Data Management in Kubernetes Using Kanister

Tom Manville | April 25th, 2018
yes*
Database Operations in Kubernetes

kanister.io
Agenda

• Background
  • DB in a container
  • DB in Kubernetes

• Kanister
  • Use Cases
  • Components
  • Walkthrough
  • Demo
whoami
DB in a Container

$ docker run -d \n  percona/percona-server:5.7
DB in a Container

```bash
$ docker run -d \
  -v /local/datadir:/var/lib/mysql \
percona/percona-server:5.7
```
Cloud Native

- Container Package
- Dynamically Managed
- Micro-Services Oriented
Raw results from the Kubernetes App Survey run by @mattfarina are out!

Quick glance:
- 75% of users have some kind of stateful workload!
- (Shocking!) 38% of users don’t use Helm
- 50% of users are using NoSQL or SQL databases
- No one uses Kompose
docs.google.com/spreadsheets/d…

10:50 AM - Apr 23, 2018

❤️ 4  🌐 See Niraj Tolia’s other Tweets
Requirements

https://kubernetes.io/
https://github.com/kubernetes/kubernetes

https://www.helm.sh/
https://github.com/kubernetes/helm
DB in Kubernetes

$ helm install stable/percona
Application Centric Use Cases

- Portability between cloud providers
- Testing on real data
  - Filtering
  - Masking
- Backup/Recover
  - Distributed point-in-time consistency
  - Incremental backups
Install Kanister

$ helm install stable/kanister-operator

$ helm repo add kanister http://charts.kanister.io
$ helm install kanister/kanister-mysql
   --set kanister.s3_bucket="mysql-backup-bucket"
   --set kanister.s3_api_key="${AWS_ACCESS_KEY_ID}"
   --set kanister.s3_api_secret="${AWS_SECRET_ACCESS_KEY}"
Kanister Components

API Objects
- ActionSet
- Blueprints
- Profiles (coming soon!)
  - Locations
  - Credentials

Controller
- Watches for new API objects
- Launches actions
- Creates events/logs status
An Abridged ActionSet

```yaml
spec:
  actions:
    - name: backup
      blueprint: mysql-blueprint
      object:
        kind: Deployment
        name: mysql-sakila
        namespace: video-store-app

status:
...
```
Blueprint: Backup

backup:
phases:
  - func: KubeExec
    args:
      - "{{ .Deployment.Namespace }}"
      - "{{ index .Deployment.Pods 0 }}"
      - kanister-sidecar
      - bash
      - -c
      - |
        s3_path="s3://${S3_BUCKET}/{{ .ArtifactsOut.mysqlCloudDump.path }}"
        s3_cmd=(aws "$S3_VERIFY_SSL" s3 cp - "$s3_path")
        mysqlDump --password="${MYSQL_ROOT_PASSWORD}" | gzip - | $s3_cmd[@]
Blueprint: Restore

restore:
  phases:
  - func: KubeExec
    args:
      - "{{ .Deployment.Namespace }}"
      - "{{ index .Deployment.Pods 0 }}"
      - kanister-sidecar
      - bash
      - -c
      - |
        s3_path="s3://${S3_BUCKET}/{{ .ArtifactsIn.mysqlCloudDump.path }}"
        s3_cmd=(aws "${S3_VERIFY_SSL}" s3 cp "${s3_path}" -)
        ${s3_cmd[@]} | gunzip -c - | mysql --password="${MYSQL_ROOT_PWD}"
Chaining ActionSets

$ kanctl perform restore --from backup-mysql-04-23
Database Operations in Kubernetes

kanister.io
Try Kanister Today!

• Site: https://kanister.io/
• Github: https://github.com/kanisterio/kanister
• Docs: https://docs.kanister.io/
• Slack: https://kasten.typeform.com/to/QBcw8T
Cloud Native Landscape

See the interactive landscape at landscape.cncf.io

Platforms

Certified Kubernetes - Distribution

Certified Kubernetes - Platform

Non-Certified Kubernetes

Portable/Container Service

Serverless

Kubernetes Certified Service Provider

Observability & Analysis

Monitoring

Logging

Tracing

Runtime

Cloud-Native Storage

Container Runtime

Cloud-Native Network

Host Management / Tooling

Infrastructure Automation

Container Registries

Secure Images

Key Management

Provisioning

Public

Private

This landscape is intended as a map through the previously uncharted terrain of cloud-native technologies. There are many routes to deploying a cloud-native application, with CNCF Projects representing a particularly well-traveled path.

github.com/cncf/landscape
Common Databases

Elasticsearch
Shows up either standalone or as a part of the ELK Stack

MySQL
Most common relational choice. Includes MariaDB, Vitess.

MongoDB
Cross-vertical NoSQL choice w/ large clusters

Postgres

Cassandra
Operated at Scale

Redis
First "stateful" application deployed but usually as cache