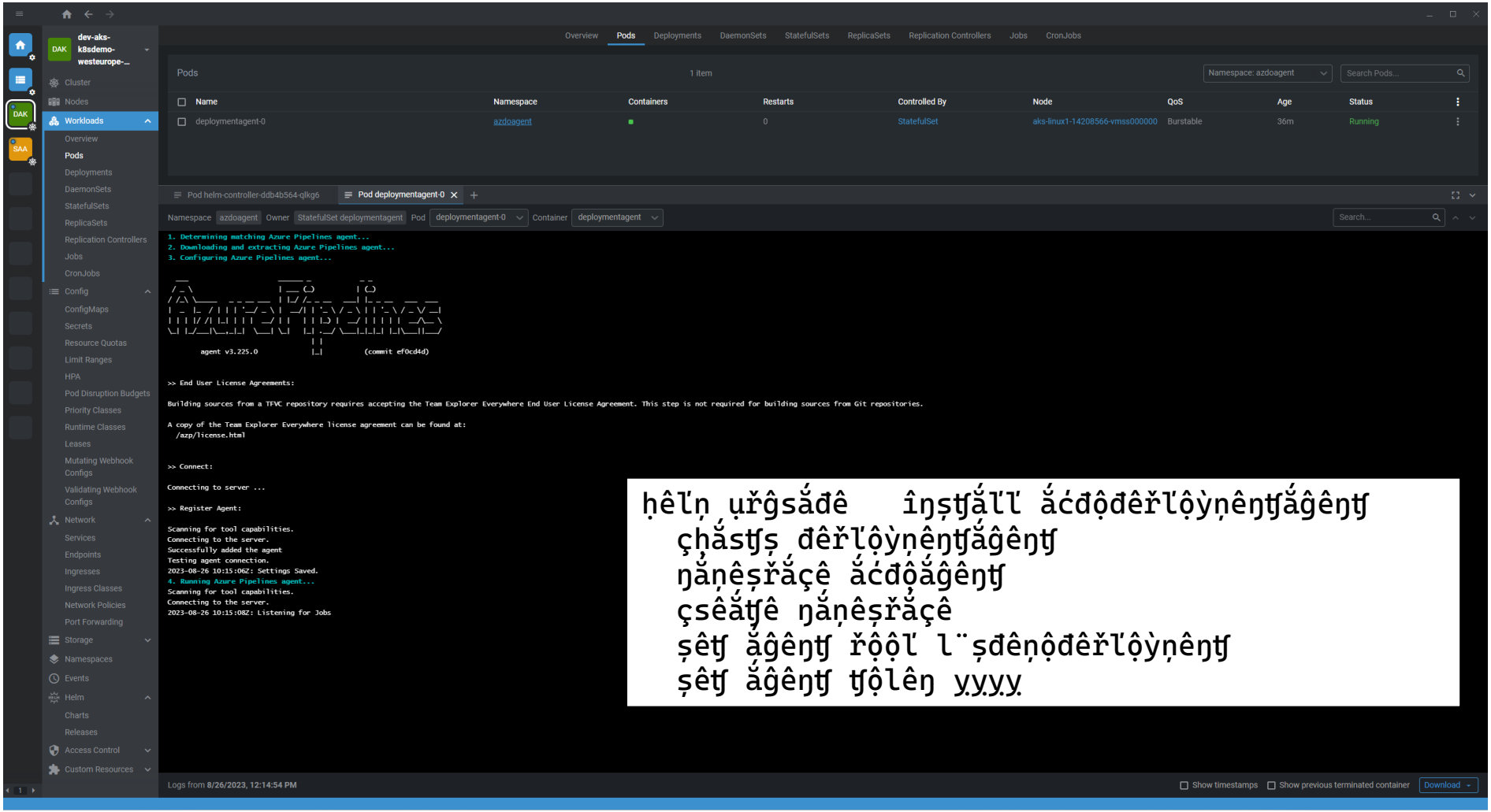


Configuration as Code

- 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)


File Edit Selection View Go Run Terminal Help
Dockerfile - Infrastructure - Visual Studio Code
EXPLORER
INFRASTRUCTURE
.vscode
azure
aks.bicep
arm_old.bicep
arm_old.json
k8senvironment-dev.parameters.j...
k8senvironment.bicep
k8senvironment.parameters.json
rbac.bicep
registry.bicep
sql.bicep
vnet.bicep
deploymentagent
charts \ deploymentagent
templates
agent.yaml U
agentsecret.yaml U
.helmignore U
Chart.yaml U
values.yaml U
Dockerfile U
README.md U
start.sh U
kubernetes-config
linuxworker
scripts
.gitignore
azure-pipelines.yml
README.md
OUTLINE
TIMELINE
Dockerfile U
deploymentagent > Dockerfile > ...
volume
No results
1 FROM ubuntu:20.04
2 RUN DEBIAN_FRONTEND=noninteractive apt-get update
3 RUN DEBIAN_FRONTEND=noninteractive apt-get upgrade -y
4
5 RUN DEBIAN_FRONTEND=noninteractive apt-get install -y -qq --no-install-recommends \
6 apt-transport-https \
7 apt-utils \
8 ca-certificates \
9 curl \
10 git \
11 iputils-ping \
12 jq \
13 lsb-release \
14 software-properties-common \
15 libicu66
16
17 RUN curl -sL https://aka.ms/InstallAzureCLIDeb | bash
18
19 # Can be 'linux-x64', 'linux-arm64', 'linux-arm', 'rhel.6-x64'.
20 ENV TARGETARCH=linux-x64
21
22 # Install custom tooling
23 RUN apt-get install -y wget apt-transport-https software-properties-common \
24 && wget -q https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb \
25 && dpkg -i packages-microsoft-prod.deb \
26 && apt-get update \
27 && apt-get install -y powershell \
28 && pwsh -Command Install-Module SqlServer -AllowPrerelease -force \
29 && apt-get install -y --no-install-recommends unzip \
30 && rm -rf /var/lib/apt/lists/* \
31 && wget -q -O /opt/sqlpackage.zip https://go.microsoft.com/fwlink/?linkid=2236425 && unzip -qq /opt/sqlpackage.zip -d /opt/sqlpackage && chmod +x /opt/s
32 && curl -sL https://aka.ms/InstallAzureCLIDeb | bash
33
34
35 WORKDIR /azp
36
37 COPY ./start.sh .
38 RUN chmod +x start.sh
39
40 ENTRYPOINT ["./start.sh"]
feature/deploymentagent* 0 0 Live Share dev-aks-k8sdemo-westeurope-admin default
[Azurite Table Service] [Azurite Queue Service] [Azurite Blob Service] Ln 40, Col 28 Spaces: 4 UTF-8 CRLF Dockerfile Spell No matches found


File Edit Selection View Go Run Terminal Help
agent.yaml - Infrastructure - Visual Studio Code
EXPLORER
INFRASTRUCTURE
.vscode
azure
aks.bicep
arm_old.bicep
arm_old.json
k8senvironment-dev.parameters.j...
k8senvironment.bicep
k8senvironment.parameters.json
rbac.bicep
registry.bicep
sql.bicep
vnet.bicep
deploymentagent
charts \ deploymentagent
templates
agent.yaml
agentsecret.yaml
.helmignore
Chart.yaml
values.yaml
Dockerfile
README.md
start.sh
kubernetes-config
linuxworker
scripts
.gitignore
azure-pipelines.yml
README.md
OUTLINE
TIMELINE
Dockerfile
agent.yaml
deploymentagent > charts > deploymentagent > templates > agent.yaml > {map} > apiVersion
1 apiVersion: apps/v1
2 kind: StatefulSet
3 metadata:
4 name: {{ .Chart.Name }}
5 namespace: {{ .Release.Namespace }}
6 labels:
7 chart: {{ .Chart.Name }}
8 version: {{ .Chart.Version }}
9 spec:
10 replicas: {{ .Values.replicas }}
11 serviceName: {{ .Chart.Name }}
12 selector:
13 matchLabels:
14 app: {{ .Chart.Name }}
15 template:
16 metadata:
17 labels:
18 app: {{ .Chart.Name }}
19 spec:
20 containers:
21 - name: {{ .Chart.Name }}
22 image: "{{ .Values.image.repository }}:{{ .Values.image.tag }}"
23 imagePullPolicy: {{ .Values.image.pullPolicy }}
24 env:
25 - name: AZP_TOKEN
26 valueFrom:
27 secretKeyRef:
28 name: azp-agent-secret
29 key: token
30 - name: AZP_URL
31 value: {{ .Values.agent.url }}
32 - name: AZP_POOL
33 value: {{ .Values.agent.pool | default "kubernetes-azp-agents" }}
34 {{- if .Values.agent.name }}
35 - name: AZP_AGENT_NAME
36 value: {{ .Values.agent.name }}
37 {{ else }}
38 - name: AZP_AGENT_NAME
39 valueFrom:
40 fieldRef:
41 fieldPath: metadata.name
42 {{ end }}
43 - name: AZP_WORK
44 value: {{ .Values.agent.work | default "/workspace" }}
45 {{- range $key, $value := .Values.extraEnv }}
46 - name: {{ $key }}
47 value: {{ $value | quote }}
48 {{- end }}
50 - name: workspace
51 mountPath: {{ .Values.agent.work | default "/workspace" }}
feature/deploymentagent
Live Share
dev-aks-k8sdemo-westeurope-admin
default
[Azurite Table Service]
[Azurite Queue Service]
[Azurite Blob Service]
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
helm-template
Spell
No matches found

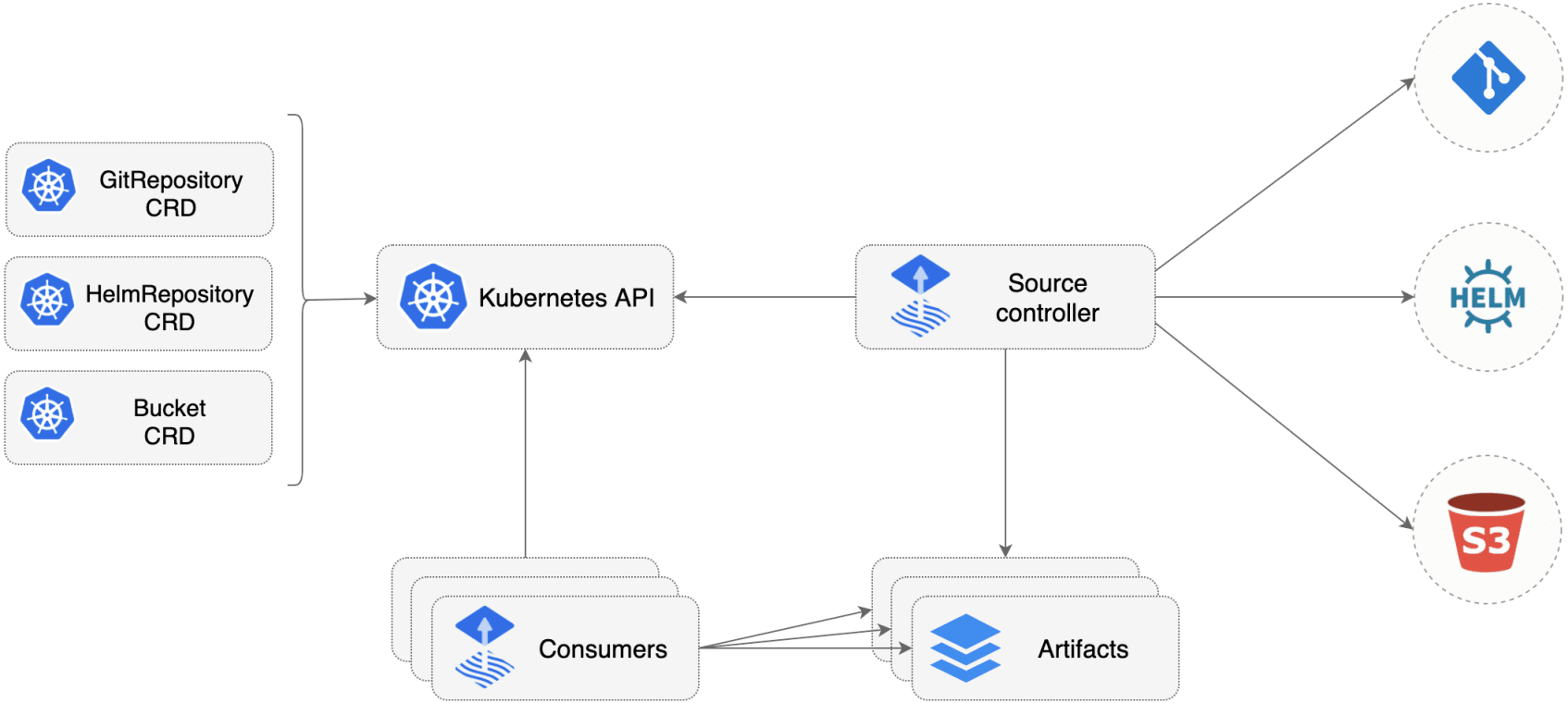

Overview
Pods
Deployments
DaemonSets
StatefulSets
ReplicaSets
Replication Controllers
Jobs
CronJobs
dev-aks-k8sdemo-westeurope-...
Cluster
Nodes
Workloads
Overview
Pods
Deployments
DaemonSets
StatefulSets
ReplicaSets
Replication Controllers
Jobs
CronJobs
Config
ConfigMaps
Secrets
Resource Quotas
Limit Ranges
HPA
Pod Disruption Budgets
Priority Classes
Runtime Classes
Leases
Mutating Webhook Configs
Validating Webhook Configs
Network
Services
Endpoints
Ingresses
Ingress Classes
Network Policies
Port Forwarding
Storage
Namespaces
Events
Helm
Charts
Releases
Access Control
Custom Resources
Pods
1 item
Namespace: azdoagent
Search Pods...
Name
Namespace
Containers
Restarts
Controlled By
Node
QoS
Age
Status
deploymentagent-0
azdoagent
0
StatefulSet
aks-linux1-14208566-vmss000000
Burstable
36m
Running
Pod helm-controller-ddb4b564-qlkg6
Pod deploymentagent-0
Namespace azdoagent
Owner StatefulSet deploymentagent
Pod deploymentagent-0
Container deploymentagent
Search...
1. Determining matching Azure Pipelines agent...
2. Downloading and extracting Azure Pipelines agent...
3. Configuring Azure Pipelines agent...
agent v3.225.0
(commit ef0cd4d)
>> End User License Agreements:
Building sources from a TFVC repository requires accepting the Team Explorer Everywhere End User License Agreement. This step is not required for building sources from Git repositories.
A copy of the Team Explorer Everywhere license agreement can be found at:
/azp/license.html
>> Connect:
Connecting to server ...
>> Register Agent:
Scanning for tool capabilities.
Connecting to server.
Successfully added the agent
Testing agent connection.
2023-08-26 10:15:06Z: Settings Saved.
4. Running Azure Pipelines agent...
Scanning for tool capabilities.
Connecting to server.
2023-08-26 10:15:08Z: Listening for Jobs
Logs from 8/26/2023, 12:14:54 PM
Show timestamps
Show previous terminated container
Download

Flux

FluxCD

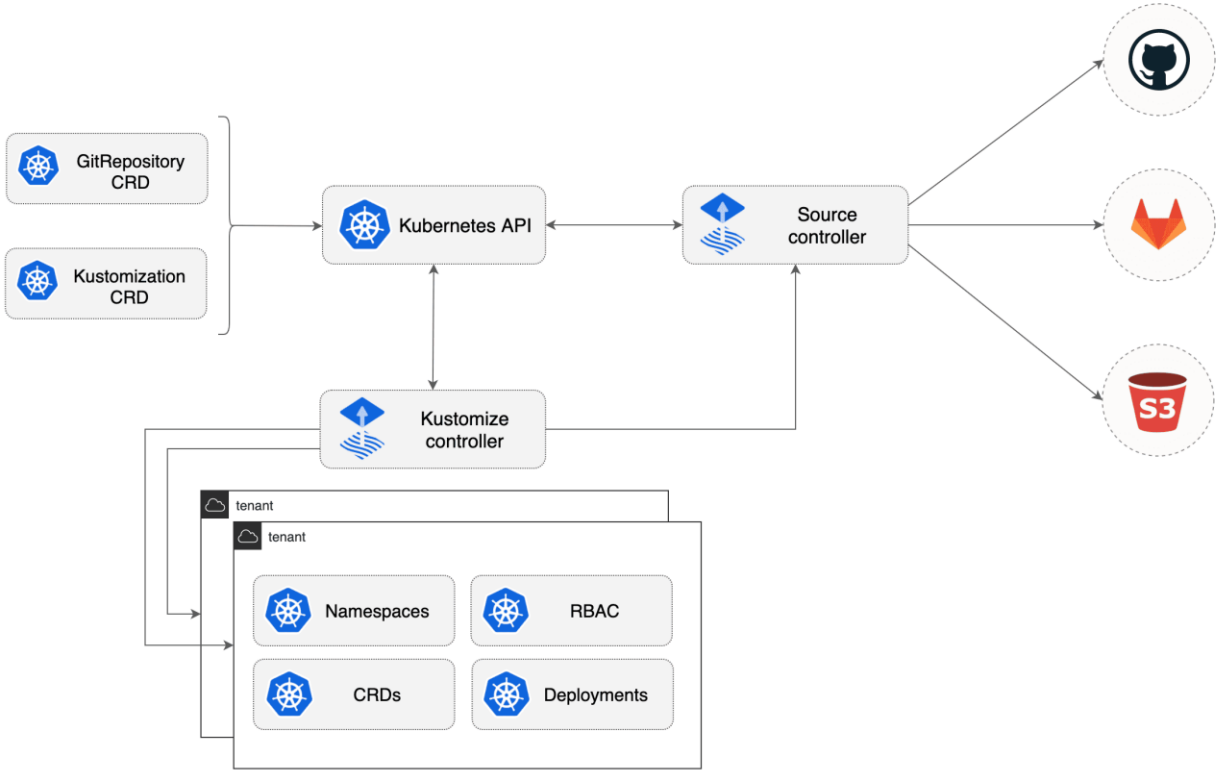
- Open-source GitOps toolkit for Kubernetes
- Part of the CNCF
- Monitors Git repositories
- Automatically applies the changes to the cluster

Flux Source Controller



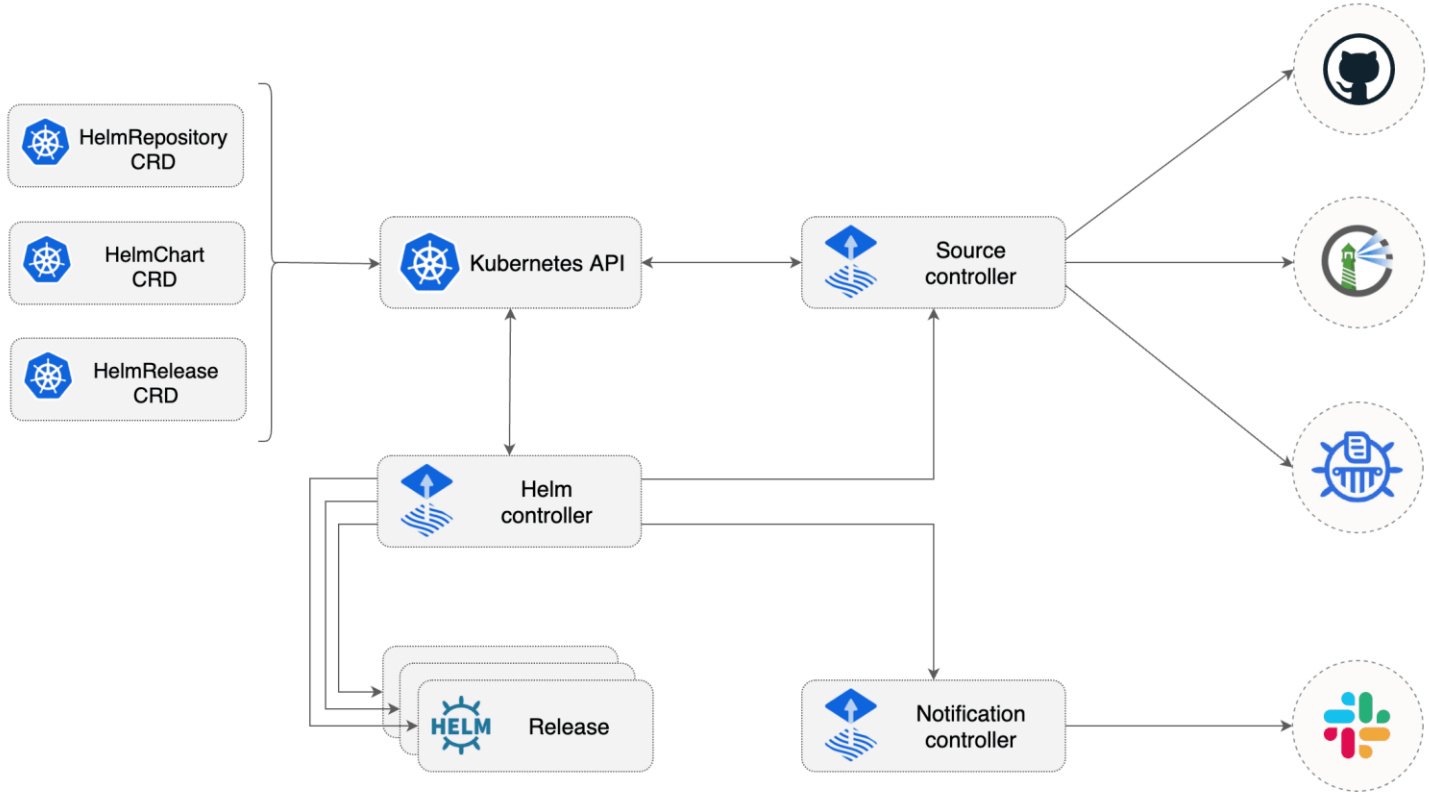
Source: https://fluxcd.io/flux/components/source/

Flux Kustomize Controller



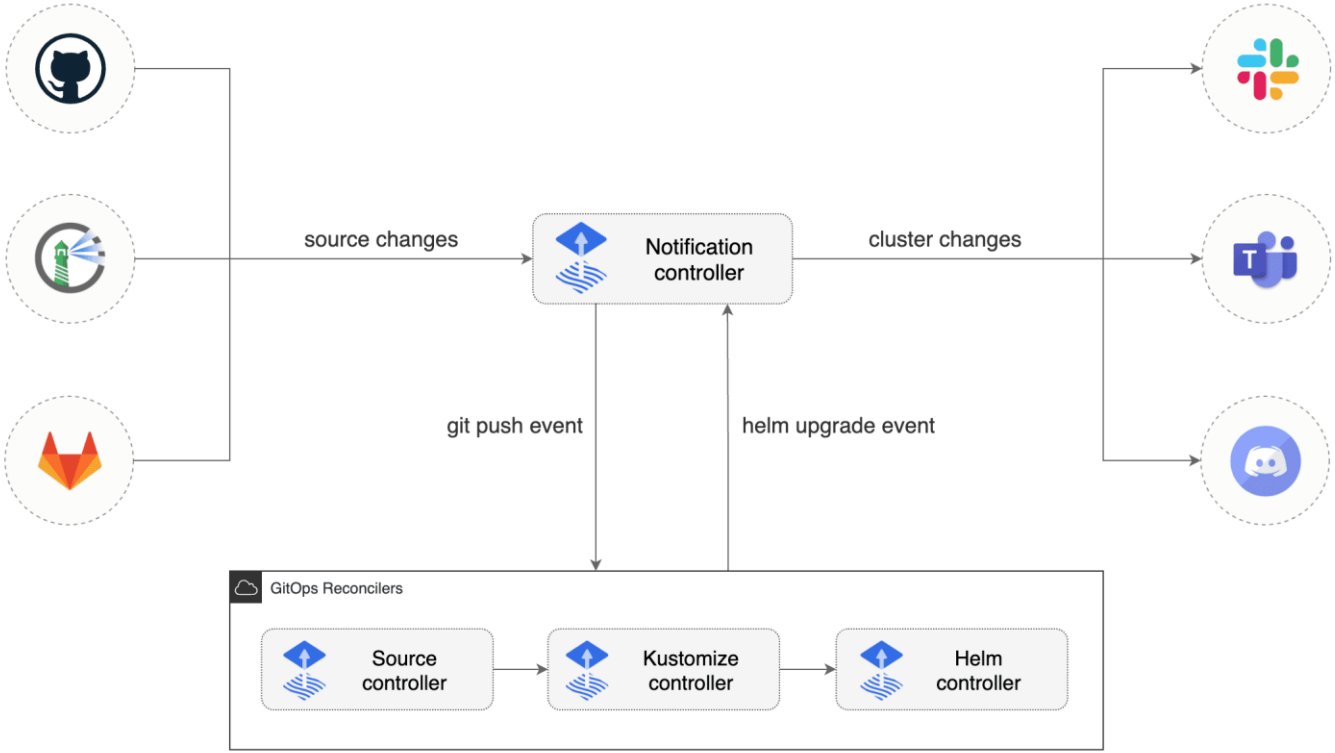
Source: https://fluxcd.io/flux/components/kustomize/

Flux Helm Controller



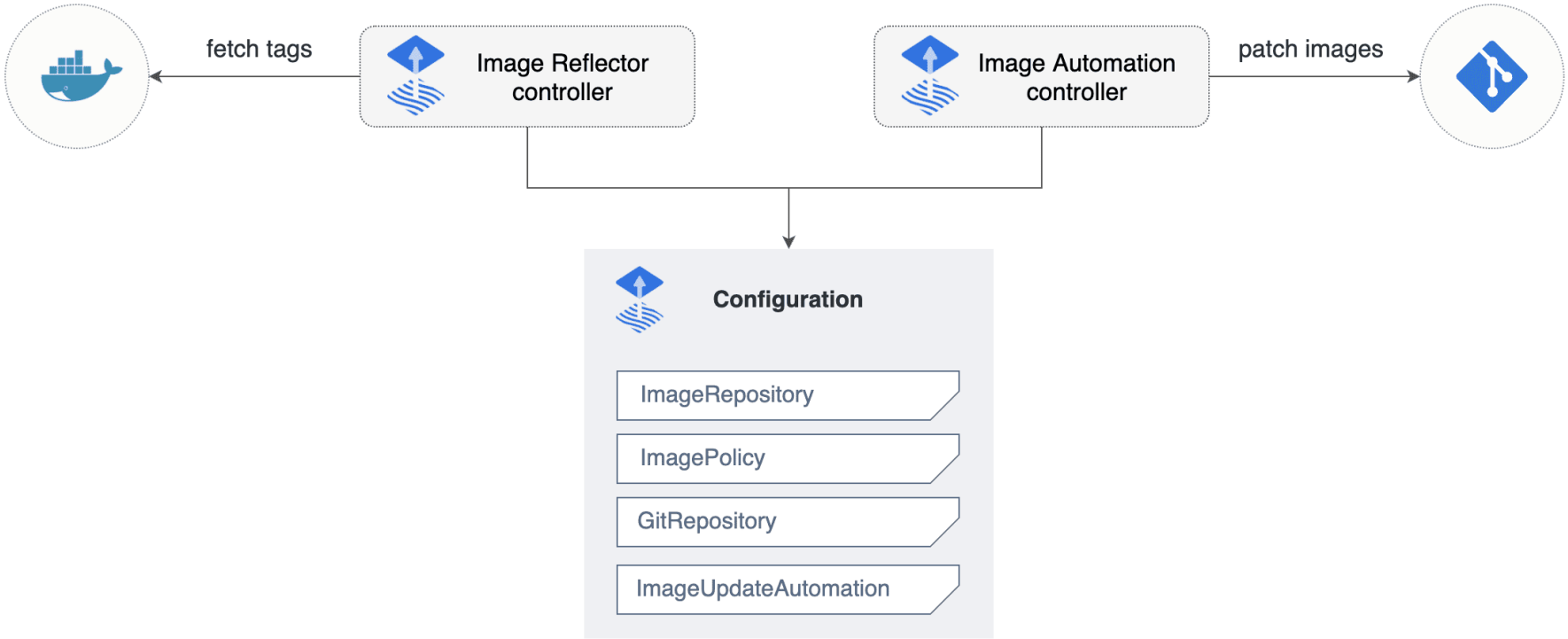
Source: https://fluxcd.io/flux/components/helm/

Flux Notification Controller



Source: https://fluxcd.io/flux/components/notification/

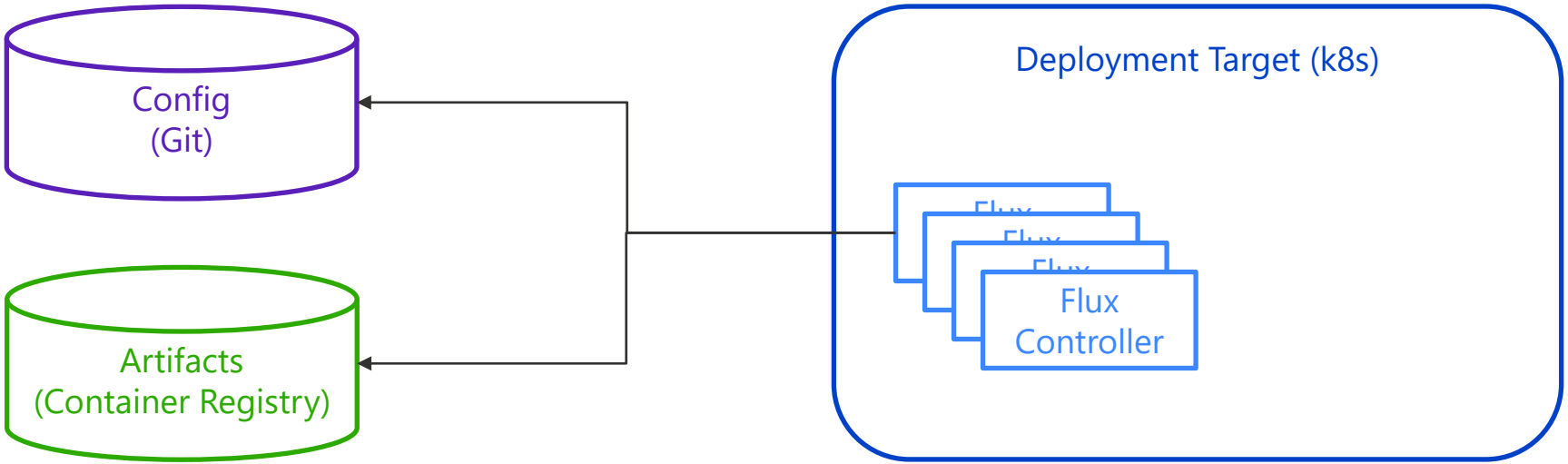
Flux Image Update Automation



Source: https://fluxcd.io/flux/components/image/

Setup Flux

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

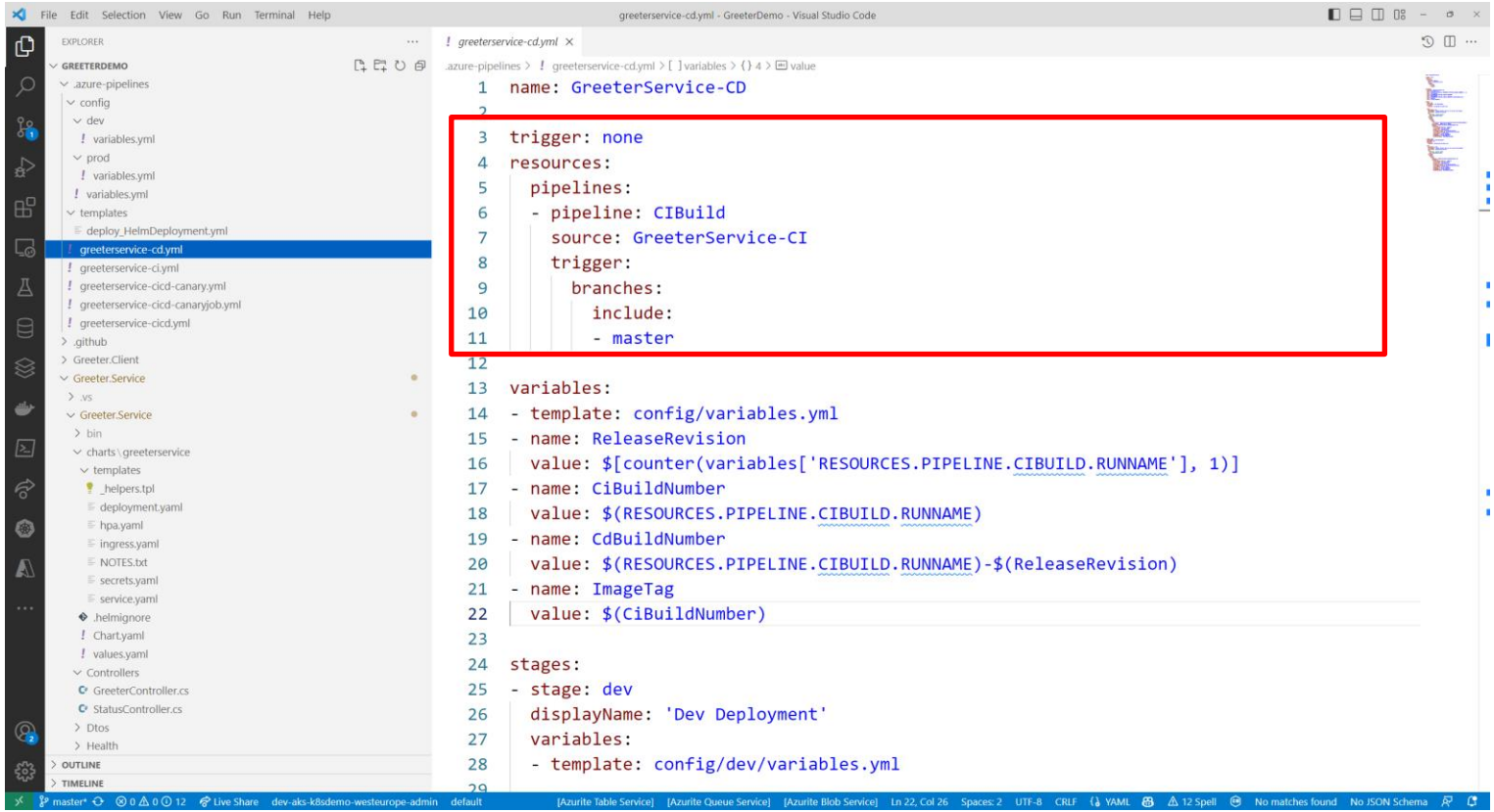


Implementing the Deployment

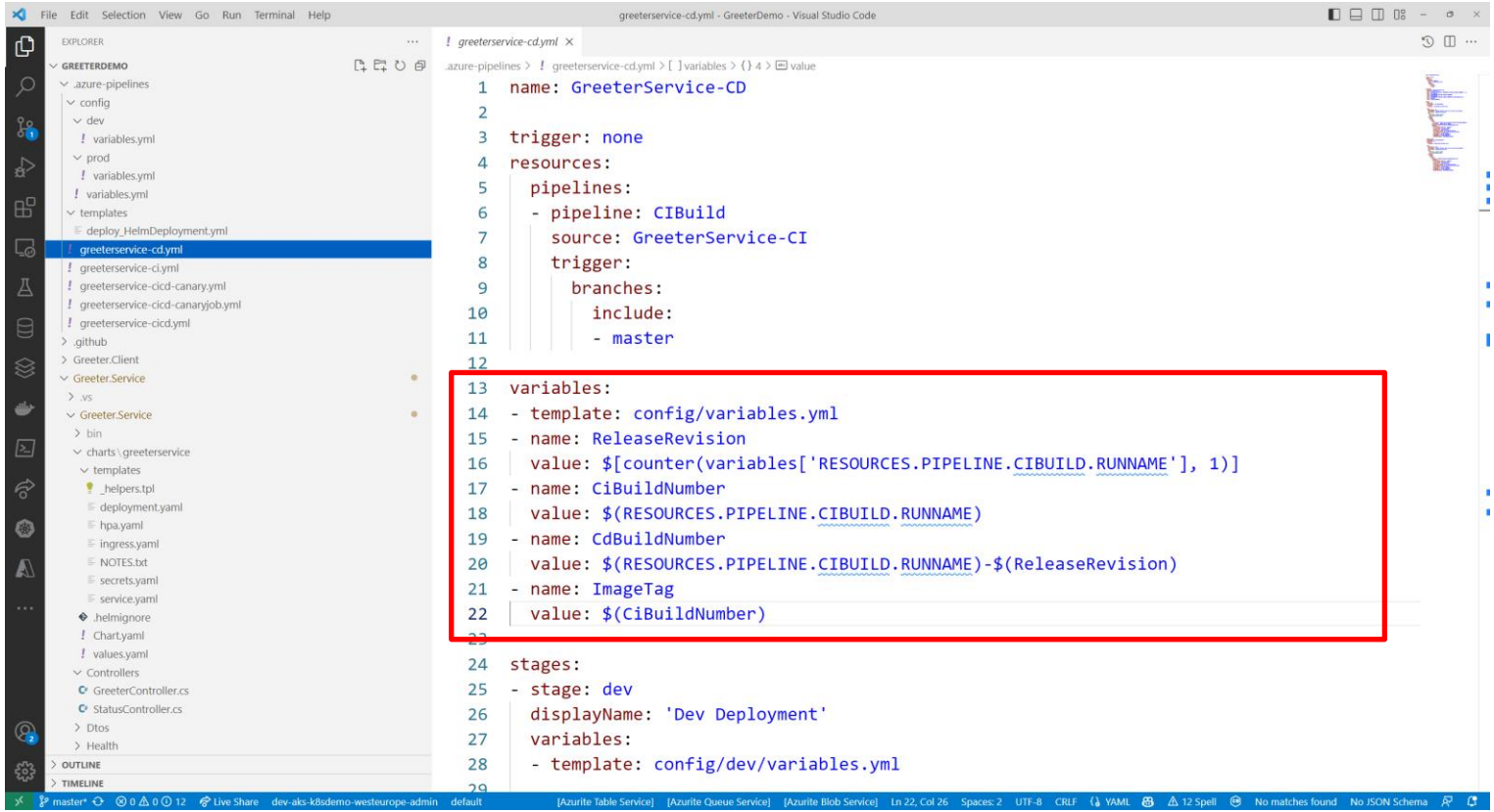
4tecture
empower your software solutions

Azure Pipelines

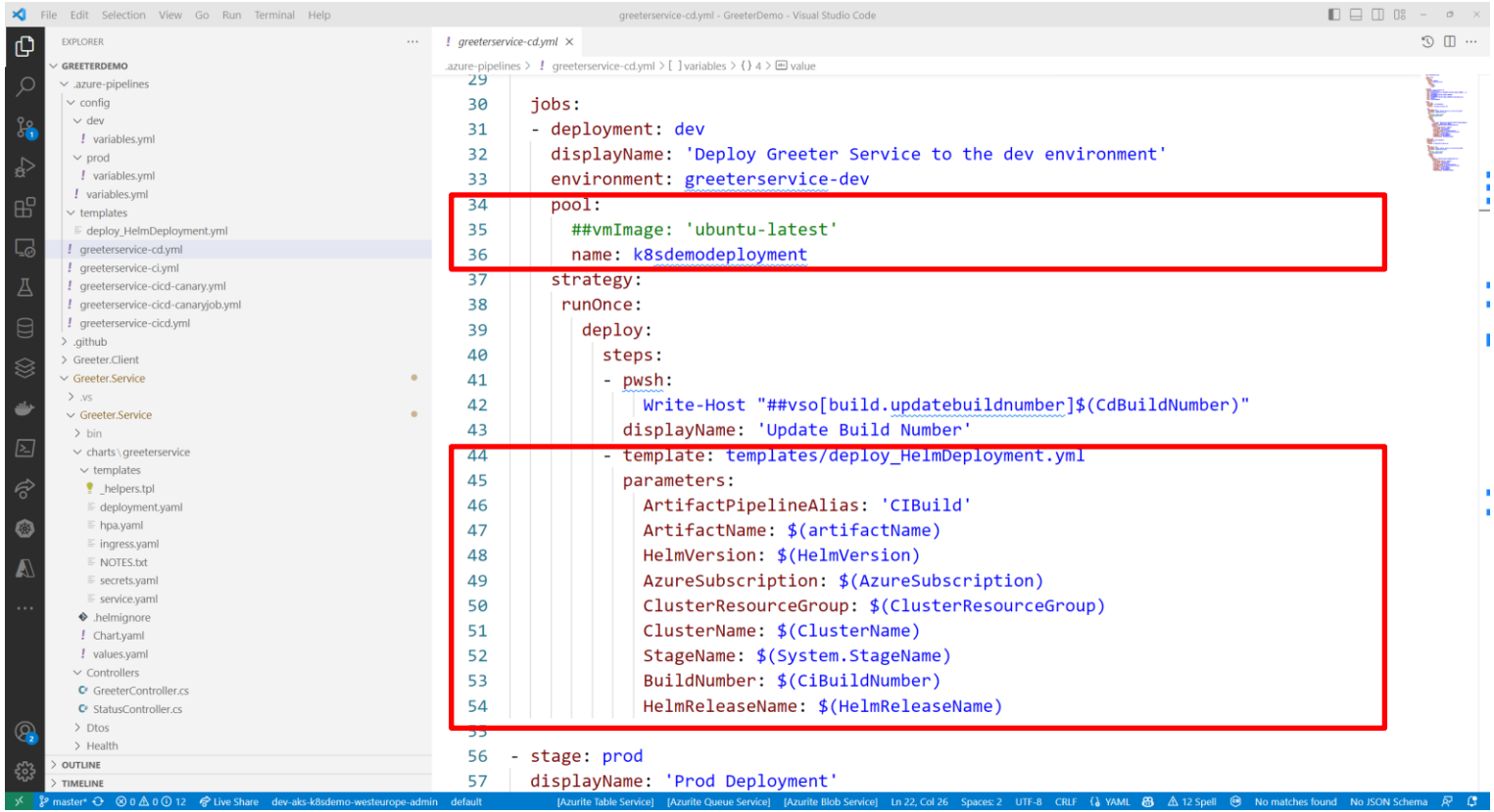
CD Pipeline with CI Resource



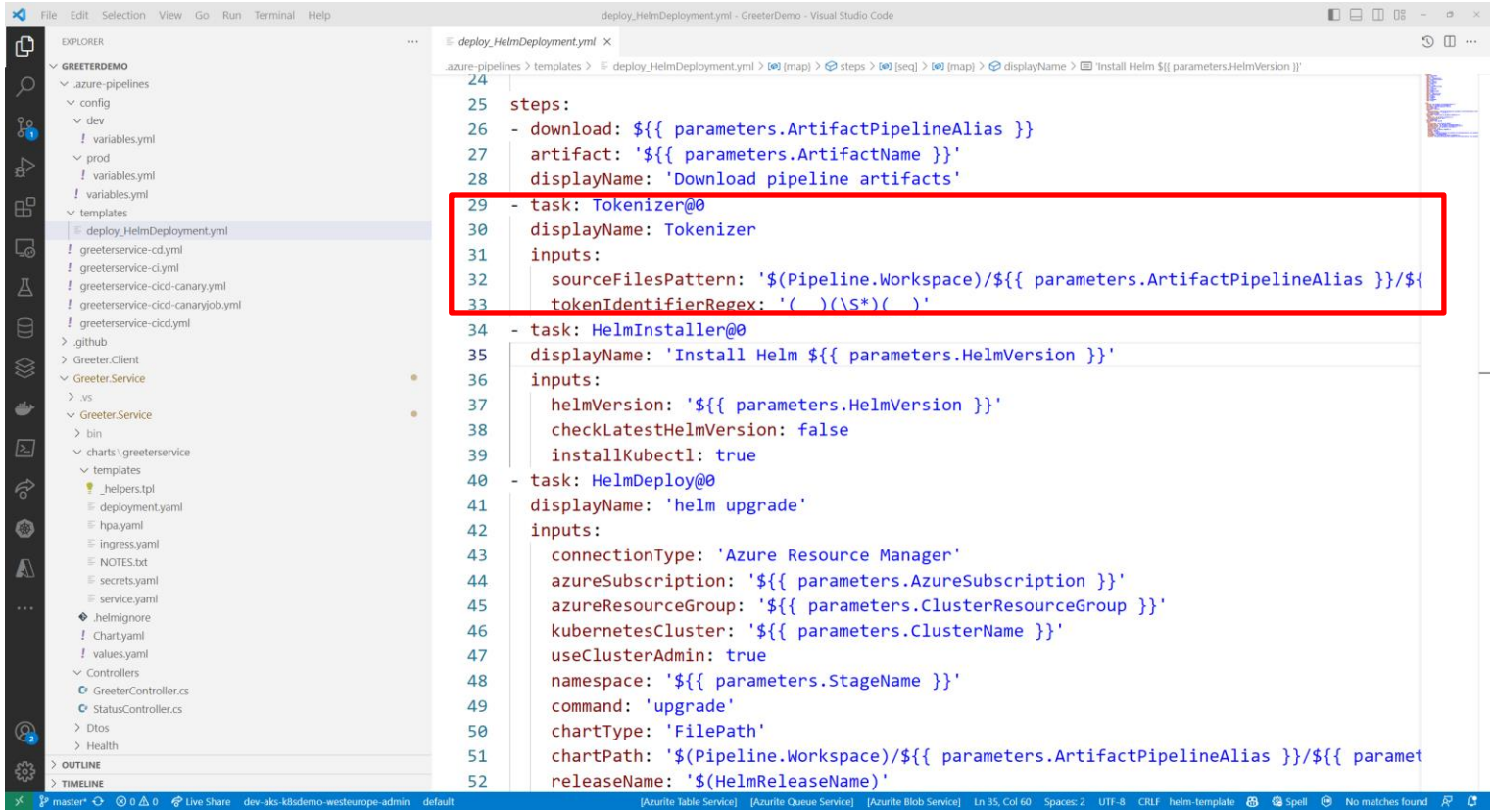
Define Versioning



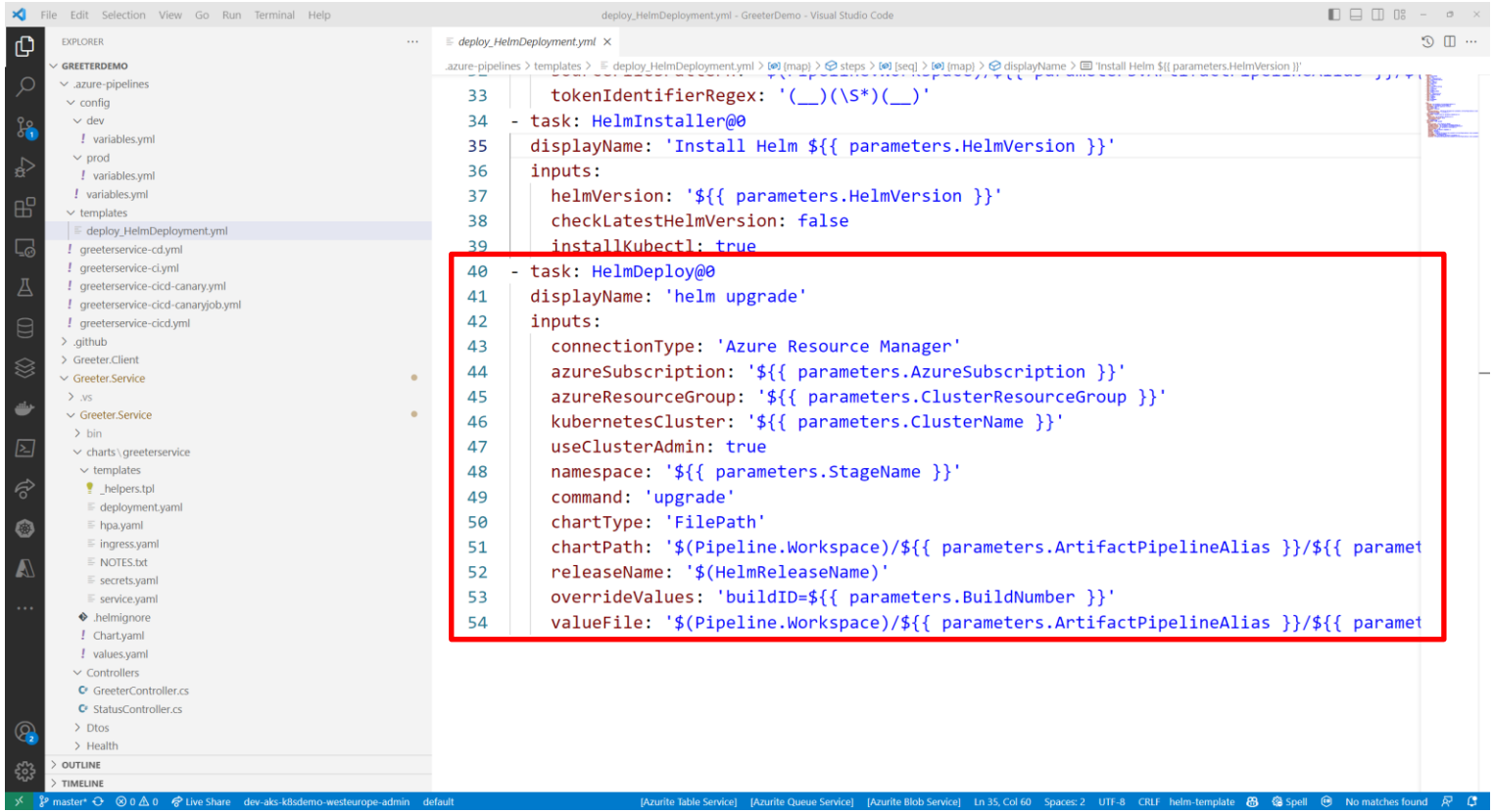
Deployment Job



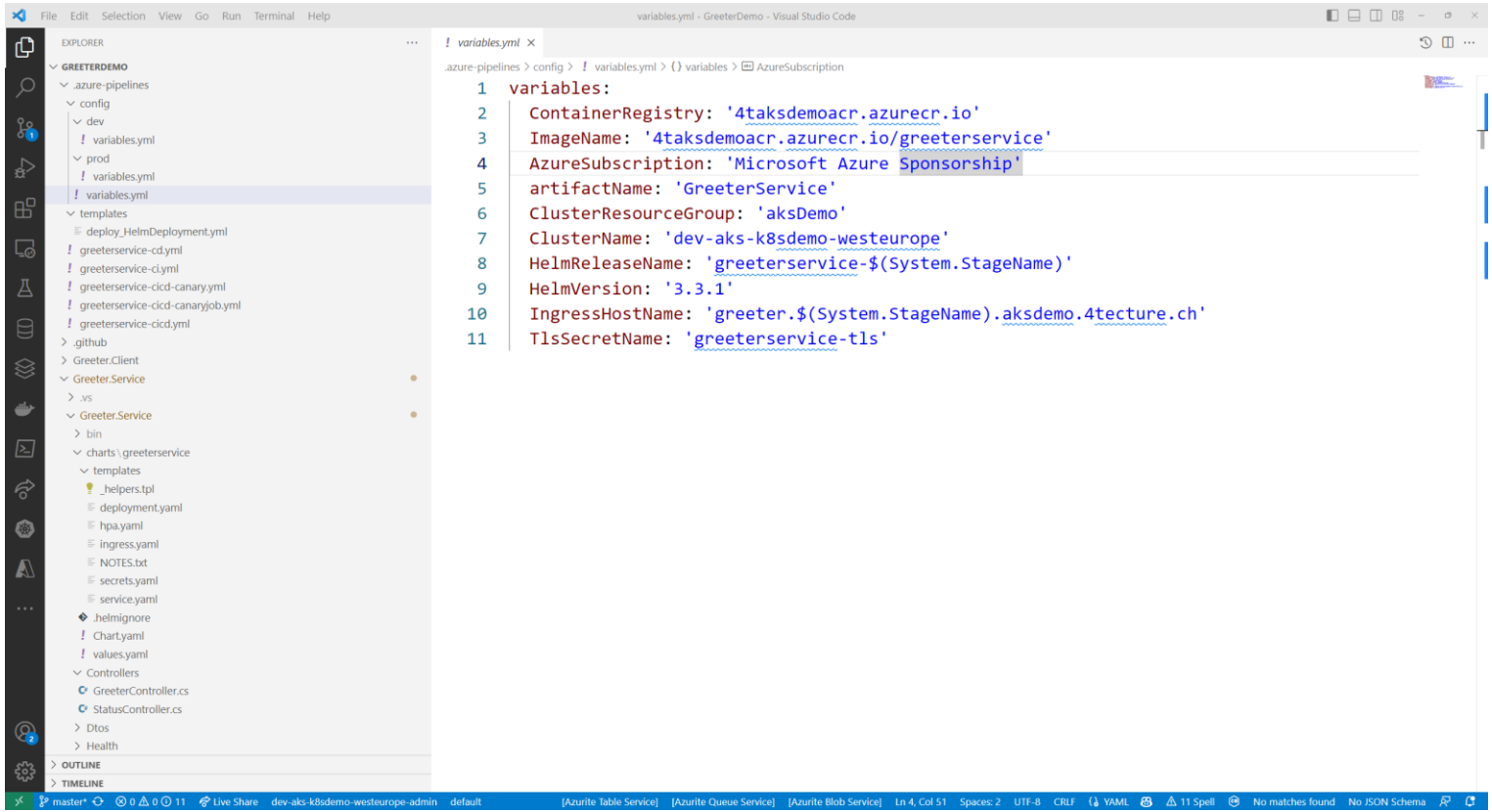
Deployment


deploy_HelmDeployment.yml - GreeterDemo - Visual Studio Code
EXPLORER
GREETERDEMO
.azure-pipelines
config
dev
variables.yml
prod
variables.yml
variables.yml
templates
deploy_HelmDeployment.yml
greeterservice-cd.yml
greeterservice-ci.yml
greeterservice-cicd-canary.yml
greeterservice-cicd-canaryjob.yml
greeterservice-cicd.yml
.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
OUTLINE
TIMELINE
deploy_HelmDeployment.yml
24
25 steps:
26 - download: ${{ parameters.ArtifactPipelineAlias }}
27 artifact: '${{ parameters.ArtifactName }}'
28 displayName: 'Download pipeline artifacts'
29 - task: Tokenizer@0
30 displayName: Tokenizer
31 inputs:
32 sourceFilesPattern: '$(Pipeline.Workspace)/${{ parameters.ArtifactPipelineAlias }}/${
33 tokenIdentifierRegex: '( )(\S*)( )'
34 - task: HelmInstaller@0
35 displayName: 'Install Helm ${{ parameters.HelmVersion }}'
36 inputs:
37 helmVersion: '${{ parameters.HelmVersion }}'
38 checkLatestHelmVersion: false
39 installKubectl: true
40 - task: HelmDeploy@0
41 displayName: 'helm upgrade'
42 inputs:
43 connectionType: 'Azure Resource Manager'
44 azureSubscription: '${{ parameters.AzureSubscription }}'
45 azureResourceGroup: '${{ parameters.ClusterResourceGroup }}'
46 kubernetesCluster: '${{ parameters.ClusterName }}'
47 useClusterAdmin: true
48 namespace: '${{ parameters.StageName }}'
49 command: 'upgrade'
50 chartType: 'FilePath'
51 chartPath: '$(Pipeline.Workspace)/${{ parameters.ArtifactPipelineAlias }}/${{ paramet
52 releaseName: '$(HelmReleaseName)'
master*
Live Share
dev-aks-k8sdemo-westeurope-admin
default
[Azurite Table Service]
[Azurite Queue Service]
[Azurite Blob Service]
Ln 35, Col 60
Spaces: 2
UTF-8
CRLF
helm-template
Spell
No matches found

Deployment

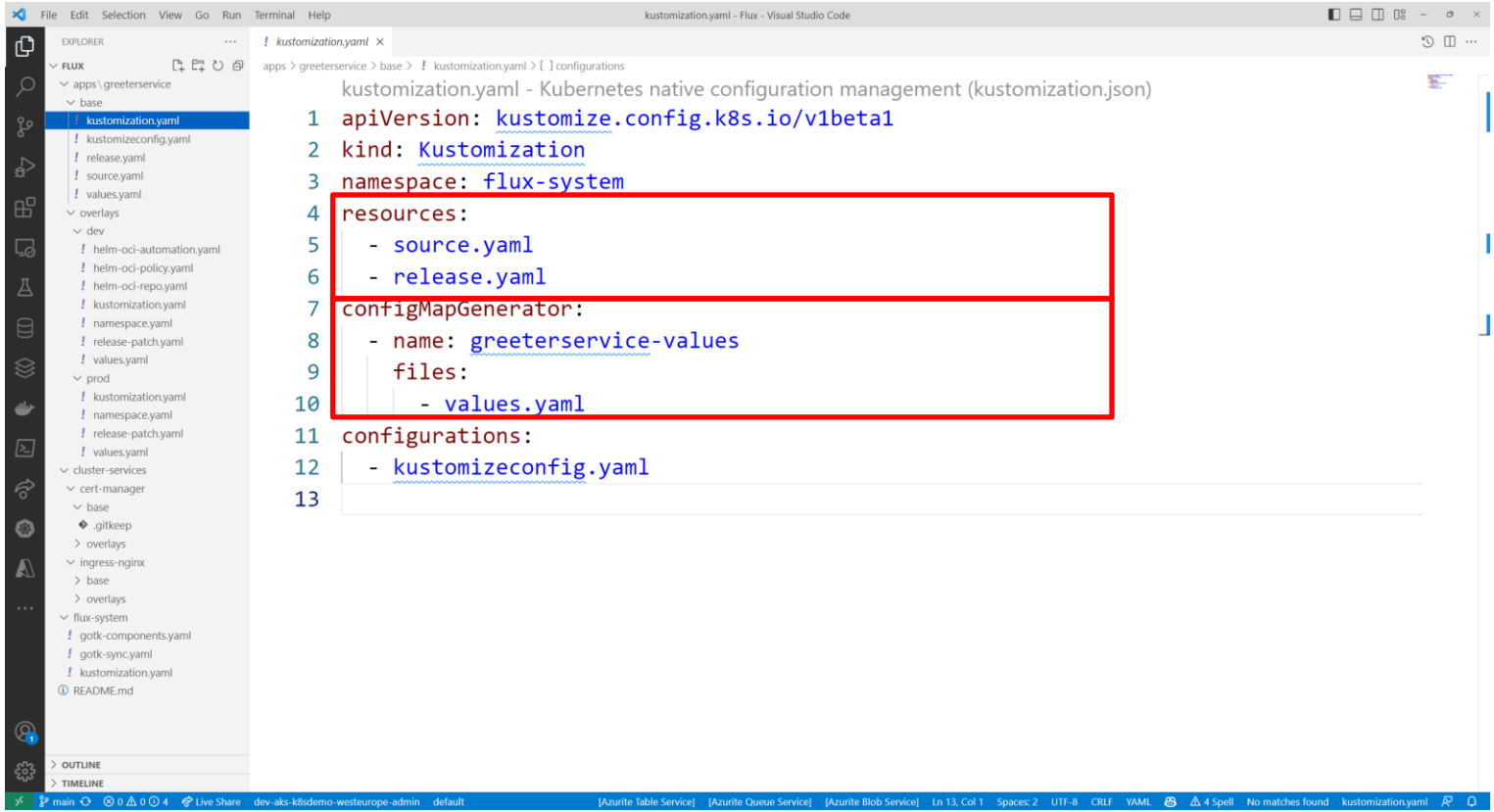

File Edit Selection View Go Run Terminal Help
deploy_HelmDeployment.yml - GreeterDemo - Visual Studio Code
EXPLORER
GREETERDEMO
.azure-pipelines
config
dev
variables.yml
prod
variables.yml
variables.yml
templates
deploy_HelmDeployment.yml
greeterservice-cd.yml
greeterservice-ci.yml
greeterservice-cicd-canary.yml
greeterservice-cicd-canaryjob.yml
greeterservice-cicd.yml
.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
OUTLINE
TIMELINE
deploy_HelmDeployment.yml
.azure-pipelines > templates > deploy_HelmDeployment.yml > {map} > steps > [seq] > {map} > displayName > 'Install Helm ${{ parameters.HelmVersion }}'
33 tokenIdentifierRegex: '(__)(\S*)(__)'
34 - task: HelmInstaller@0
35 displayName: 'Install Helm ${{ parameters.HelmVersion }}'
36 inputs:
37 helmVersion: '${{ parameters.HelmVersion }}'
38 checkLatestHelmVersion: false
39 installKubectl: true
40 - task: HelmDeploy@0
41 displayName: 'helm upgrade'
42 inputs:
43 connectionType: 'Azure Resource Manager'
44 azureSubscription: '${{ parameters.AzureSubscription }}'
45 azureResourceGroup: '${{ parameters.ClusterResourceGroup }}'
46 kubernetesCluster: '${{ parameters.ClusterName }}'
47 useClusterAdmin: true
48 namespace: '${{ parameters.StageName }}'
49 command: 'upgrade'
50 chartType: 'FilePath'
51 chartPath: '$(Pipeline.Workspace)/${{ parameters.ArtifactPipelineAlias }}/${{ paramet
52 releaseName: '$(HelmReleaseName)'
53 overrideValues: 'buildID=${{ parameters.BuildNumber }}'
54 valueFile: '$(Pipeline.Workspace)/${{ parameters.ArtifactPipelineAlias }}/${{ paramet
master* Live Share dev-aks-k8sdemo-westeurope-admin default
[Azurite Table Service] [Azurite Queue Service] [Azurite Blob Service] Ln 35, Col 60 Spaces: 2 UTF-8 CRLF helm-template Spell No matches found

Configuration

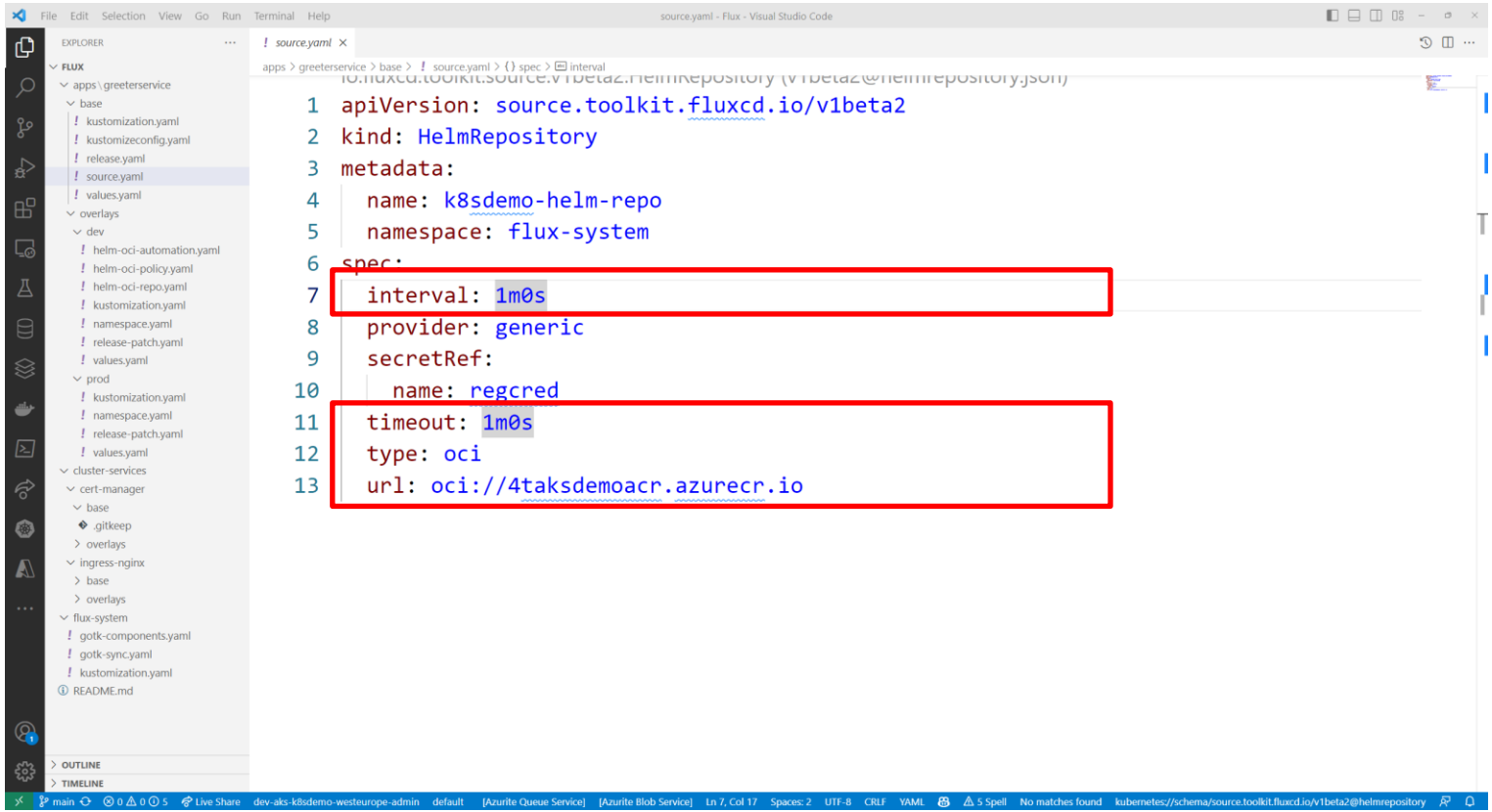


Flux

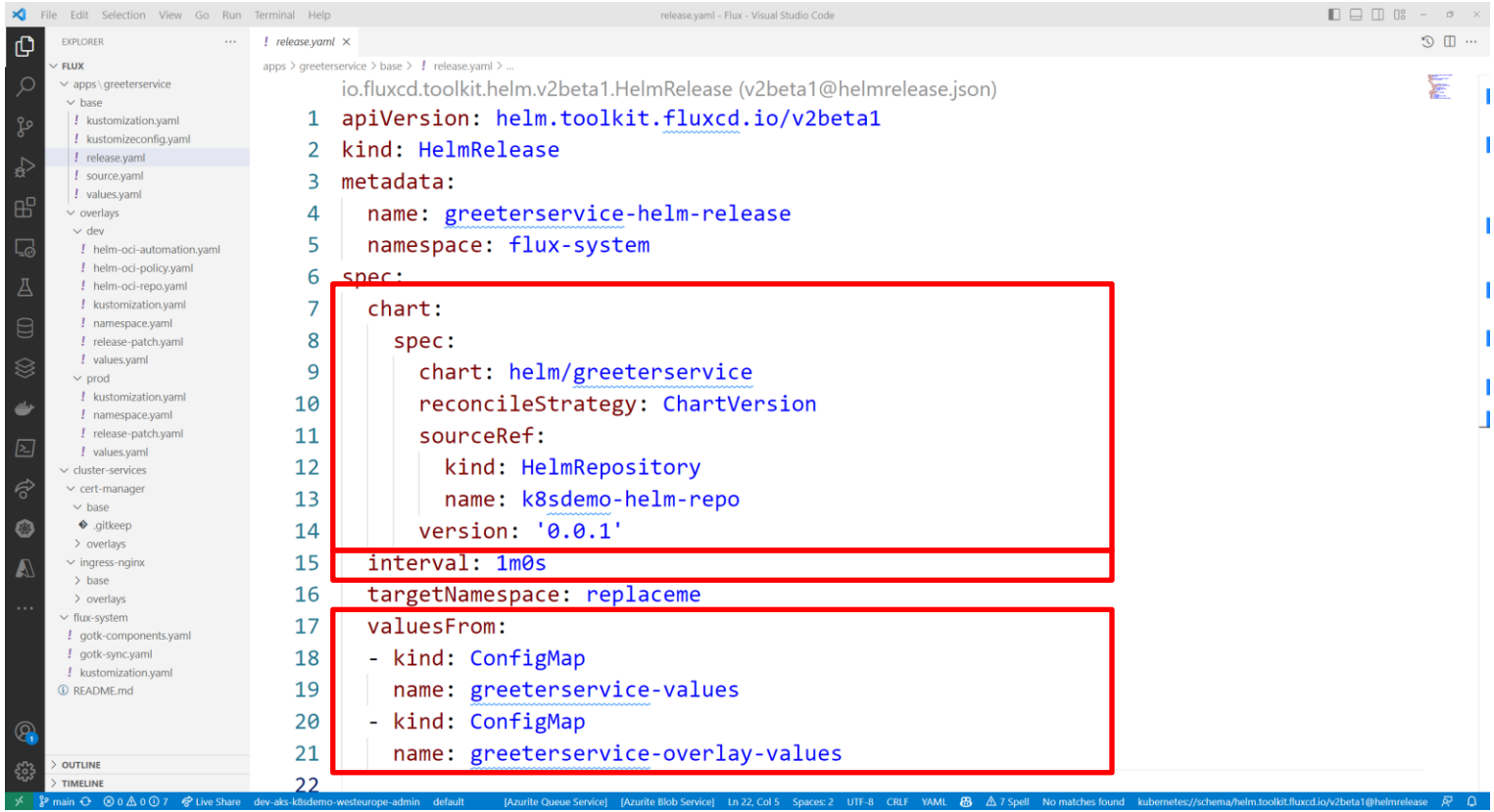
Kustomization



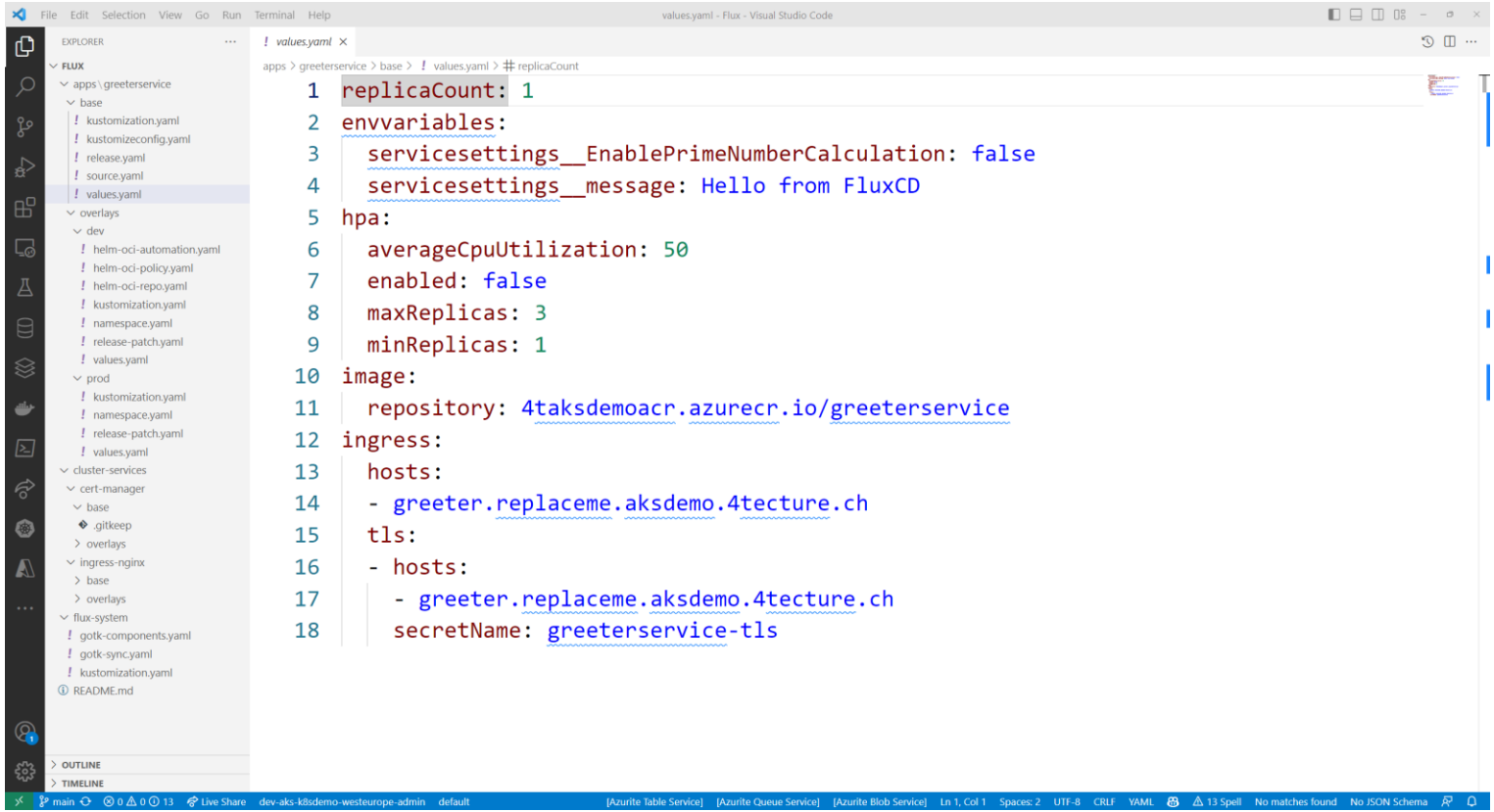
Helm Repository



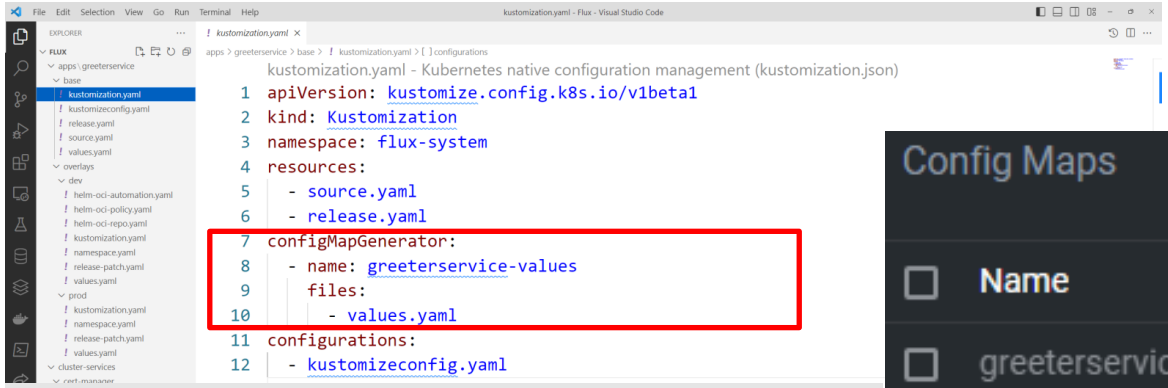
Helm Release

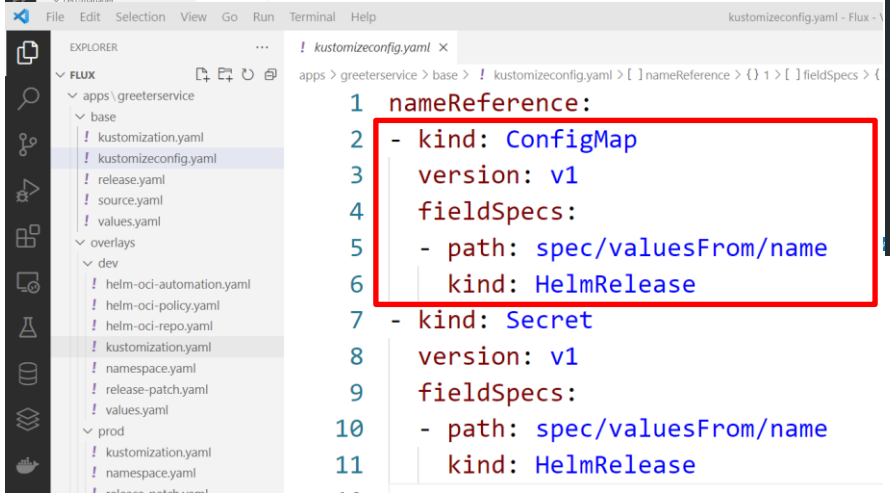


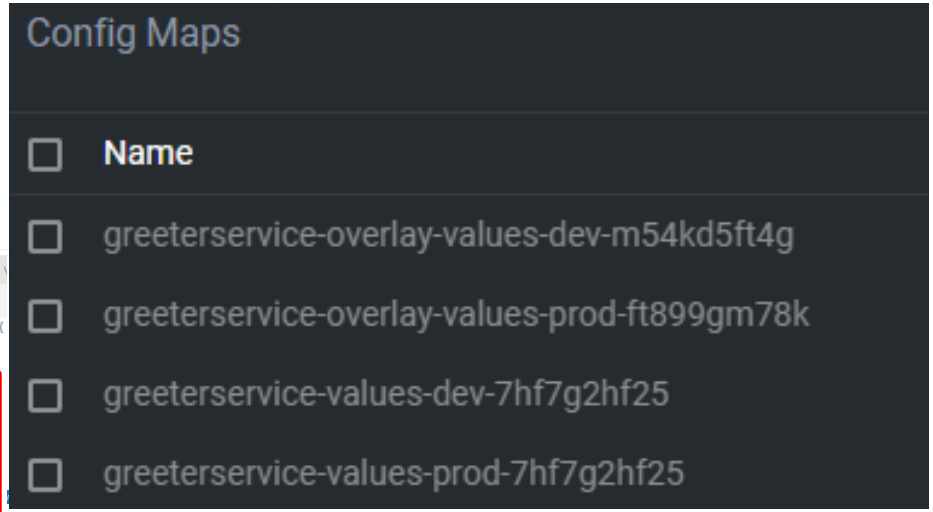
Configuration



Unique Resource Names





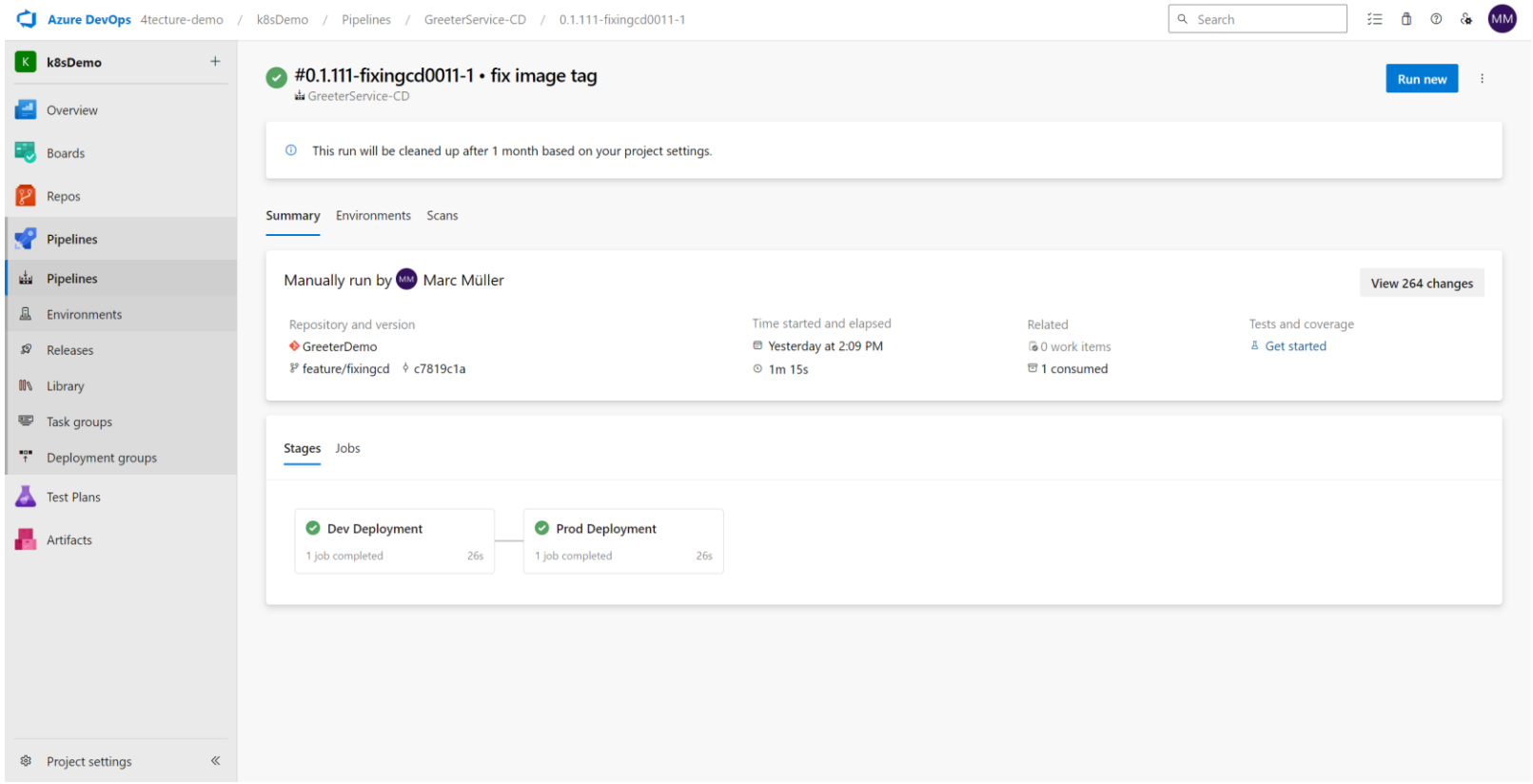


Staging

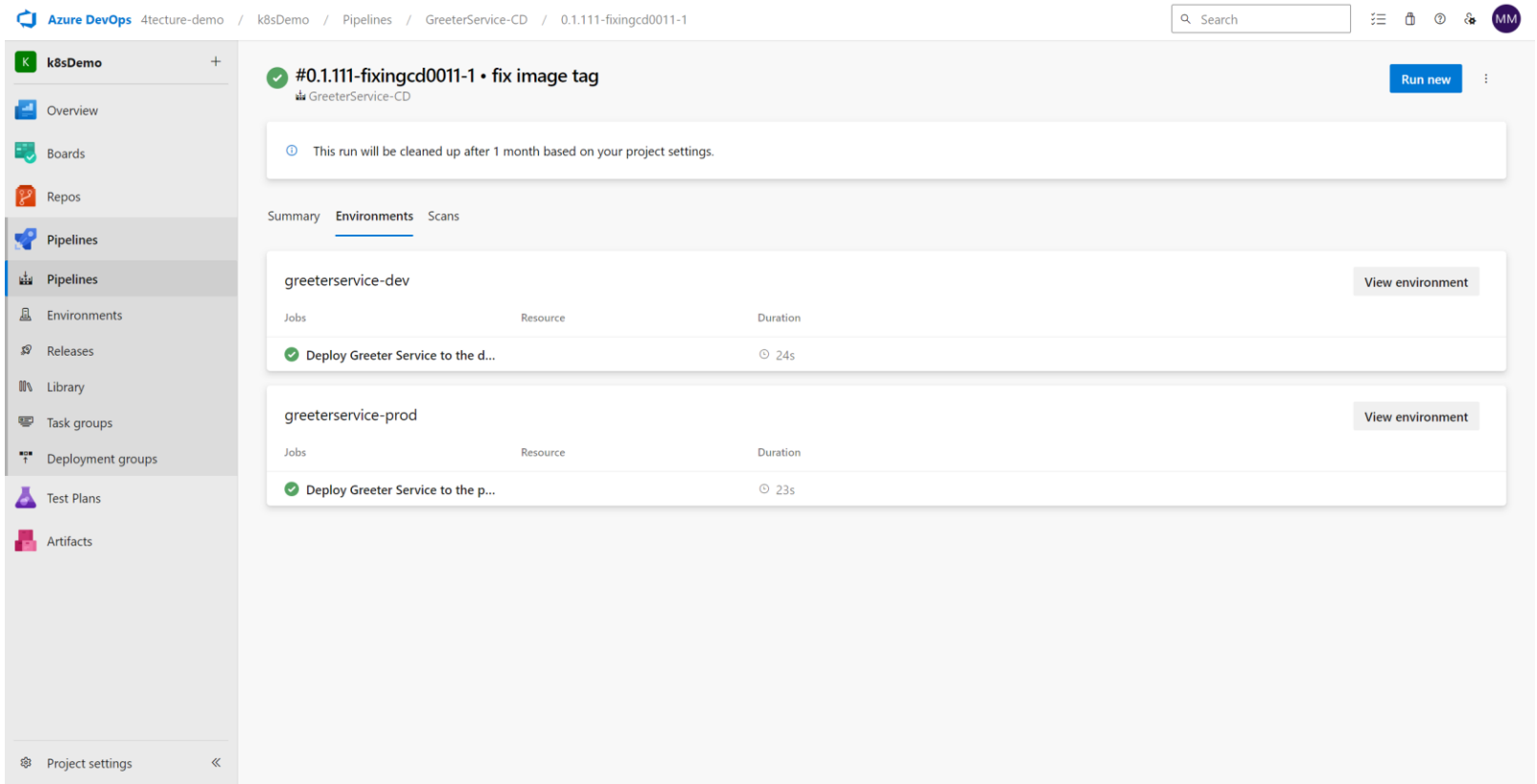
4tecture®
empower your software solutions

Azure Pipelines

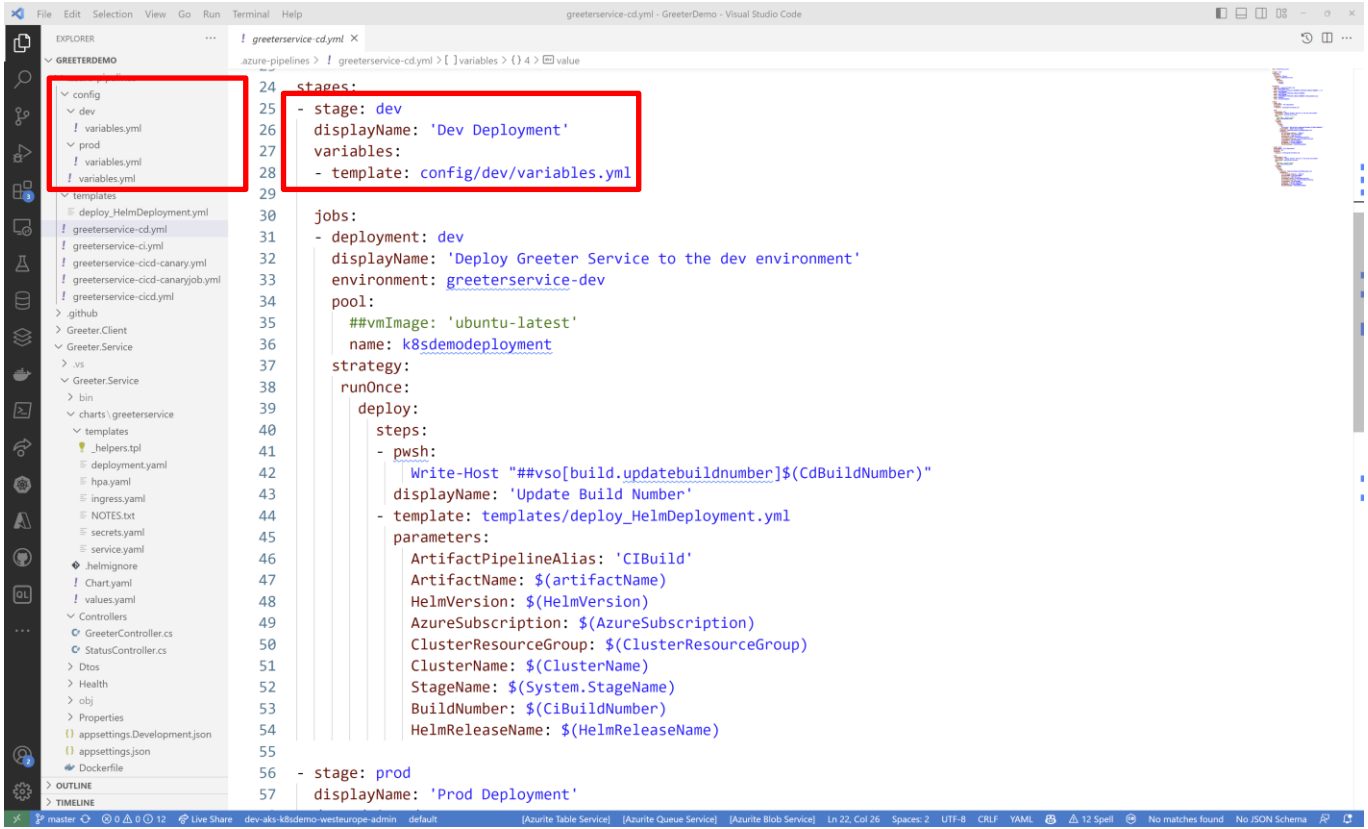
Pipeline Stages



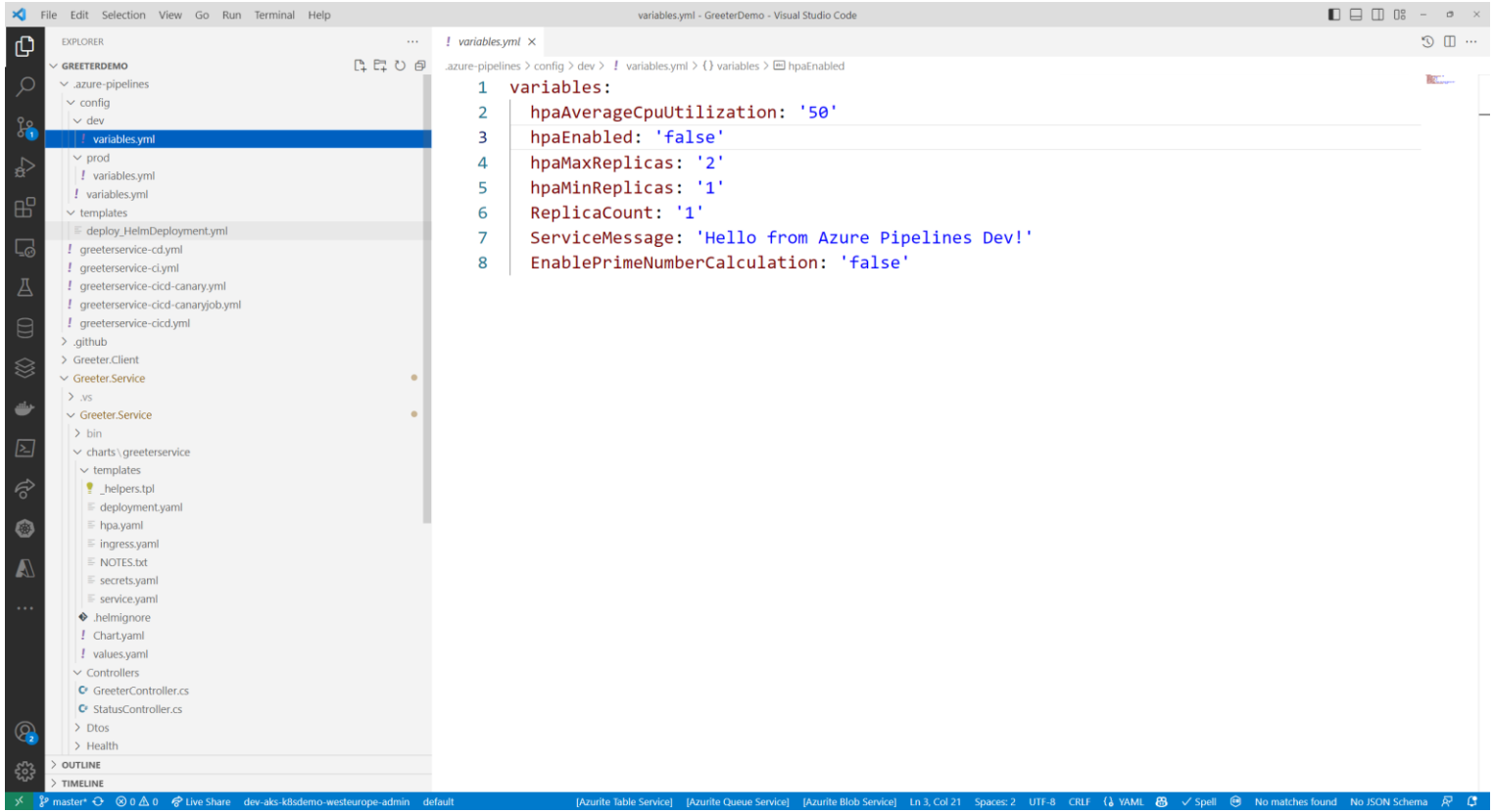
Environments



Dedicated Settings per Stage



Dedicated Settings per Stage



Flux

Structuring Best Practices

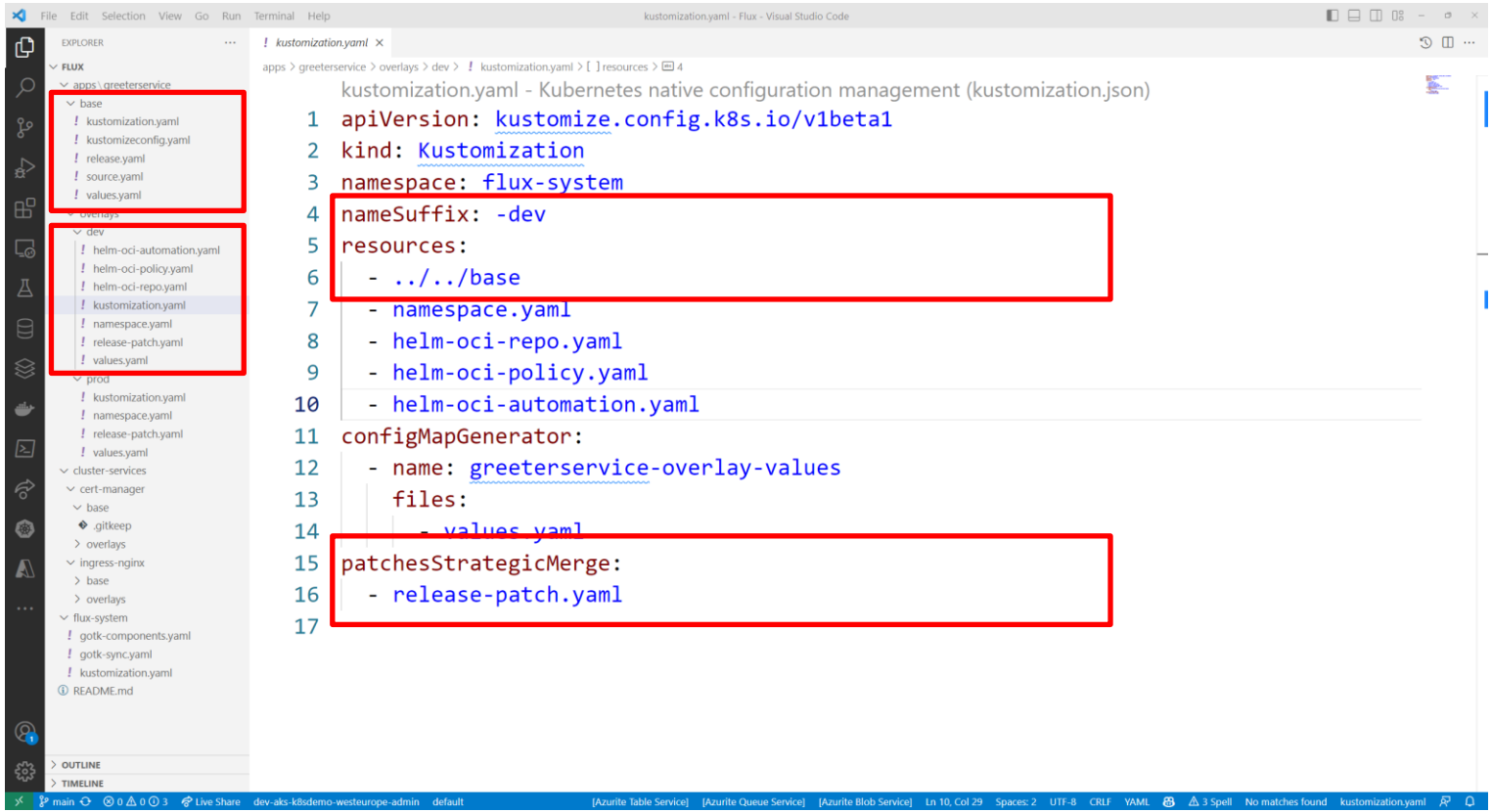
- Use base configuration with overlays
- Overlay config maps / secrets
- Overlay objects with patches

- apps
  - application1
    - base
    - overlays
      - dev
      - prod
  - application2
    - base
    - overlays
      - dev
      - prod

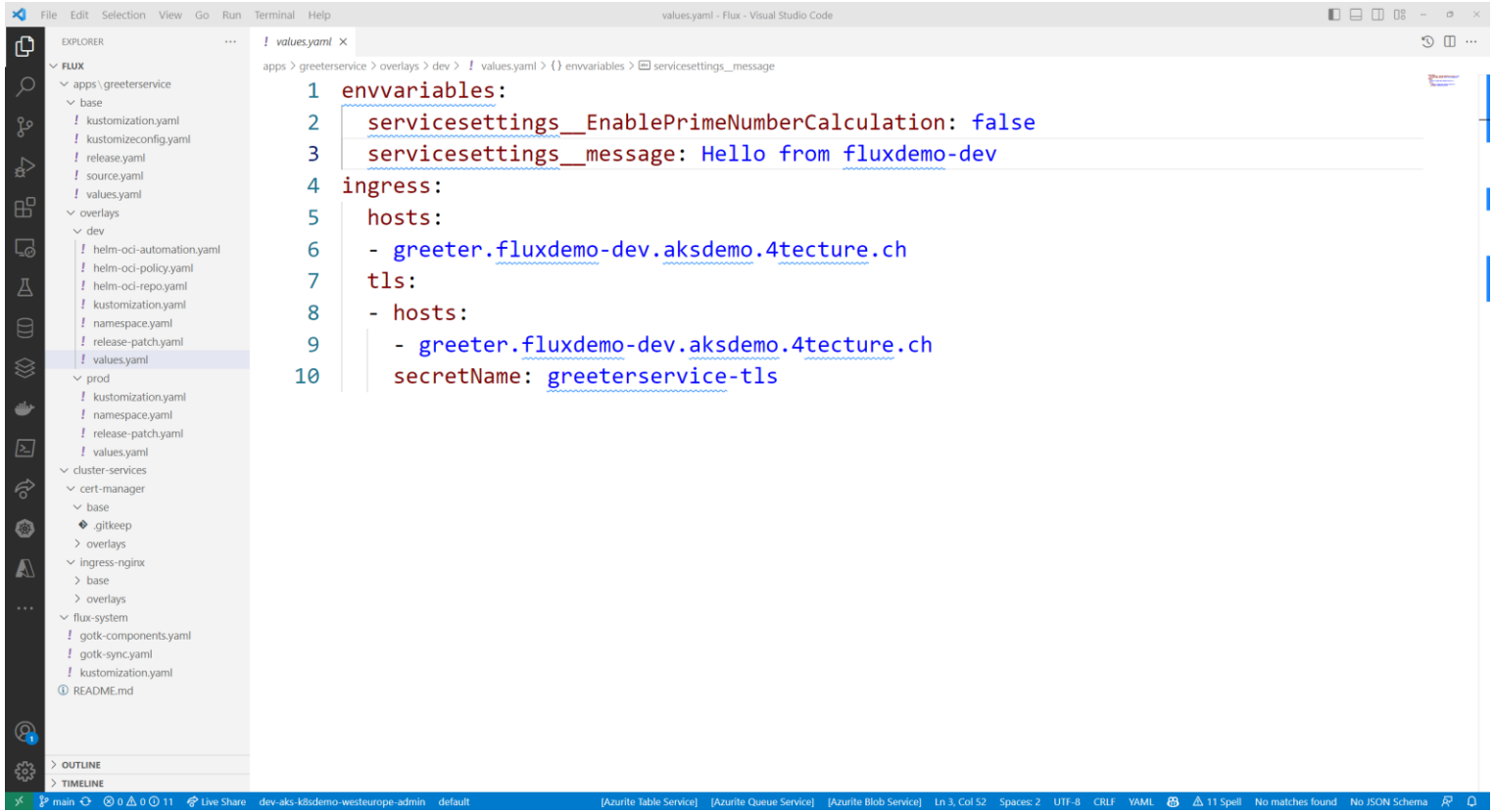
Structure


FLUX
apps \ greeterservice
base
kustomization.yaml
kustomizeconfig.yaml
release.yaml
source.yaml
values.yaml
overlays
dev
helm-oci-automation.yaml
helm-oci-policy.yaml
helm-oci-repo.yaml
kustomization.yaml
namespace.yaml
release-patch.yaml
values.yaml
prod
kustomization.yaml
namespace.yaml
release-patch.yaml
values.yaml

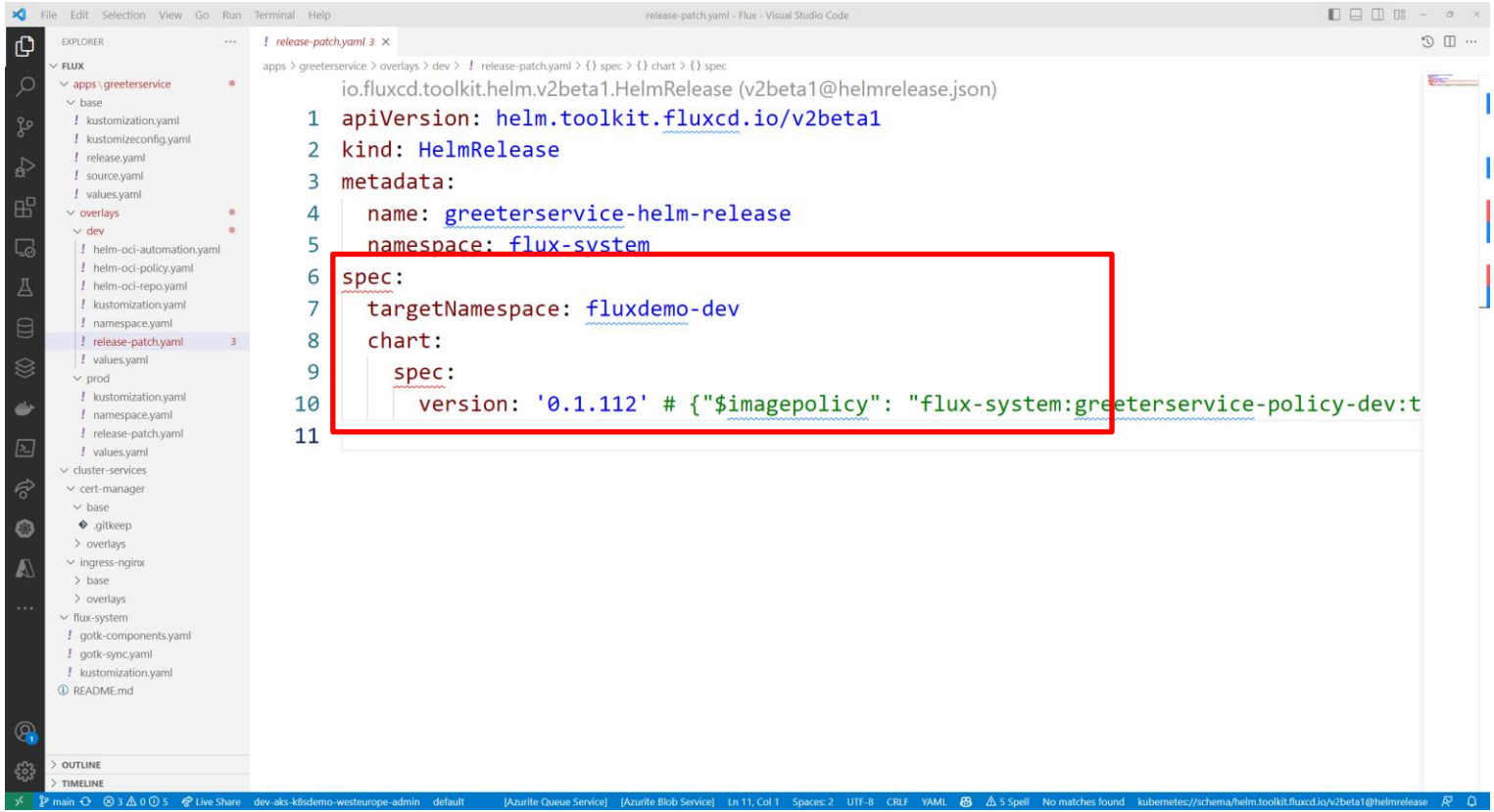
Overlay Kustomization



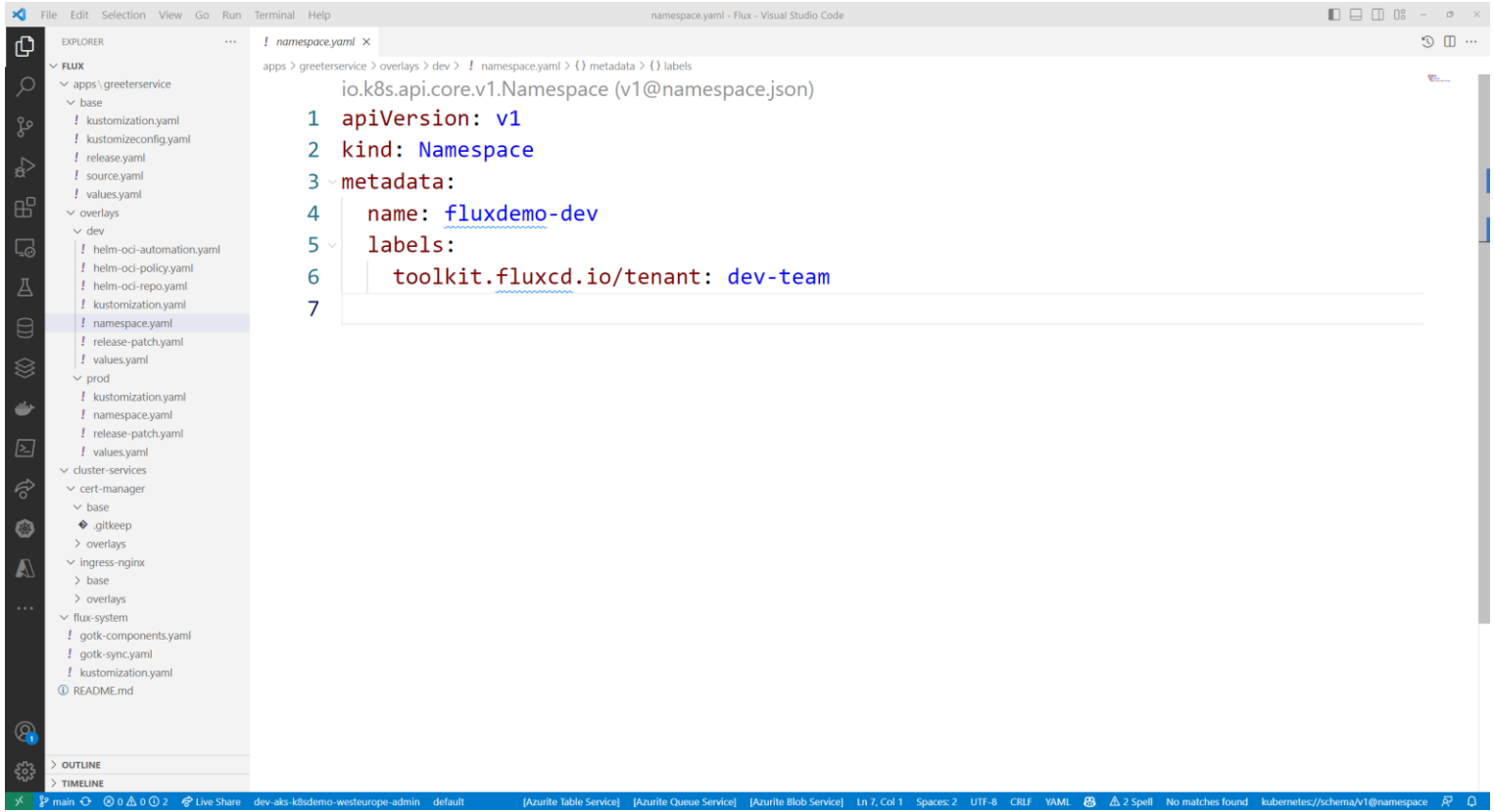
Overlay Values



Patch Release



Add dedicated objects



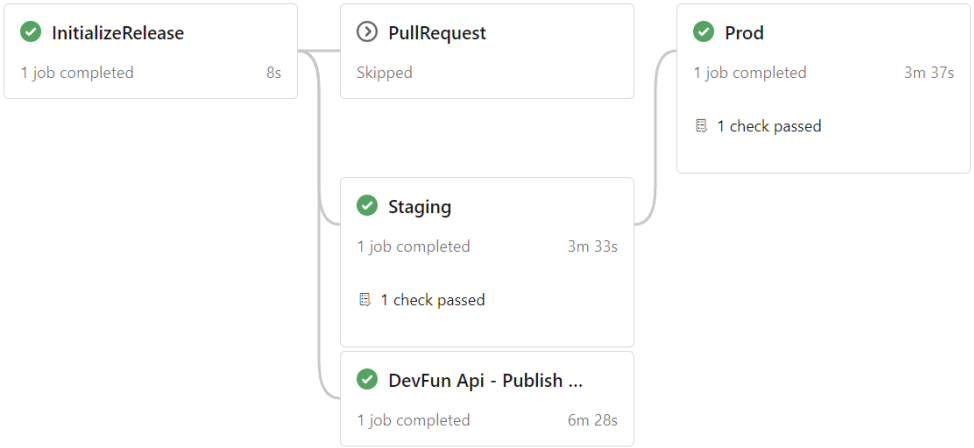
Automatic Updates

4tecture®
empower your software solutions

Azure Pipelines

Out-of-the box: triggers and branches

- Pipeline Triggers on Source Update
- Use a branching concept and PR workflow
- PR deployments
- Checks and Approvals



Flux

GitOps

- Git Repo defines what is deployed
- No update without new commit

Repo Updates

- Task after successful CI
- Listen for new images / Helm charts

Flux Image Automation

- Monitor Container Registry
- Analyze updates based on policy
- Change configuration – commit change in GIT repo
- Standard GitOps flow is triggered

Image Repository

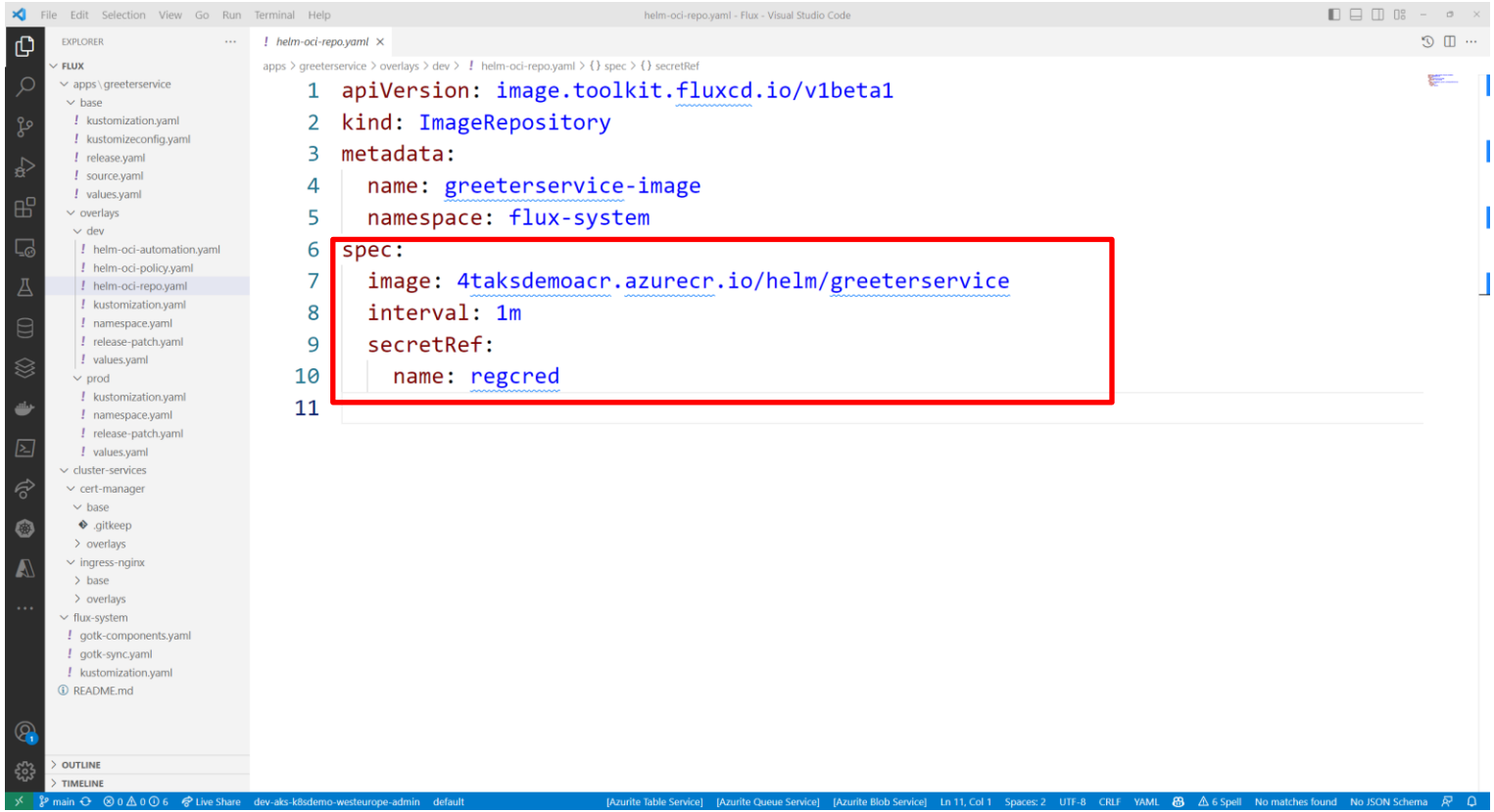


Image Policy

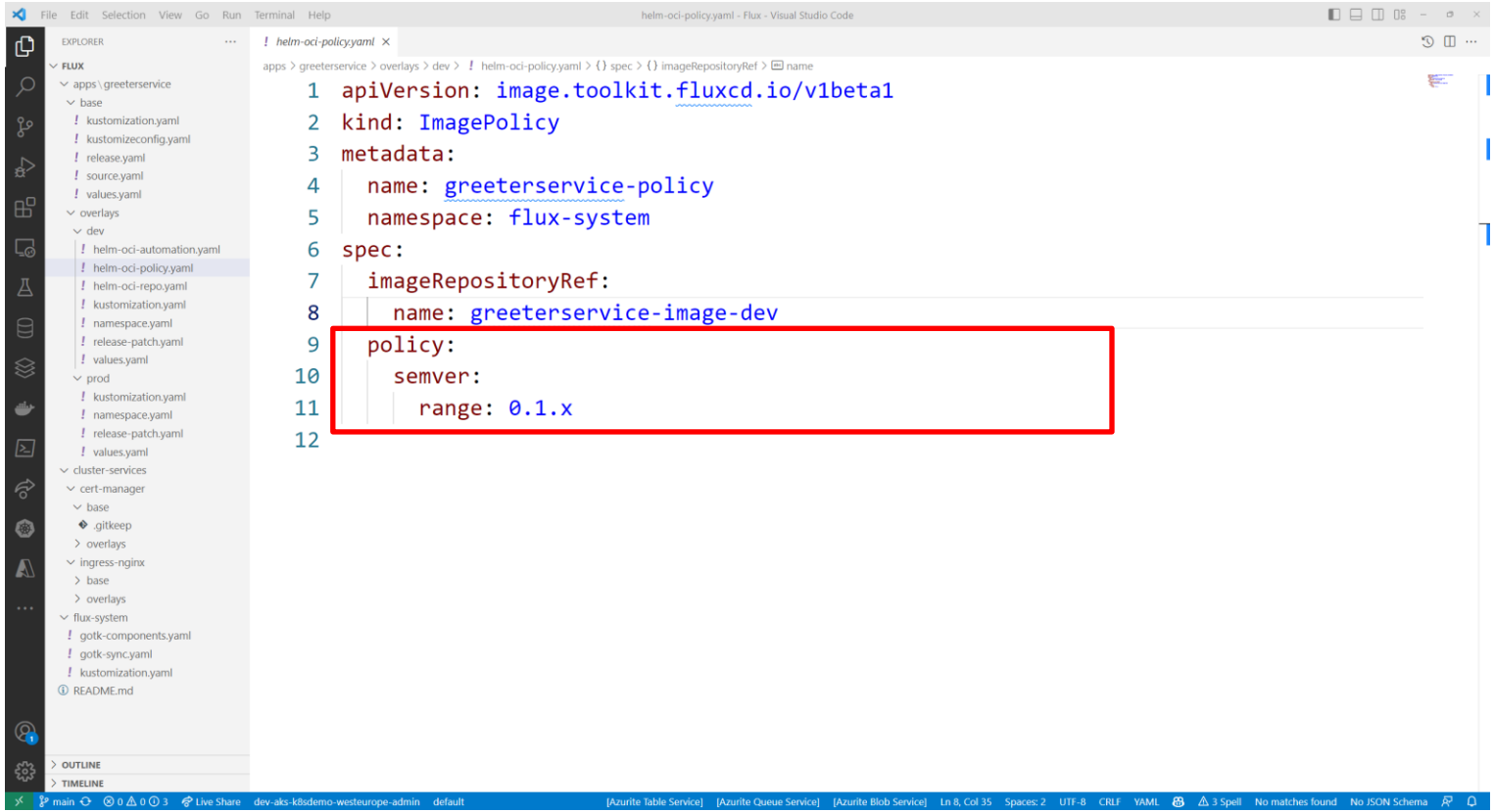
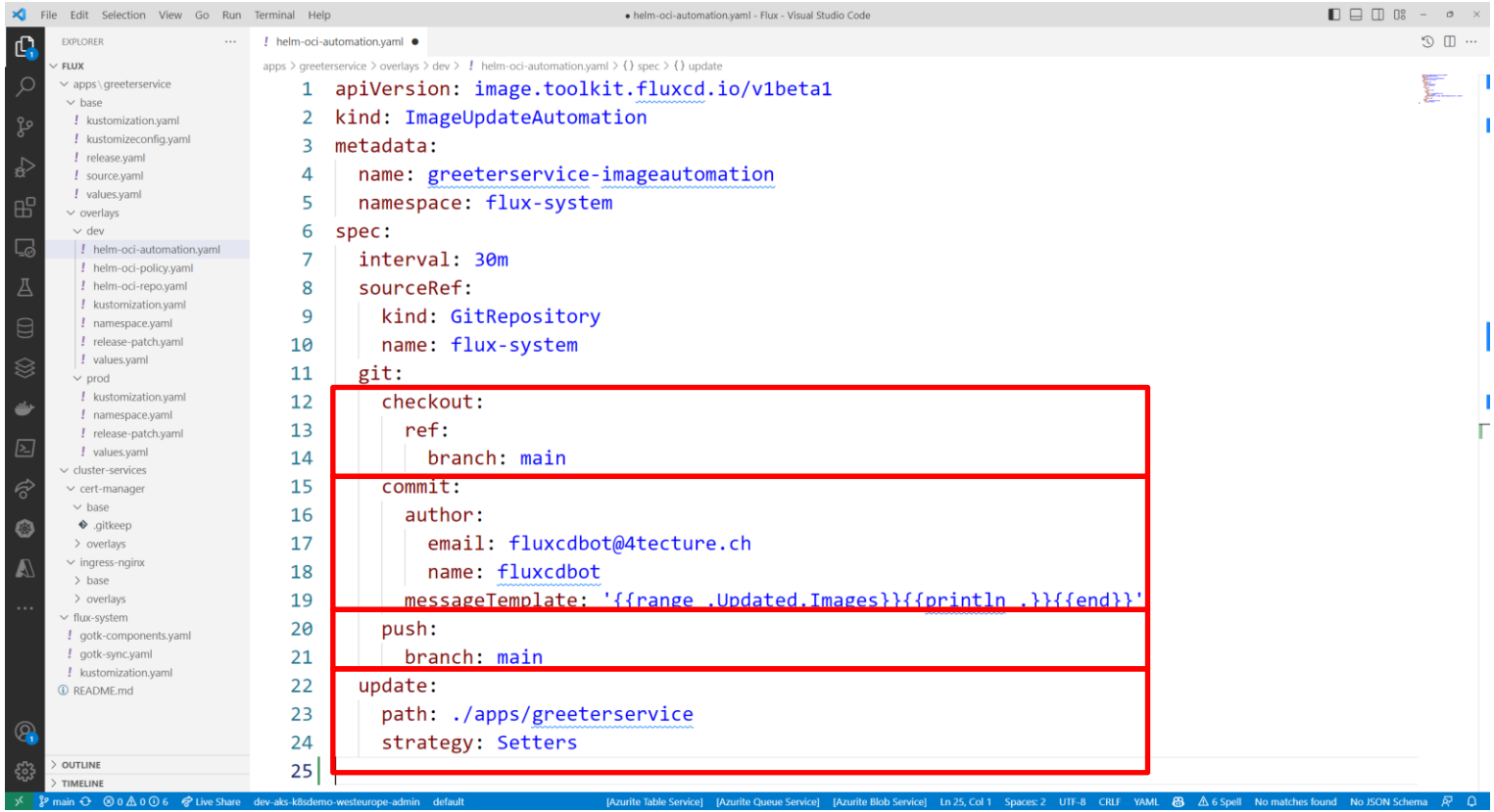
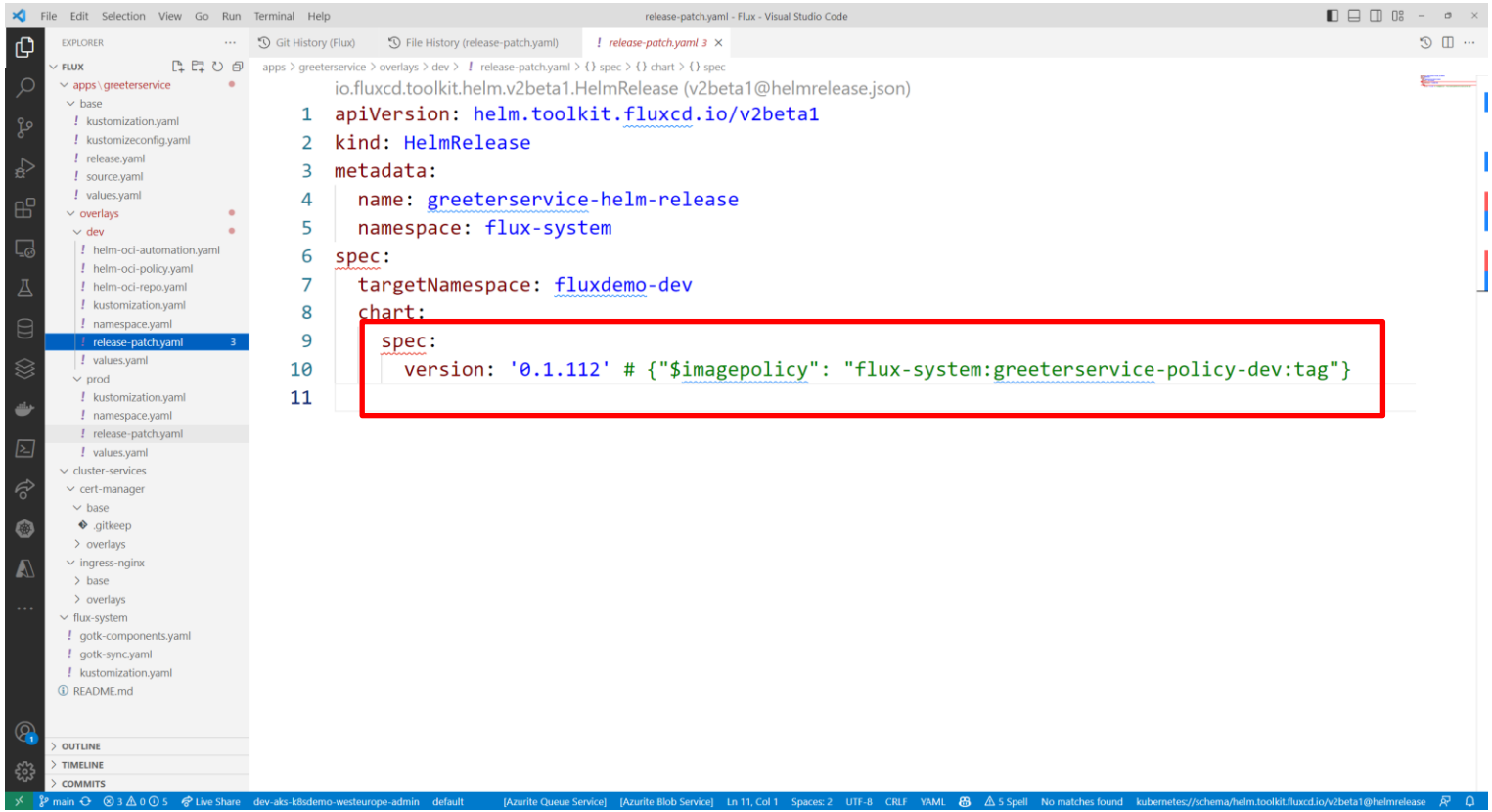


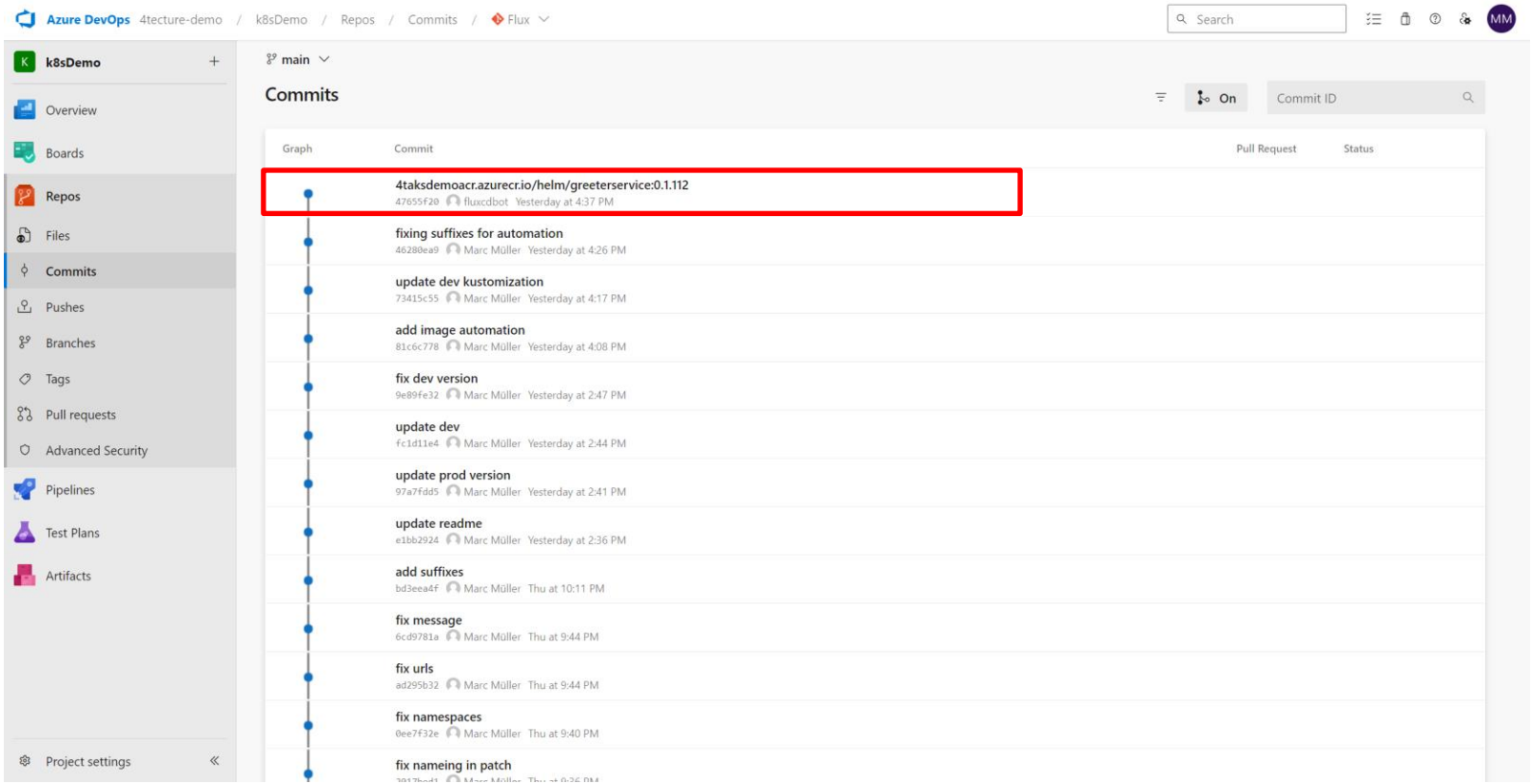
Image Update Automation



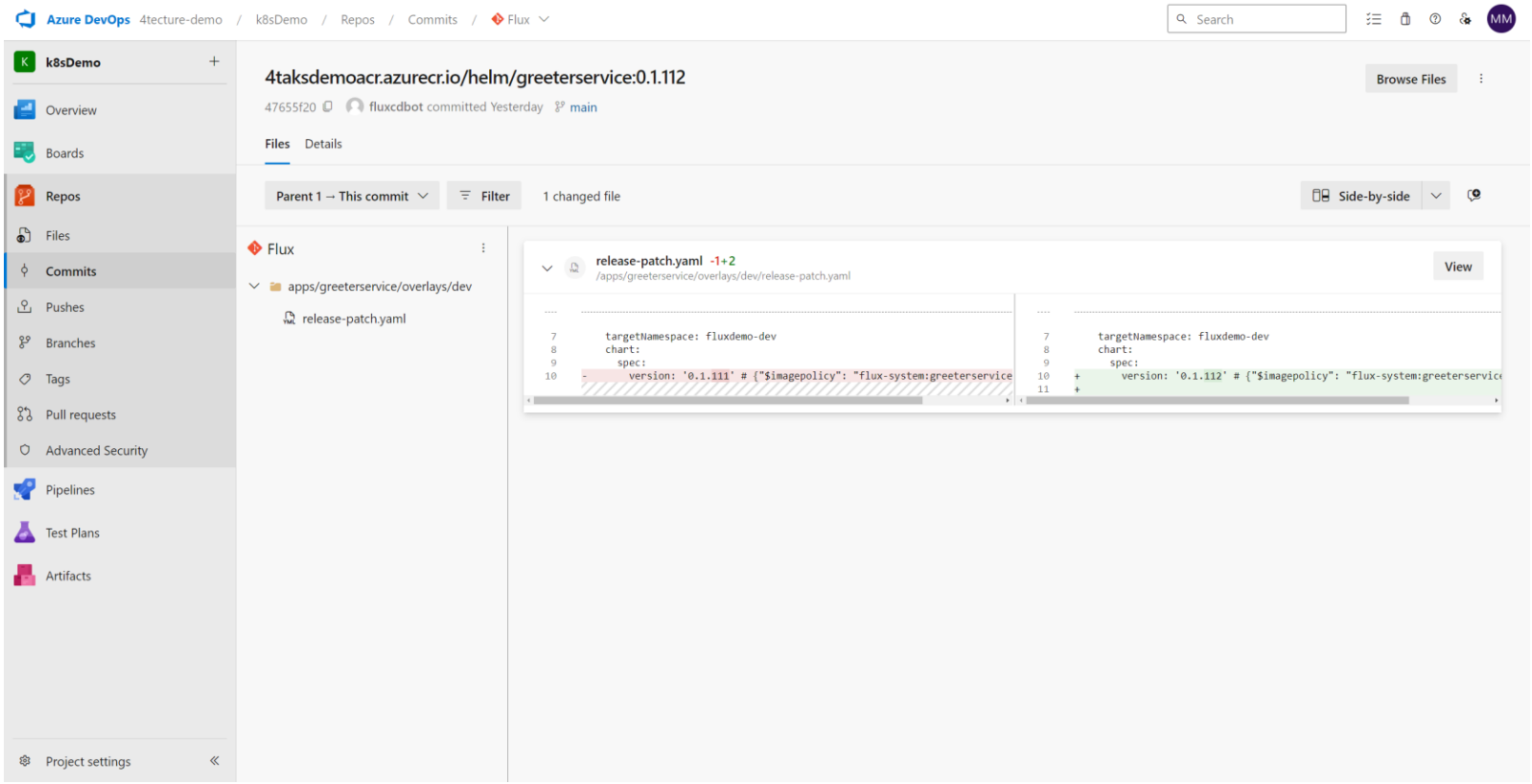
Update markers



Updates


Azure DevOps 4tecture-demo / k8sDemo / Repos / Commits / Flux
Search
MM
k8sDemo
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Test Plans
Artifacts
Project settings
main
Commits
On
Commit ID
Graph
Commit
Pull Request
Status
4taksdemoacr.azurecr.io/helm/greeterservice:0.1.112
47655f20 fluxcdbot Yesterday at 4:37 PM
fixing suffixes for automation
46280ea9 Marc Müller Yesterday at 4:26 PM
update dev kustomization
73415c55 Marc Müller Yesterday at 4:17 PM
add image automation
81c6c778 Marc Müller Yesterday at 4:08 PM
fix dev version
9e89fe32 Marc Müller Yesterday at 2:47 PM
update dev
fc1d11e4 Marc Müller Yesterday at 2:44 PM
update prod version
97a7fdd5 Marc Müller Yesterday at 2:41 PM
update readme
e1bb2924 Marc Müller Yesterday at 2:36 PM
add suffixes
bd3eea4f Marc Müller Thu at 10:11 PM
fix message
6cd9781a Marc Müller Thu at 9:44 PM
fix urls
ad295b32 Marc Müller Thu at 9:44 PM
fix namespaces
0ee7f32e Marc Müller Thu at 9:40 PM
fix nameing in patch

Updates


Azure DevOps 4tecture-demo / k8sDemo / Repos / Commits / Flux
Search
MM
k8sDemo
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Test Plans
Artifacts
Project settings
4taksdemoacr.azurecr.io/helm/greeterservice:0.1.112
47655f20 fluxcdbot committed Yesterday main
Browse Files
Files Details
Parent 1 → This commit
Filter
1 changed file
Side-by-side
Flux
apps/greeterservice/overlays/dev
release-patch.yaml
release-patch.yaml -1+2
/apps/greeterservice/overlays/dev/release-patch.yaml
View
targetNamespace: fluxdemo-dev
chart:
spec:
- version: '0.1.111' # {"$imagepolicy": "flux-system:greeterservice
targetNamespace: fluxdemo-dev
chart:
spec:
+ version: '0.1.112' # {"$imagepolicy": "flux-system:greeterservic

Cluster Setup with Azure Pipelines

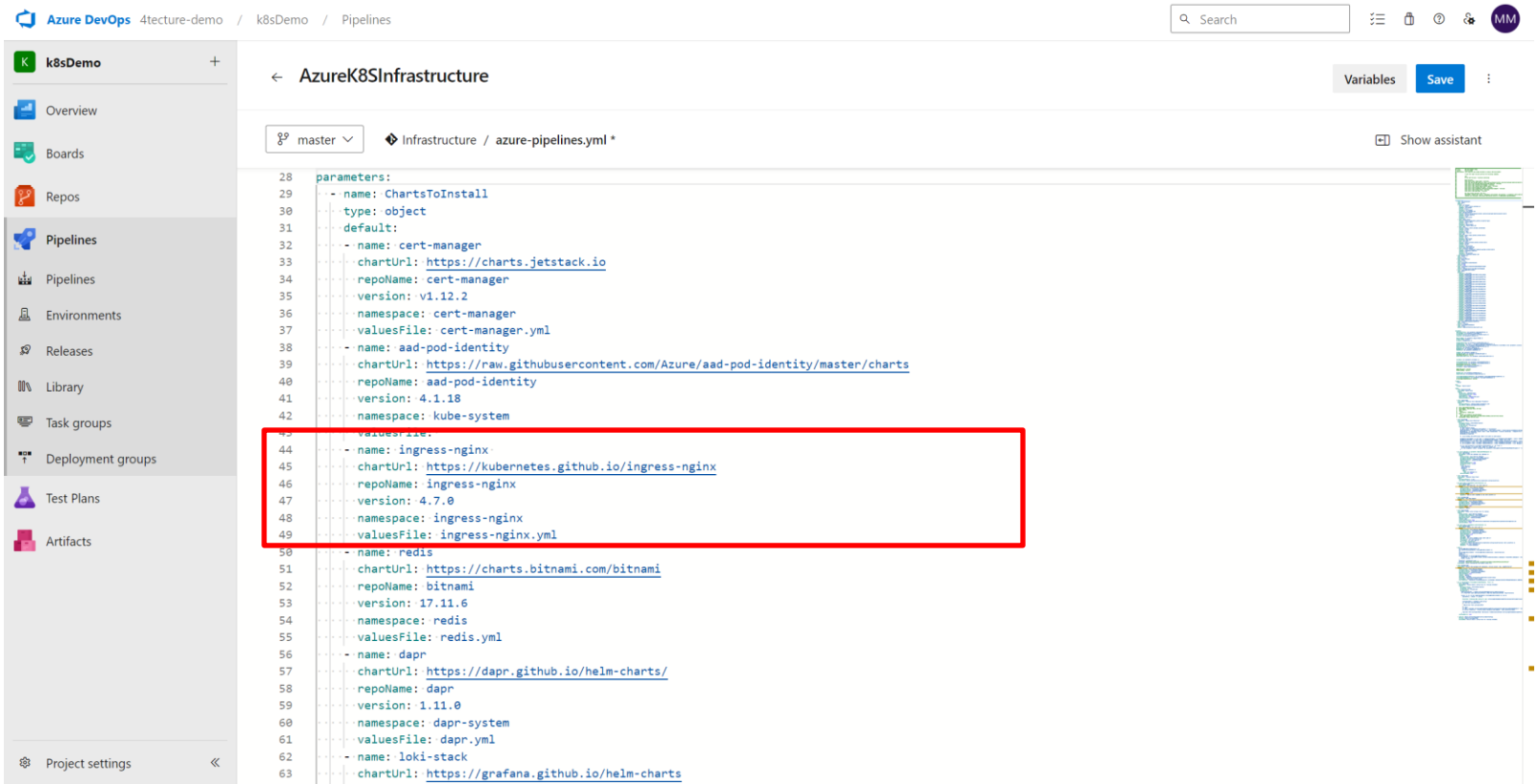
4tecture
empower your software solutions

Deploy Cluster Services

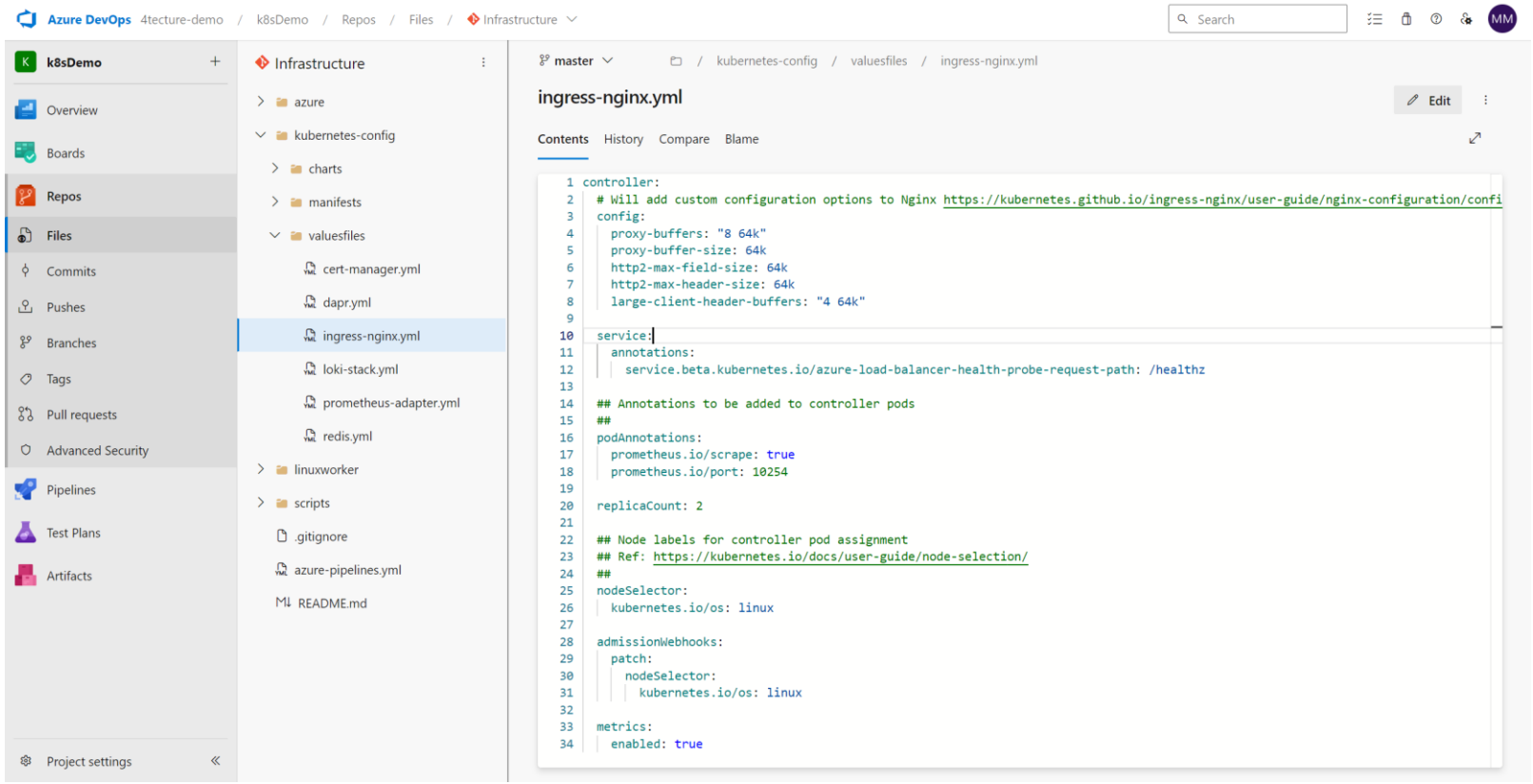
- Cluster setup has a lot of Helm chart deployments
- Configuration as Code
- Pipeline automation on configuration object

- Runtime Parameters / Parameters support objects
- Define your cluster configuration as object

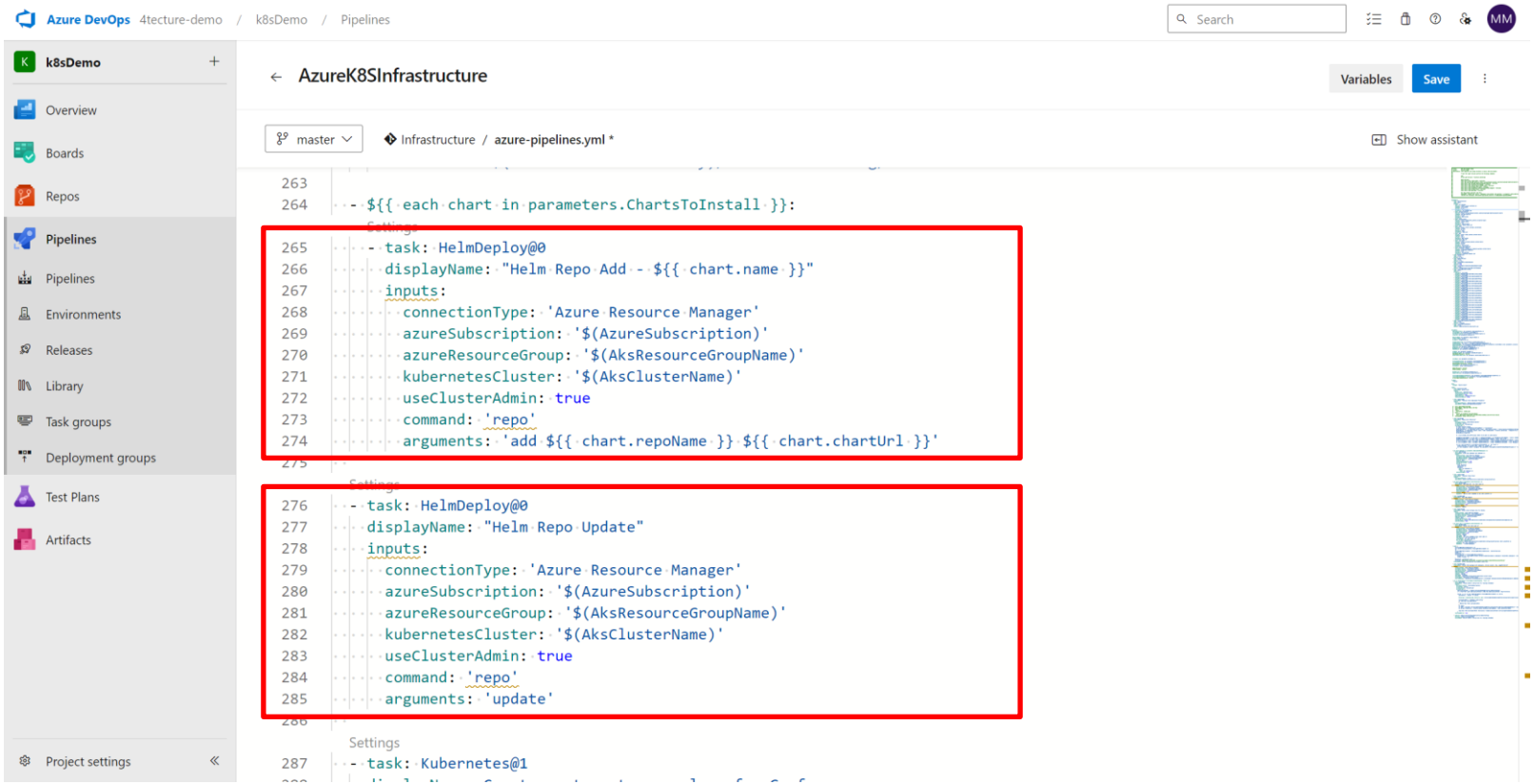
Parameter with all charts



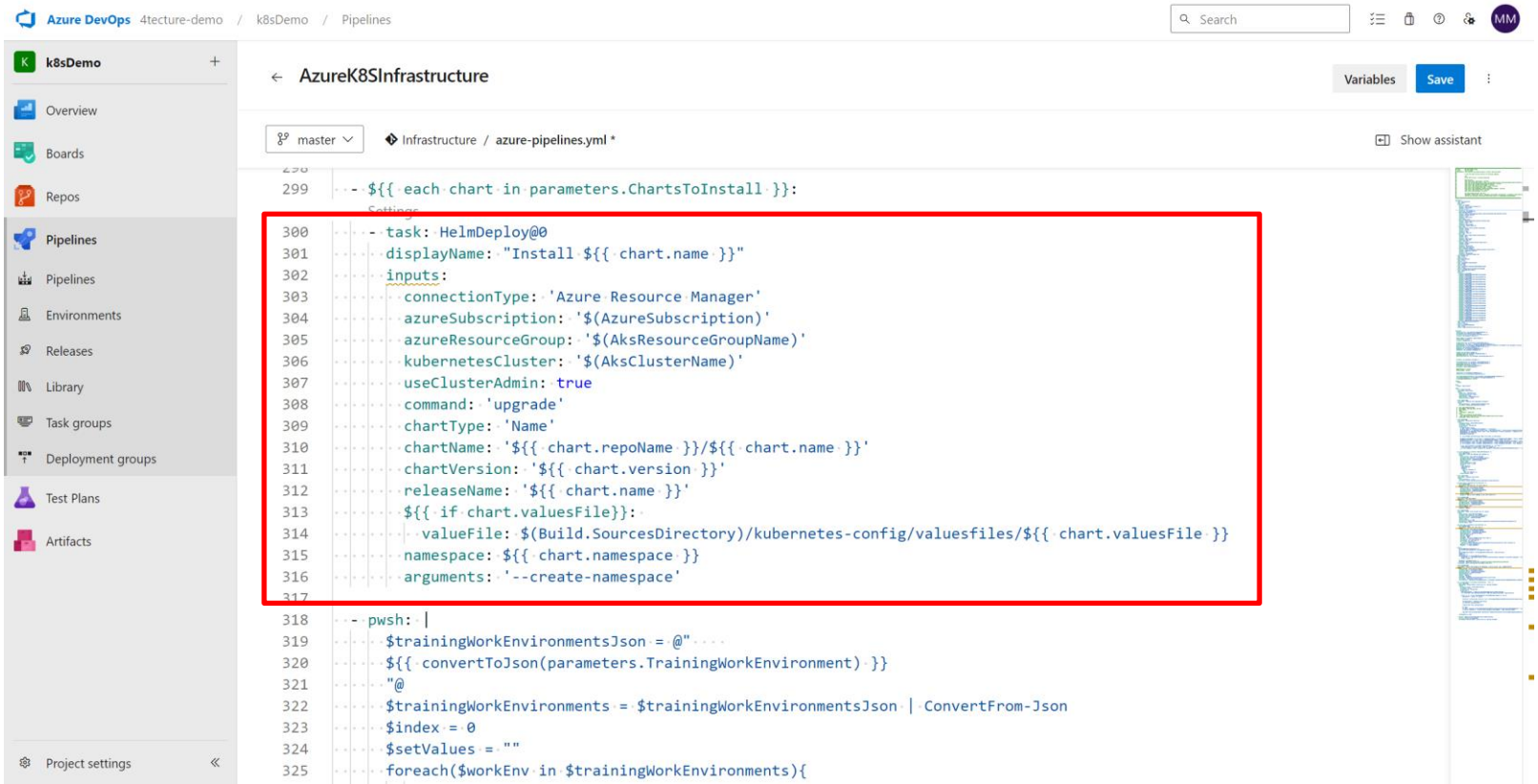
Configuration as Code



Add and update Helm Repos



Install helm charts



Conclusion

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

Conclusion

- Both provide a state-of-the-art deployment
- Git is the source of truth
- We use
  - Azure Pipelines: Classic Development to Deployment (CI/CD)
  - Flux: disconnected from development / prod cluster deployment


Q & A
4tecture
empower your software solutions

Thank you for your attention!

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

4tecture GmbH
Industriestrasse 25
CH-8604 Volketswil

+41 44 508 37 00
info@4tecture.ch
www.4tecture.ch

Marc Müller
Principal Consultant

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

4tecture
empower your software solutions

ProAuth

ReaFx


4tecture©
empower your software solutions