Level Up your Kubernetes Scaling with KEDA

Wolfgang Ofner

Agenda

- Übersicht Architektur in SW Projekten
- Einführung in KEDA
- Skalierung anhand von Messages in Azure Service Bus Queue
- Fazit zu KEDA
- Q&A

About Me

Senior Software Architekt, bbv Software Services, Zürich

Consultant und Trainer

Fokus auf Azure, Kubernetes, DevOps and .NET

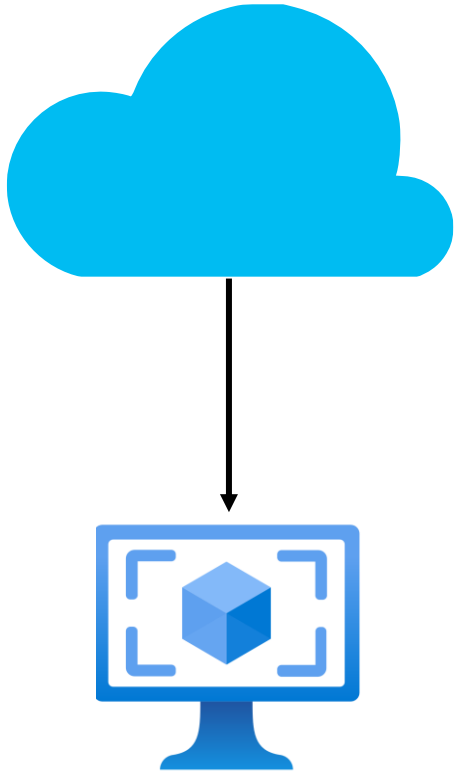
https://programmingwithwolfgang.com

https://www.linkedin.com/in/wolfgangofner

https://twitter.com/wolfgang_ofner

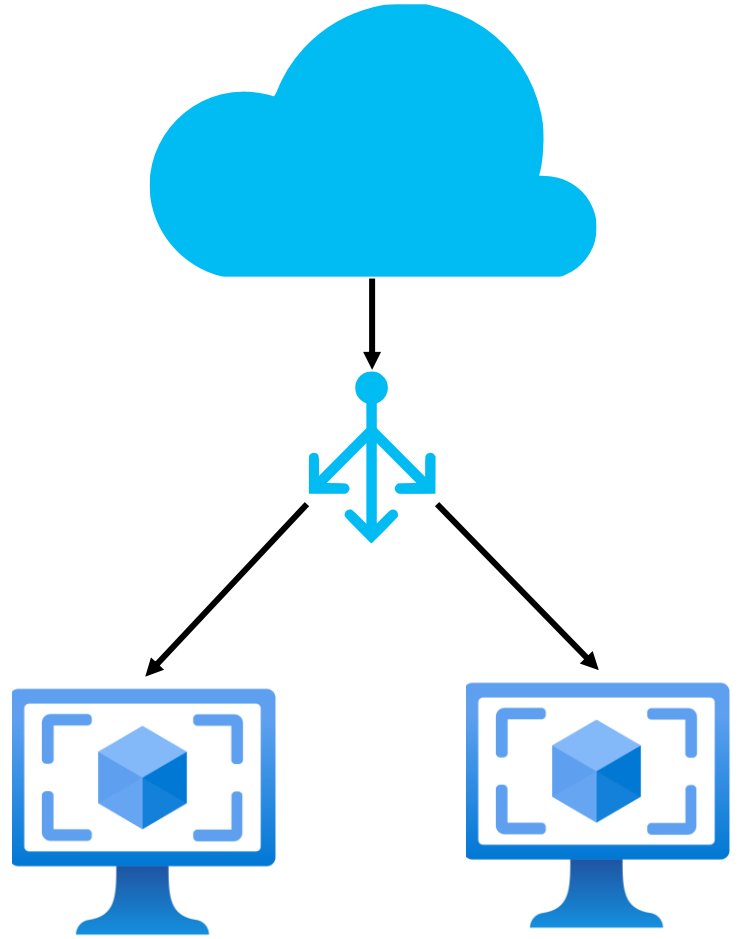
On-Premises Architektur

- Server – Client Architektur
- Begrenzte Anzahl an Clients
- Keine Redundanzen
- Keine hohe Verfügbarkeit



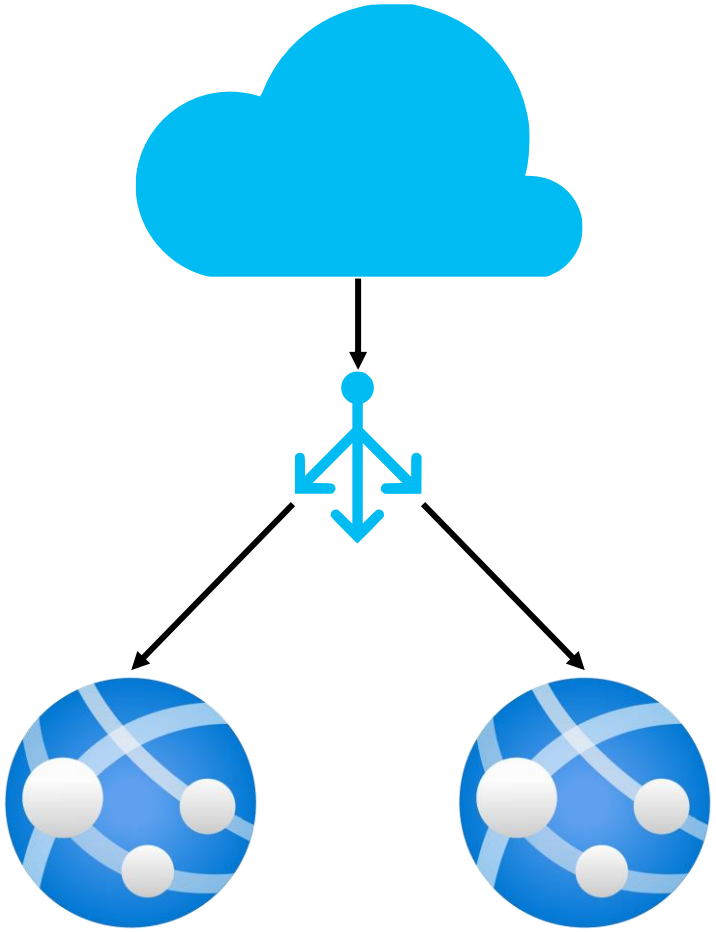
On-Premises Architektur

- Statisches load-balancing
- Neue VMs müssen manuell erstellt werden
- Teure on-premises Hardware

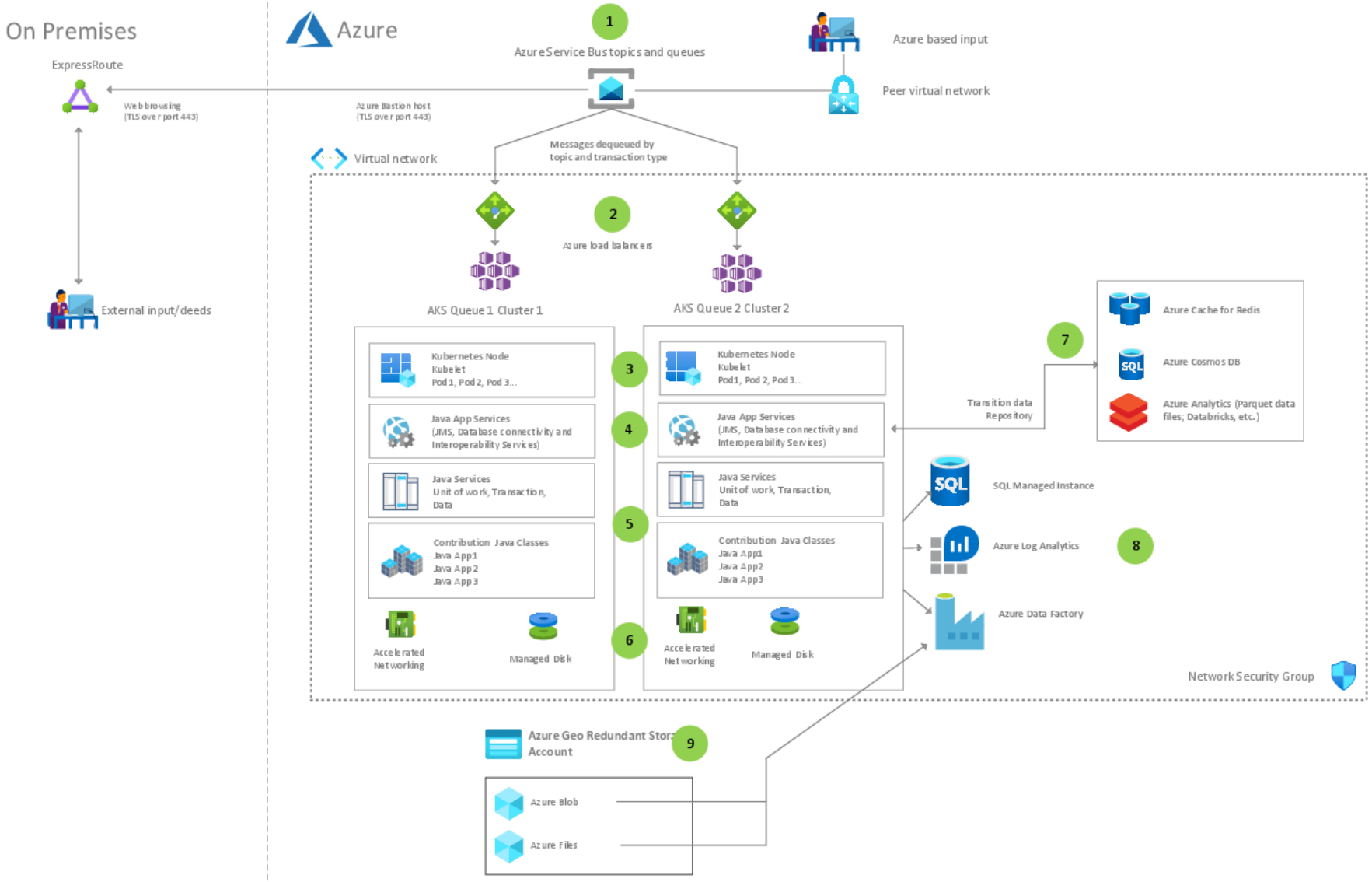


Cloud Architektur

- Skalierung anhand von CPU oder RAM
- Neue Hardware wird automatisch hinzugefügt (Scale-out)
- Pay as you go



Architektur in modernen Systemen



Kubernetes

- Moderne Systeme werden immer komplexer
- Horizontal Pod Autoscaler (HPA)
  - Skalierung anhand von CPU und/oder RAM
- Abhängig von externen Komponenten
- Anwendungen müssen auf Events reagieren
  - Datenbanken
  - Service Bus
  - Streams
  - Logs

Horizontal Pod Autoscaler (HPA)

- Skaliert Deployments oder StatefulSets
- Erstellt oder entfernt Pods
- Skaliert anhand der CPU oder RAM Auslastung

Horizontal Pod Autoscaler Konfiguration

```yaml
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: customerapi
  namespace: customerapi-test
spec:
  maxReplicas: 10
  minReplicas: 1
  averageCpuUtilization: 50
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: customerapi
```

```yaml
behavior:
  scaleDown:
    policies:
    - type: Pods
      value: 4
      periodSeconds: 60
    - type: Percent
      value: 10
      periodSeconds: 60
    selectPolicy: Min
```

```yaml
  scaleUp:
    policies:
    - type: Pods
      value: 5
      periodSeconds: 60
    - type: Percent
      value: 12
      periodSeconds: 60
    selectPolicy: Max
```

Horizontal Pod Autoscaler

- Skaliert Deployments oder StatefulSets
- Erstellt oder entfernt Pods
- Skaliert anhand der CPU oder RAM Auslastung
- Skalierung mit “Custom Metrics”
  - Abfrage von selbst definierten Metriken von der Kubernetes API
  - Prometheus
  - Requests pro Sekunde

Limitierung des HPAs

- Black Friday
- Tausende Bestellungen gespeichert in einer Queue
- Skalierung anhand der CPU oder RAM hilft nicht
- Keine Möglichkeit zur Skalierung

KEDA – Kubernetes Event-driven Autoscaling

- Kubernetes Event-driven Autoscaling
- Open source
- CNCF Projekt
- Maintainer
  - Docplanner Tech
  - Microsoft
  - Red Hat

KEDA – Kubernetes Event-driven Autoscaling

- ~48 built-in Scaler
  - Apache Kafka
  - Azure Blob Storage
  - Azure Monitor
  - Azure Service Bus
  - Elastic Search
  - MongoDB
  - Prometheus
  - Redis Streams

KEDA Use Cases

- Skalierung anhand von externen Events
- “Scale to 0”
  - Serverless Architektur on-premises
  - Nachbau der Azure Functions Architektur
  - Besserer Ressourcenverbrauch

KEDA Installation

- Installation via Helm Charts
- Namespace: keda

KEDA Installation

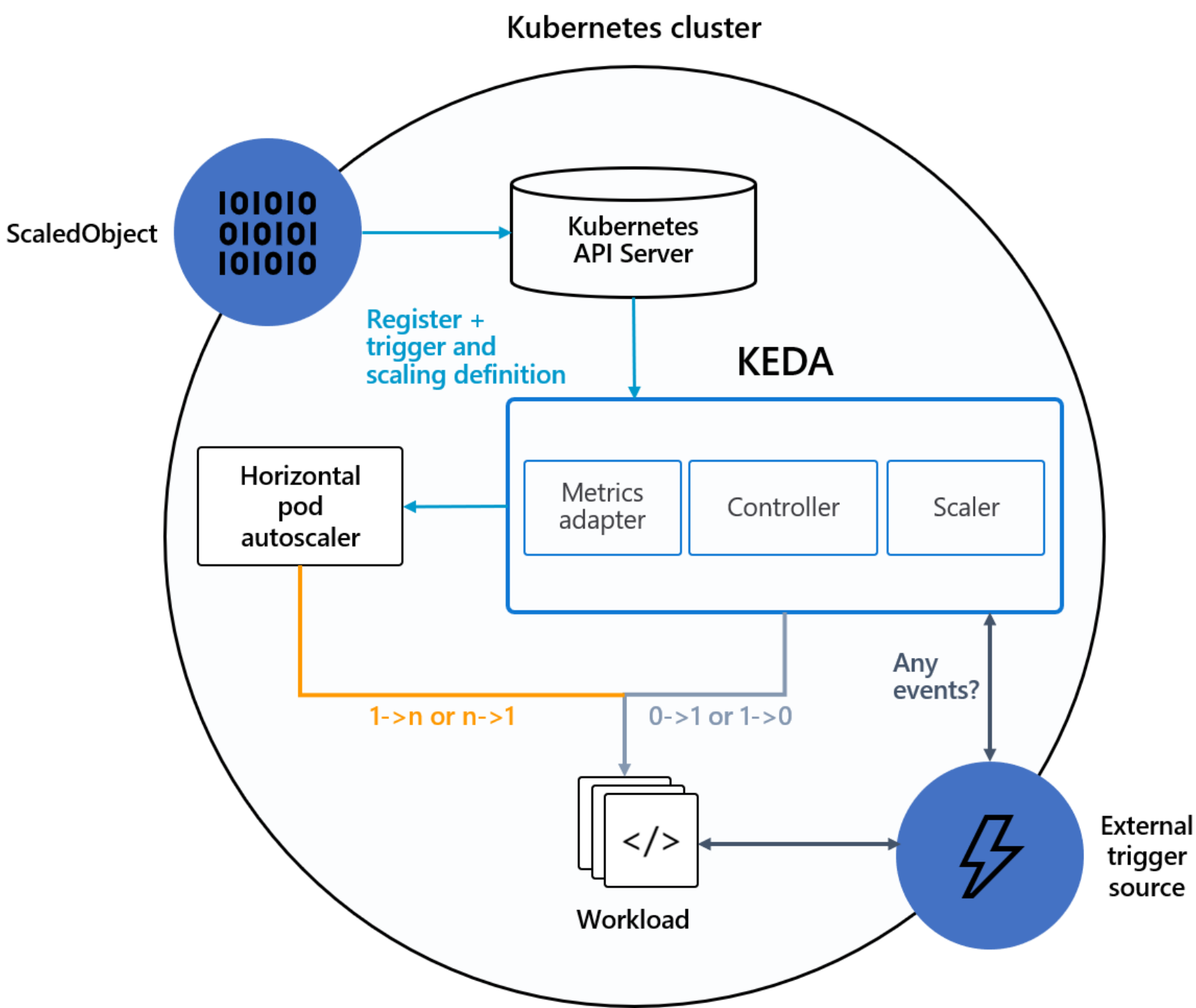
```
kubectl create namespace keda

helm repo add kedacore https://kedacore.github.io/charts
helm repo update
helm install keda kedacore/keda --namespace keda
```

KEDA Ressourcen

```
PS C:\Users\Wolfgang> kubectl get all -n keda
NAME                                                READY   STATUS    RESTARTS   AGE
pod/keda-operator-5748df494c-mxz9p                  1/1     Running   0          124m
pod/keda-operator-metrics-apiserver-cb649dd48-jjhpc 1/1     Running   0          124m

NAME                                       TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)          AGE
service/keda-operator-metrics-apiserver    ClusterIP   10.0.241.182   <none>        443/TCP,80/TCP   124m

NAME                                           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/keda-operator                  1/1     1            1           124m
deployment.apps/keda-operator-metrics-apiserver 1/1    1            1           124m

NAME                                                      DESIRED   CURRENT   READY   AGE
replicaset.apps/keda-operator-5748df494c                  1         1         1       124m
replicaset.apps/keda-operator-metrics-apiserver-cb649dd48 1         1         1       124m
```

KEDA Architektur

- 2 Komponenten für KEDA
  - Agent/Operator
  - Metrics Server
- Verwendet den HPA zum Skalieren
- Integration ohne Anpassungen der bestehenden Anwendungen


Kubernetes cluster
ScaledObject
Kubernetes API Server
Register + trigger and scaling definition
KEDA
Horizontal pod autoscaler
Metrics adapter
Controller
Scaler
1->n or n->1
0->1 or 1->0
Any events?
Workload
External trigger source

KEDA Architektur

- 2 Komponenten für KEDA
  - Agent/Operator
  - Metrics Server
- Verwendet den HPA zum Skalieren
- Integration ohne Anpassungen der bestehenden Anwendungen
- 2 Custom K8s Ressourcen für den Scaler
  - ScaledObject
  - TriggerAuthentication

ScaledObject

```yaml
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: kedademoapi-scaler
```

```yaml
spec:
  scaleTargetRef:
    name: kedademoapi
  minReplicaCount: 0
  maxReplicaCount: 10
  pollingInterval: 30
  cooldownPeriod: 30
```

```yaml
triggers:
  - type: azure-servicebus
    metadata:
      queueName: KedaDemo
      queueLength: '5'
    authenticationRef:
      name: trigger-authentication-kedademoapi
```

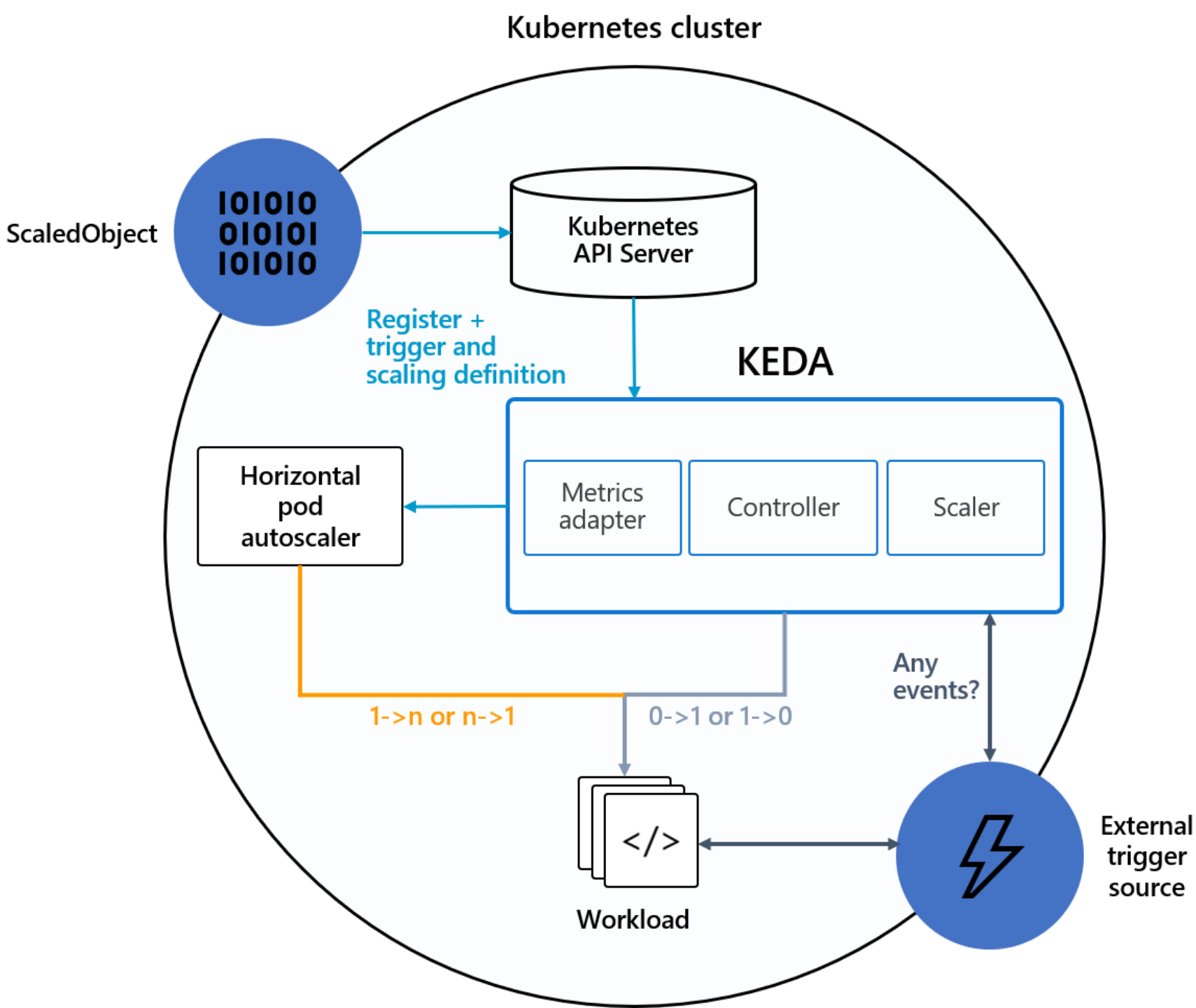
TriggerAuthentication

```yaml
apiVersion: keda.sh/v1alpha1
kind: TriggerAuthentication
metadata:
  name: trigger-authentication-kedademoapi
spec:
  secretTargetRef:
  - parameter: connection
    name: kedademoapi-connectionstrings
    key: AzureServiceBus__ConnectionString
```

Kubernetes Secret

```
PS C:\Users\Wolfgang> kubectl get secrets
NAME                                                  TYPE                                  DATA   AGE
default-token-88lzb                                   kubernetes.io/service-account-token   3      26h
kedademoapi-connectionstrings                         Opaque                                1      26h
kedademoapi-tls                                       kubernetes.io/tls                     2      26h
sh.helm.release.v1.kedademoapi-kedademoapi-test.v1    helm.sh/release.v1                    1      26h
sh.helm.release.v1.kedademoapi-kedademoapi-test.v2    helm.sh/release.v1                    1      22h
PS C:\Users\Wolfgang> kubectl describe secret kedademoapi-connectionstrings
Name:         kedademoapi-connectionstrings
Namespace:    kedademoapi-test
Labels:       app.kubernetes.io/managed-by=Helm
Annotations:  meta.helm.sh/release-name: kedademoapi-kedademoapi-test
              meta.helm.sh/release-namespace: kedademoapi-test

Type:  Opaque

Data
====
AzureServiceBus__ConnectionString:  165 bytes
```

Kubernetes Secret


Kubernetes cluster
ScaledObject
Kubernetes API Server
Register + trigger and scaling definition
KEDA
Horizontal pod autoscaler
Metrics adapter
Controller
Scaler
1->n or n->1
0->1 or 1->0
Any events?
Workload
External trigger source

Demo

Demo

- Skalierung anhand von Message in einer Azure Service Bus Queue
- “Scale to 0”
- Skalierung auf 1

Skalierung mit KEDA

```
PS C:\Users\Wolfgang> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
kedademoapi-6f986c4b76-hvbrq   1/1     Running   0          13m
```

kedademo (wolfgangkedademo/kedademo) | Service Bus Explorer

Service Bus Queue

Search (Ctrl+/)

Overview

Access control (IAM)

Diagnose and solve problems

Settings

Shared access policies

Service Bus Explorer (preview)

Properties

Locks

Refresh

Authentication type

Access key | Active Directory

Send | Receive | Peek

Receive performs a destructive read (ReceiveAndDelete) from Queue ***kedademo***. from the Queue. Messages shown here are no longer stored.

Active	Dead-Lettered	Scheduled
1 MESSAGES	0 MESSAGES	0 MESSAGES

Please Select Queue or DeadLetter

(•) Queue ( ) DeadLetter

POST /v1/ServiceBusProcessing Action to add new messages to the queue.

Parameters

Name	Description
numberOfQueueItems integer($int32) *(query)*	270

Execute

Responses

Curl

```
curl -X 'POST' \
  'https://test.kedademo.programmingwithwolfgang.com/v1/ServiceBusProcessing?numberOfQueueItems=270' \
  -H 'accept: */*' \
  -d ''
```

Request URL

```
https://test.kedademo.programmingwithwolfgang.com/v1/ServiceBusProcessing?numberOfQueueItems=270
```

Server response

Code	Details

200	Response headers
	`content-length: 0`
	`date: Fri,18 Feb 2022 15:45:21 GMT`
	`strict-transport-security: max-age=15724800; includeSubDomains`

Scale-Out

```
PS C:\Users\Wolfgang> kubectl get pods --sort-by=.status.phase
NAME                                READY   STATUS    RESTARTS   AGE
kedademoapi-6f986c4b76-9gnd7   0/1     Pending   0          3m10s
kedademoapi-6f986c4b76-cl4p6   0/1     Pending   0          3m10s
kedademoapi-6f986c4b76-w8fs5   0/1     Pending   0          2m55s
kedademoapi-6f986c4b76-z8dkd   0/1     Pending   0          3m10s
kedademoapi-6f986c4b76-jzxp7   0/1     Pending   0          3m10s
kedademoapi-6f986c4b76-l59bb   0/1     Pending   0          3m25s
kedademoapi-6f986c4b76-pb5z7   0/1     Pending   0          2m55s
kedademoapi-6f986c4b76-srkdj   1/1     Running   0          3m25s
kedademoapi-6f986c4b76-h6gbz   1/1     Running   0          3m25s
kedademoapi-6f986c4b76-hvbrq   1/1     Running   0          18m
```

GET /v1/ServiceBusProcessing Action to start processing the queue items.

Parameters

No parameters

Execute

Responses

Curl

```
curl -X 'GET' \
  'https://test.kedademo.programmingwithwolfgang.com/v1/ServiceBusProcessing' \
  -H 'accept: application/json'
```

Request URL

```
https://test.kedademo.programmingwithwolfgang.com/v1/ServiceBusProcessing
```

Server response

Code	Details
200	**Response body** 271 **Response headers** content-type: application/json; charset=utf-8 date: Fri,18 Feb 2022 15:51:31 GMT strict-transport-security: max-age=15724800; includeSubDomains

Scale to 0

```
PS C:\Users\Wolfgang> kubectl get pods
No resources found in kedademoapi-test namespace.
```

Scale to 0


503 Service Temporarily Unavaila
test.kedademo.programmingwithwolfgang.com/index.html
503 Service Temporarily Unavailable
nginx

kedademo (wolfgangkedademo/kedademo) | Service Bus Explorer

Service Bus Queue

Search (Ctrl+/)

Overview

Access control (IAM)

Diagnose and solve problems

Settings

Shared access policies

Service Bus Explorer (preview)

Properties

Locks

Refresh

Authentication type

Access key | Active Directory

Send | Receive | Peek

Send Message to Queue ***kedademo***

Content Type *

Text/Plain

```
new message
```

kedademo (wolfgangkedademo/kedademo) | Service Bus Explorer

Service Bus Queue

Search (Ctrl+/)

Overview

Access control (IAM)

Diagnose and solve problems

Settings

Shared access policies

Service Bus Explorer (preview)

Properties

Locks

Refresh

Authentication type

Access key | Active Directory

Send | Receive | Peek

Receive performs a destructive read (ReceiveAndDelete) from Queue ***kedademo***. from the Queue. Messages shown here are no longer stored.

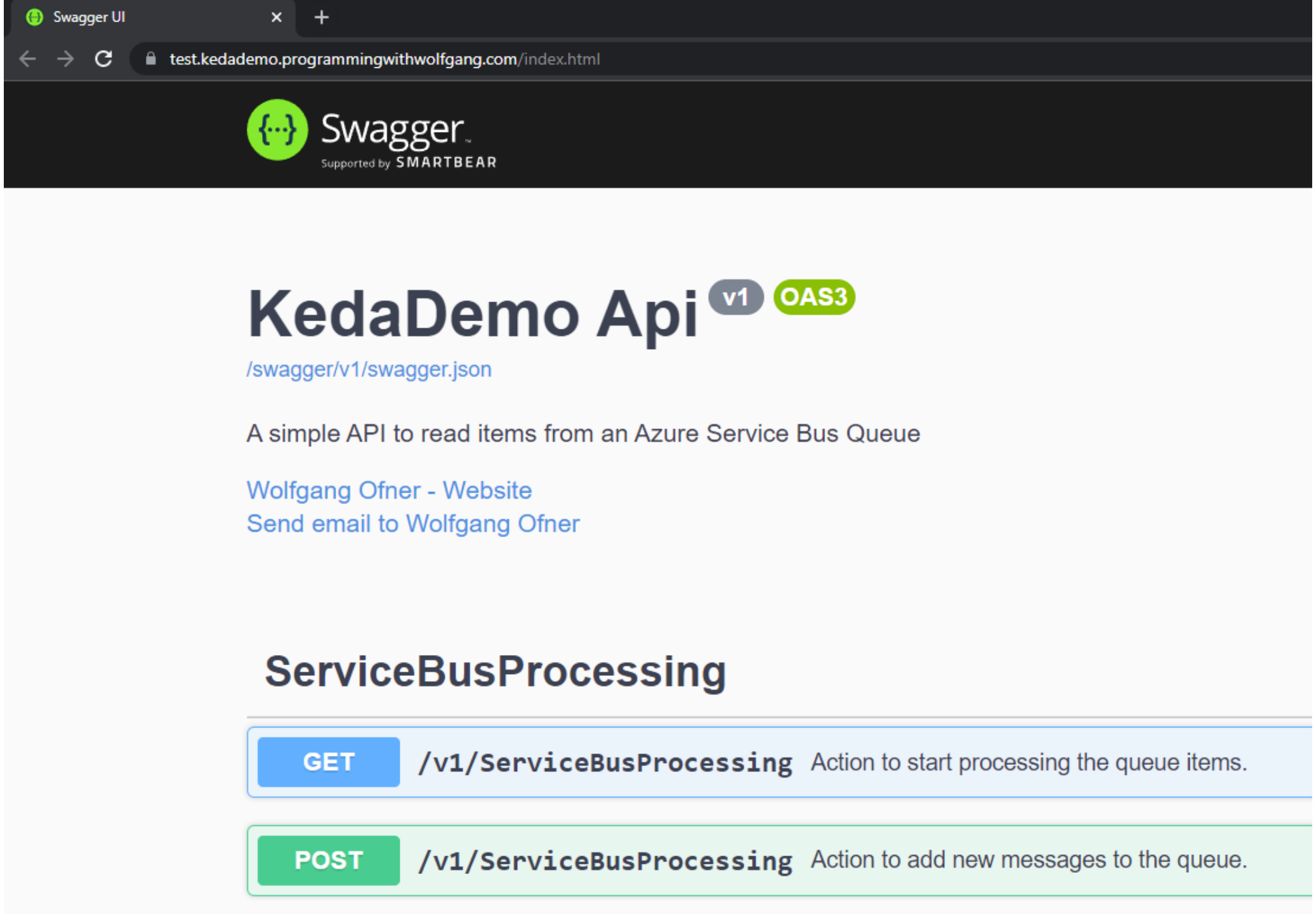
Active	Dead-Lettered	Scheduled
1 MESSAGES	0 MESSAGES	0 MESSAGES

Please Select Queue or DeadLetter

(•) Queue ( ) DeadLetter

Skalierung von 0 auf 1

```
PS C:\Users\Wolfgang> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
kedademoapi-6f986c4b76-b8pgj   1/1     Running   0          40s
```


Swagger UI
test.kedademo.programmingwithwolfgang.com/index.html
Swagger
Supported by SMARTBEAR
KedaDemo Api
v1
OAS3
/swagger/v1/swagger.json
A simple API to read items from an Azure Service Bus Queue
Wolfgang Ofner - Website
Send email to Wolfgang Ofner
ServiceBusProcessing
GET
/v1/ServiceBusProcessing
Action to start processing the queue items.
POST
/v1/ServiceBusProcessing
Action to add new messages to the queue.

KEDA Skalierung Logs

- keda-operator Pod schreibt Logs beim Auslösen einer Skalierung

Limitierung

- Scaler nicht verfügbar für verwendete Technologie
- Cluster hat zu wenig Ressourcen

Limitierung

Pods

Name	Labels	Ready	Phase	Restarts	Node
kedademoapi-6f986c4b76-2zfxc	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-6w9tc	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-777r8	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-9vs76	app:kedademoapi draft:draft-app 1+	1/1	Running	0	aks-nodepool1-35436033-vmss000000
kedademoapi-6f986c4b76-jdd8x	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-mdj62	app:kedademoapi draft:draft-app 1+	1/1	Running	0	aks-nodepool1-35436033-vmss000000
kedademoapi-6f986c4b76-qg298	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-rzgfm	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-s56q6	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-wb7rr	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>

Limitierung

Pods

Name	Labels	Ready	Phase	Restarts	Node
kedademoapi-6f986c4b76-2zfxc	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-6w9tc	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-777r8	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-9vs76	app:kedademoapi draft:draft-app 1+	1/1	Running	0	aks-nodepool1-35436033-vmss000000
kedademoapi-6f986c4b76-jdd8x	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-mdj62	app:kedademoapi draft:draft-app 1+	1/1	Running	0	aks-nodepool1-35436033-vmss000000
kedademoapi-6f986c4b76-qg298	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-rzgfm	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-s56q6	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>
kedademoapi-6f986c4b76-wb7rr	app:kedademoapi draft:draft-app 1+	0/1	Pending	0	<not scheduled>

Limitierung

Events

Message	Reason
0/1 nodes are available: 1 Insufficient cpu.	FailedScheduling

Limitierung

- Scaler nicht verfügbar für verwendete Technologie
- Cluster hat keine Ressourcen
  - Maximale Anzahl Replicas definieren
  - Cluster Auslastung überwachen
  - Azure Cluster Autoscaler

KEDA in Produktion

- Azure Container Apps verwendet KEDA
  - Serverless Container
- KEDA 1.0.0 wurde am 17. Nov 2019 veröffentlicht
- Derzeit 2.6.1

Ressourcen

- Demo Applikation
  - https://github.com/WolfgangOfner/MicroserviceDemo/tree/master/KedaDemoApi
- KEDA
  - https://keda.sh
- KEDA Github
  - https://github.com/kedacore/keda
- KEDA Architektur Screenshot
  - https://keda.sh/docs/2.6/concepts/#architecture

Q&A

Level Up your Kubernetes Scaling with KEDA

Wolfgang Ofner

https://programmingwithwolfgang.com

https://www.linkedin.com/in/wolfgangofner

https://twitter.com/wolfgang_ofner