

DEEPSHORE

DEEPTALK

Persistenz in Kubernetes

2020

Agenda

Recap

Motivation

PersistentVolumes, PersistentVolumeClaims

StorageClasses

Zusammenfassung

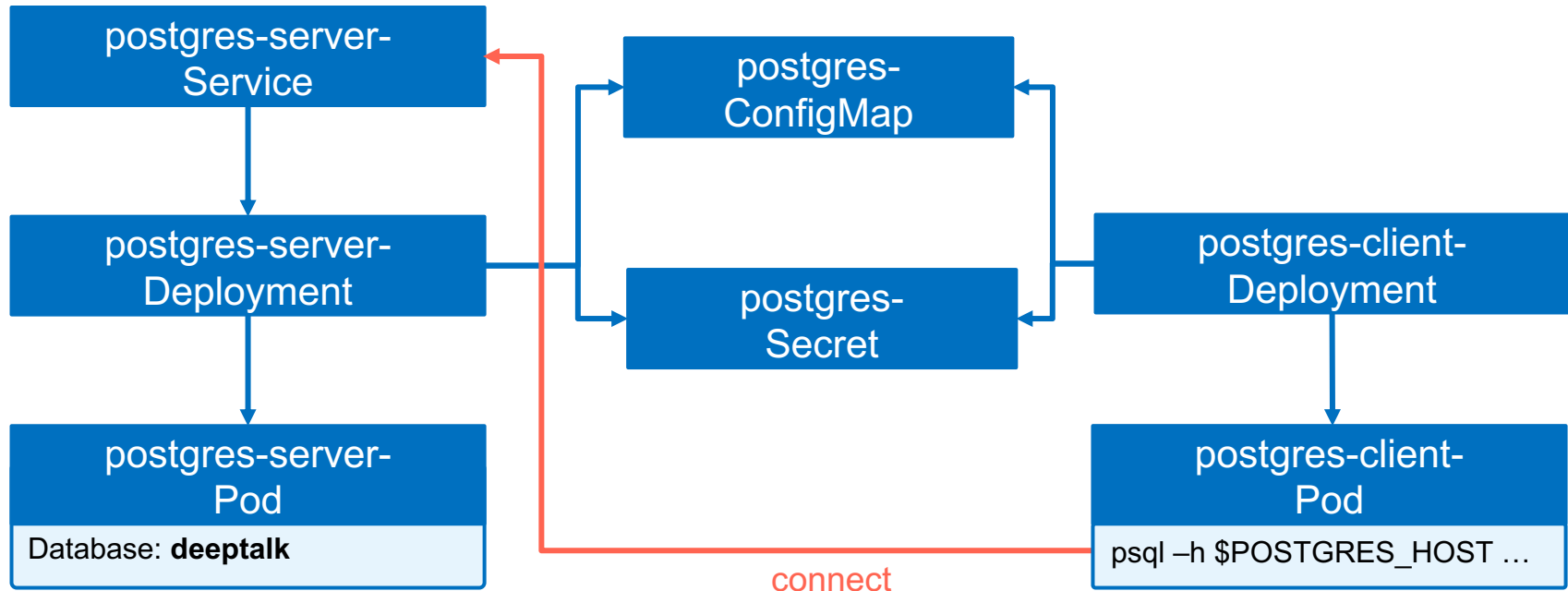
Recap – bisherige Deeptalks

- 1) Einstieg in Kubernetes mit Minikube
- 2) Deployments in Kubernetes
- 3) Von der Application zum Kubernetes-Deployment
- 4) ConfigMaps und Secrets

Aufzeichnungen:

https://deepshore.de/knowledge/_videos/deeptalk-*

Motivation: Pods sind flüchtig



Binding

Auswahlkriterien:

- Capacity
- Access Mode
- Volume Modes
- Storage Class
- Selector

PVC



accessModes:
- **ReadWriteMany**
resources:
requests:
storage: **1Gi**

PV



accessModes:
- **ReadWriteMany**
capacity:
storage: **5Gi**
hostPath:
path: /mnt/data

Dynamic Provisioning

Motivation

- automatisch
- bei Bedarf
- häufiges Erstellen und Löschen

Lösung

Verwendung von StorageClasses

StorageClass

provisioner

Name des Volume-Plugins

parameters

abhängig vom Provisioner

reclaimPolicy

Delete

Retain

volumeBindingMode

Immediate

WaitForFirstConsumer

Ablauf Dynamic Provisioning

Administrator

- 1) Legt StorageClass an

Entwickler

- 1) Erzeugt PVC mit obiger StorageClass
- 2) Provisioner erzeugt automatisch passendes PV
- 3) PVC wird an PV gebunden

Provisioner: Beispiele

internal

AWSElasticBlockStore

GCEPersistentDisk

AzureDisk

external

Longhorn

OpenEBS (MayaStor)

Plugin Interfaces

Container Storage Interface (CSI)

FlexVolume

Takeaways

- Pods sind i.A. flüchtig
- Persistenz in k8s
 - Entwickler formulieren PVCs
 - PVCs werden an PVs gebunden
- Static: PVs werden manuell erzeugt
- Dynamic: PVs werden von Provisioner erzeugt
- PVs clusterweit, PVCs namespace-bezogen
- Vielzahl an Provisioner/Volume-Plugins „in-tree“
- Erweiterbarkeit durch „out-of-tree“ Volume-Plugins

Misc

Materialien zum Talk

<https://github.com/deepshore/deeptalk>

Weiterführendes

Postgres: <https://www.postgresql.org/>

Kubernetes: <https://kubernetes.io/docs/home/>

Feedback , Anregungen, Themenvorschläge

florian.boldt@deepshore.de

malte.groth@deepshore.de

frederic.born@deepshore.de

Ausblick

Wie es weitergehen könnte:

- DBMS in k8s
- Zusammenspiel von PVCs und StatefulSets
- Storage Provisioning
- Backups von Resources und PersistentVolumes

Vorschläge?

DEEPSHORE

Vielen Dank.