RUNNING MYSQL IN K8S
Mykola Marzhan
Has been developing deployment, update and monitoring systems since 2004.
WHAT IS A KUBERNETES OPERATOR?
WHY IS OPERATOR NEEDED?
WHY IS OPERATOR NEEDED?

➤ Automated provisioning
➤ High Availability and Failover
➤ Scaling
➤ Backups and Restores
➤ Updates
IN WHAT CASE SHOULD I RUN A DATABASE IN KUBERNETES?
WHAT IF I HAVE XXL SIZE?
Vitess is a database clustering system for horizontal scaling of MySQL through generalized sharding.
VITESS OPERATOR / PLANETSCALE

➢ Solution for Sharding
➢ Open Source, CNCF graduated project
➢ Outstanding level of engineering
   but solution is complex for non-experts
➢ Both Asynchronous and Semisynchronous replication supported
   High Aviability based on Orchestrator and replication
➢ Physical backups to S3, GCS, Ceph
RECOVERY POINT OBJECTIVE (RPO)

RECOVERY TIME OBJECTIVE (RTO)

SIMPLICITY = MAINTAINABILITY
## Replication

<table>
<thead>
<tr>
<th>Async</th>
<th>Semi-sync</th>
<th>Sync</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eventual delivery</td>
<td>Eventual delivery</td>
<td>Sync delivery</td>
</tr>
<tr>
<td>Source -&gt; Replica</td>
<td>Source &lt;-&gt; Replica</td>
<td>Members &lt;-&gt; Members</td>
</tr>
<tr>
<td>Replica pull binlogs</td>
<td>Replica pull binlogs</td>
<td>Push on TRX commit</td>
</tr>
<tr>
<td>Needs external failover handling (e.g. Orchestrator)</td>
<td>Replica ack after flush to disk (slow)</td>
<td>Ack from All (Galera) or Majority (GR)</td>
</tr>
<tr>
<td>Needs external failover handling (e.g. Orchestrator)</td>
<td>Needs external failover handling (e.g. Orchestrator)</td>
<td>Automatic handling of membership</td>
</tr>
</tbody>
</table>
## Replication

<table>
<thead>
<tr>
<th>Async</th>
<th>Semi-sync</th>
<th>Sync</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="Image" alt="Performance (write QPS)" /> - Best</td>
<td><img src="Image" alt="Performance (write QPS)" /> - Worst</td>
<td><img src="Image" alt="Performance (write QPS)" /> - Average</td>
</tr>
<tr>
<td><img src="Image" alt="Recovery point objective - Non-zero" /></td>
<td><img src="Image" alt="Recovery point objective - Zero" /></td>
<td><img src="Image" alt="Recovery point objective - Zero" /></td>
</tr>
<tr>
<td><img src="Image" alt="Recovery time objective - Tens of seconds" /></td>
<td><img src="Image" alt="Recovery time objective - Tens of seconds" /></td>
<td><img src="Image" alt="Recovery time objective - Seconds" /></td>
</tr>
<tr>
<td><img src="Image" alt="Complexity/Expertise - Low" /></td>
<td><img src="Image" alt="Complexity/Expertise - Low" /></td>
<td><img src="Image" alt="Complexity/Expertise - Medium" /></td>
</tr>
</tbody>
</table>
The MySQL Operator for Kubernetes is a solution for managing MySQL InnoDB Cluster setups inside Kubernetes.
MySQL Operator / Oracle

- Solution for MySQL InnoDB Cluster
- Open Source, can run Enterprise Edition
- Technology Preview, not ready for production
- Rightly designed synchronous replication utilized
- Logical backups to Oracle Cloud
MySQL Operator / Bitpoke

- Was known as Presslabs MySQL Operator
- Open Source
- Simple and reliable, production ready
- Both Asynchronous and Semisynchronous replication supported
  - High Availability based on Orchestrator and replication
- Physical backups to S3, GCS
The Kubernetes Operator for MySQL is a solution for managing Percona XtraDB Cluster setups inside Kubernetes.
Solution for Percona XtraDB Cluster

Open Source

Most featurefull database operator, Production Ready

Synchronous replication based on Galera Replication

Physical backups to S3, GCS
The Kubernetes Operator for MySQL is a solution for managing Percona Server setups inside Kubernetes.
Solution for Percona Server

Open Source

Technology Preview, not ready for production

Both Asynchronous and Semisynchronous replication supported

High Aviability based on Orchestrator and replication

Backup are coming
WHICH OPERATOR SHOULD BE USED IN PRODUCTION?

Which operator should be used?

Is sharding necessary?

Can multiple transactions be lost during a failover?

No

Yes

No

Yes

No

Yes