



Automate the operations of your MySQL fleet on the stack of your choice

Mohamed Wadie Nsiri
Alex Lutay



Agenda

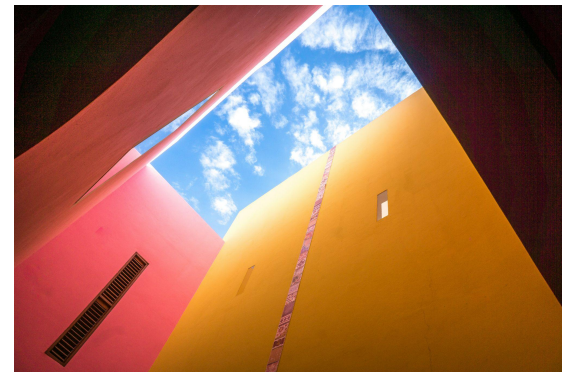
- 1 **Meet our multi-cloud operator**
- 2 **How it works ?**
- 3 **Available features**
- 4 **Planned features**
- 5 **Demo time**
- 6 **Let's get your questions answered**





Agenda

- 1 **Meet our multi-cloud operator**
- 2 How it works ?
- 3 Available features
- 4 Planned features
- 5 Demo time
- 6 Let's get your questions answered





A multi-cloud operator for MySQL

Operator for:

- [mysql-operator](#)
- [mysql-k8s-operator](#)



Canonical

Charmed Apps



Canonical

LXD



openstack®



kubernetes



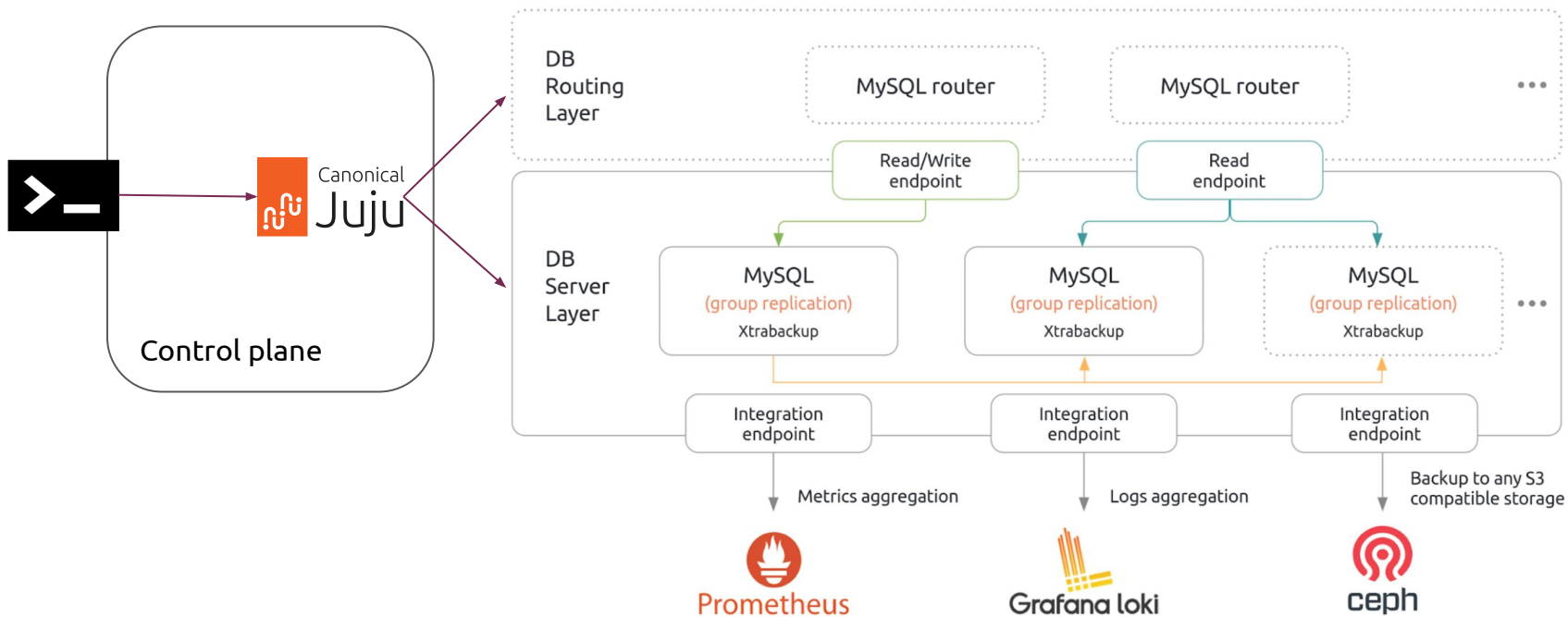
aws



Google Cloud



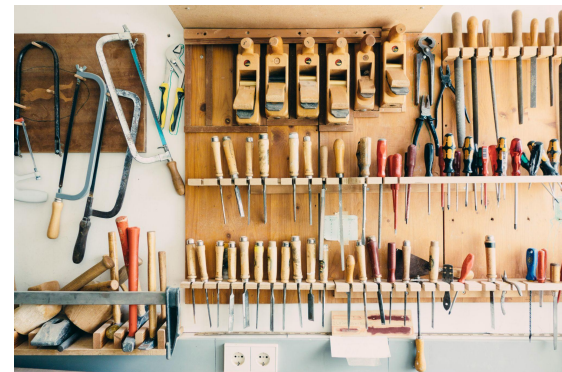
A multi-cloud operator for MySQL





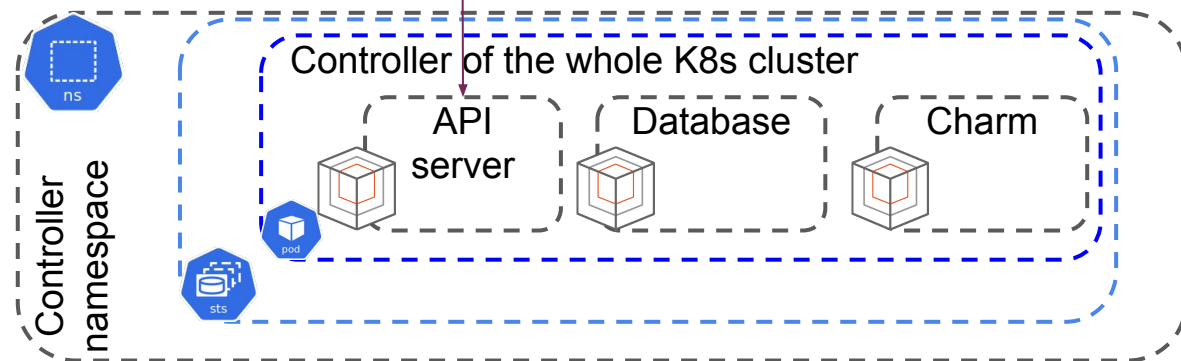
Agenda

- 1 Meet our multi-cloud operator
- 2 **How it works ?**
- 3 What can it do already
- 4 Planned features
- 5 Demo time
- 6 Let's get your questions answered



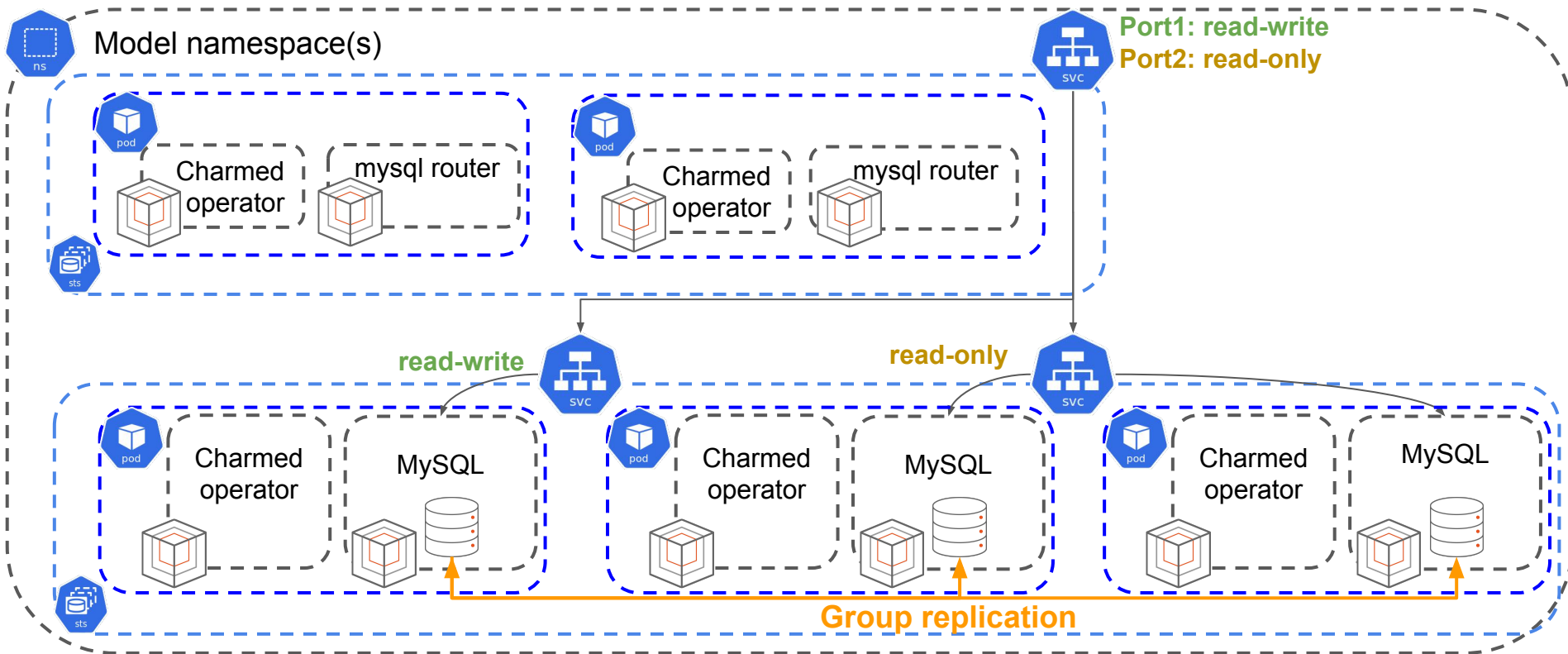


How it works ?





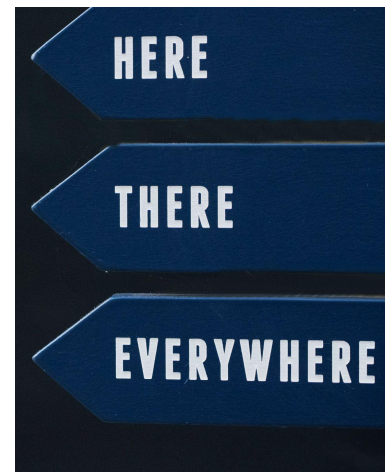
How it works ?





Agenda

- 1 Meet our multi-cloud operator
- 2 How it works ?
- 3 **Available features**
- 4 Planned features
- 5 Demo time
- 6 Let's get your questions answered





[Main] Available features

Deploy a cluster
based on group
replication

Scale-{up,down}
instances
(even to zero)

Automatic
failover and
self-healing

Full backup/restore
to/from S3
compatible storage

Encryption in
transit

Automatic config
validation and
tuning

Minor version
upgrades

Full observability
stack (metrics,
alerts, logs)



[Main] Available features

Deploying MySQL server

```
$ juju deploy mysql{,-k8s}
```

Horizontal scaling

```
$ juju {add-unit,scale-application} mysql{,-k8s} n
```

Deploying the database proxy

```
$ juju deploy mysql-router{-k8s,}
```

```
$ juju relate mysql-router{-k8s,} mysql{-k8s,}
```

Basic user management/password rotation

```
$ juju deploy data-integrator --config database-name=test
```

```
$ juju relate data-integrator mysql-router{-k8s,}
```

```
$ juju run data-integrator/leader get-credentials
```



[Main] Available features

Full backups

```
$ juju deploy s3-integrator
$ juju run s3-integrator/leader sync-s3-credentials access-key=<> secret-key=<>
$ juju config s3-integrator endpoint="<>" bucket="<>" path="<>"
$ juju relate s3-integrator mysql{,-k8s}
$ juju run mysql{,-k8s}/leader list-backups
$ juju run mysql{,-k8s}/leader create-backup
```

Restores

```
$ juju run mysql{,-k8s}/leader list-backups
$ juju run mysql{,-k8s}/leader restore backup-id=<>
```

Minor version upgrades

```
$ juju refresh mysql{,-k8s}
```



[Main] Available features

```
# Encryption on the wire
```

```
$ juju deploy tls-certificates-operator
```

```
$ juju config tls-certificates-operator
```

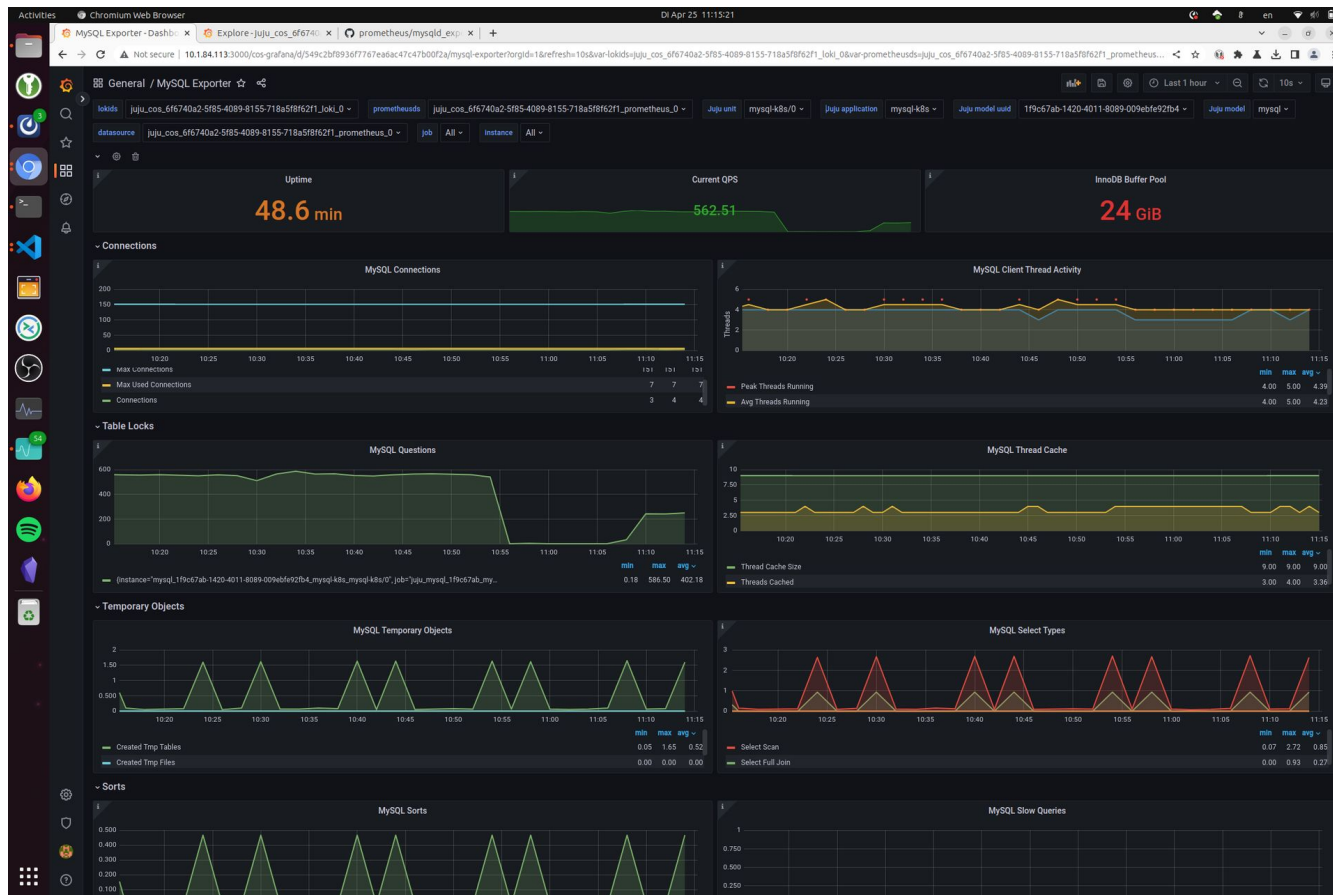
```
generate-self-signed-certificates="true" # For demonstration purposes only  
ca-common-name="Test CA"
```

```
$ juju relate tls-certificates-operator mysql{,-k8s}
```

```
$ juju relate tls-certificates-operator mysql-router{,-k8s}
```



[Main] Available features





Agenda

- 1 Meet our multi-cloud operator
- 2 How it works ?
- 3 Available features
- 4 **Planned features**
- 5 Demo time
- 6 Let's get your questions answered





Planned features

Incremental
backups

Deployment
across clouds

Advanced user
management

Out of the box
hardened set-up

ARM support

Storage layout
optimisation

Major version
upgrades

Schema
upgrades



Agenda

- 1 Meet our multi-cloud operator
- 2 How it works ?
- 3 Available features
- 4 Planned features
- 5 **Demo time**
- 6 Let's get your questions answered





Demo 1: Deploying MySQL on top of K8s

\$ # One command to deploy MySQL Cluster!

\$

Model	Controller	Cloud/Region	Version	SLA	Timestamp
simplicity	microk8s	microk8s/localhost	3.1.6	unsupported	06:01:16+01:00



Demo 2: Bootstrap magic ...

On Ubuntu 22.04+:

```
sudo snap install multipass
```

```
multipass launch --cpus 8 --memory 16G --disk 30G --name my-vm charm-dev
```

```
multipass shell my-vm
```

```
> juju deploy mysql-k8s -n 3 --trust
```



Canonical Multipass

[Install](#)

[Docs](#)

[Discourse](#)

Ubuntu VMs on demand for any workstation

Get an instant Ubuntu VM with a single command. Multipass can launch and run virtual machines and configure them with **cloud-init** like a public cloud.





Demo 2: Bootstrap magic ...


On Ubuntu 22.04+:

```
sudo snap install multipass
```

```
multipass launch --cpus 8 --memory 16G --disk 30G --name my-vm charm-dev  
multipass shell my-vm
```

```
> juju deploy mysql-k8s -n 3 --trust
```



 Canonical Multipass [Install](#) [Docs](#) [Discourse](#)

Ubuntu VMs on demand for any workstation

Get an instant Ubuntu VM with a single command. Multipass can launch and run virtual machines and configure them with **cloud-init** like a public cloud.



Demo 3: Real world use case

```
ubuntu@juju3: ~  
$ # day 0: We need a new WEB site with content in DB...  
$
```

```
ubuntu@juju3: ~  
Every 1.0s: juju status --color  
juju3: Sat Nov 4 02:21:57 2023  
ERROR model microk8s:admin/summit2023 not found
```



Demo 4: Upgrade under the load


```
$ # Let's deploy entire stack to play...
```

```
$
```

```
ERROR model microk8s:admin/test113 not found
```