DTCC

2017第八届中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2017

When TiDB meets Kubernetes

Dongxu Huang @ PingCAP

About me

- Dongxu Huang, Co-founder & CTO, PingCAP
- Infrastructure engineer / Hacker / Open source enthusiast
- Go / Rust / Python
- Codis / TiDB / TiKV

Agenda

- The problem we meet
- Brief introduction of Kubernetes / TiDB
- Operator saves the day
- Live without PersistentVolume
- The future

Cloud is the future.
But database maintenance still sucks.

Instance Specifications

DB Engine	postgres
License Model	postgresql-license
DB Engine Version	PostgreSQL 9.3.14-R1
DB Instance Class	db.r3.xlarge — 4 vCPU, 30.5 GiB R/
Multi-AZ Deployment	Yes
Storage Type	General Purpose (SSD)
Allocated Storage*	50 GB

Provisioning less than 100 GB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. **Click here** for more details.

Settings

DB Instance Identifier*	production
Master Username*	root
Master Password*	••••••••••••
Confirm Password*	••••••••••••

Select the DB instance class that allocates the computational, network, and memory capacity required by planned workload of this DB instance. Learn More.

Details:**db.r3.xlarge**

Type	Memory Optimized - Current Generation
vCPU	4 vCPU
Memory	30.5 GiB
EBS Optimized	500 Mbps
Network Performance	Moderate
Free Tier Eligible	No

* Required

Cancel Previous Next Step

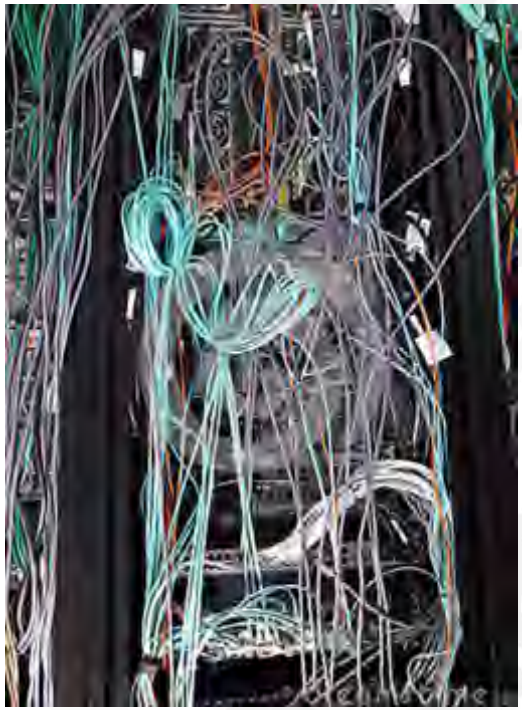
We were told everything would be scalable, easily.
But operating it makes it even harder.

Operating a single-node system

...A P2P distributed system

Unstable network infrastructure

×

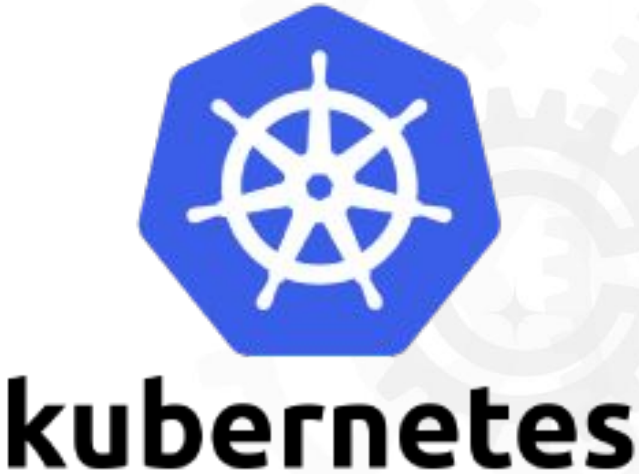


Services & Components


We are Google
resistance is futile
quickmeme.com

A brief introduction of Kubernetes

- Container-centric cluster management
- Service orchestrator
- Optimize use of hardware by using only the resources you need
- Auto deployment / Auto scaling / Auto healing



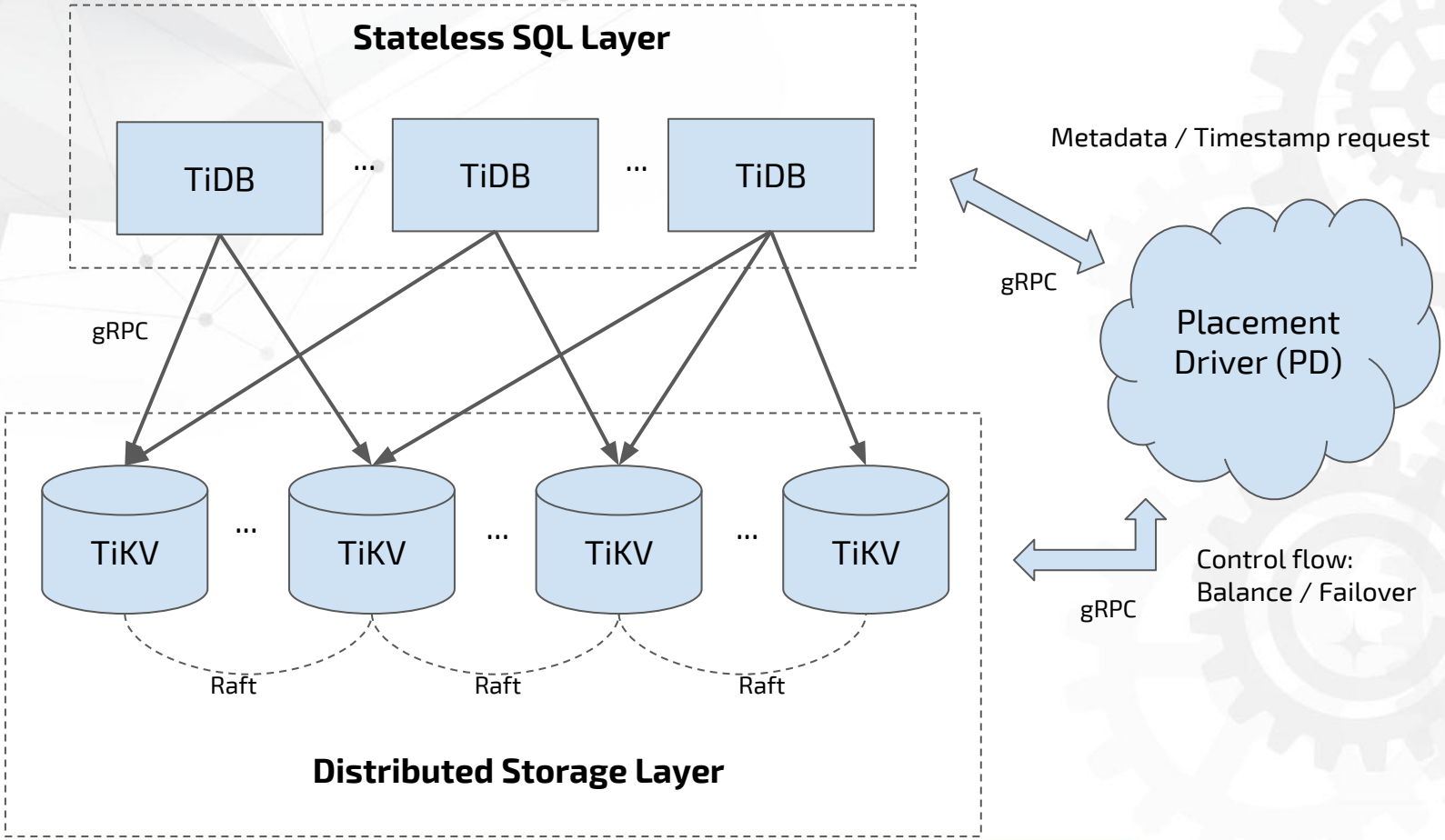
A brief introduction of TiDB

- SQL is necessary
- **Transparent sharding and data movement**
- 100% OLTP + 80% OLAP
  - Transaction + Complex query
- Compatible with MySQL, at most cases
- **24/7 availability, even in case of datacenter outages**
  - Thanks to Raft consensus algorithm
- Open source, of course.

TiDB

A Distributed SQL Database

A brief introduction of TiDB



The problem

- It's easy for stateless applications, but how about stateful?
  - Databases: MySQL / PG / TiDB
  - Coordination: Etcd / ZooKeeper
  - Streaming: Kafka
  - Big data: Hadoop / Ceph / GlusterFS
  - Search: ElasticSearch
  - ...
- Or even kubernetes itself.

What is the hard part?

Domain knowledge of the distributed system.

For example,
if you want to operate a redis cluster well, you must be a redis expert.

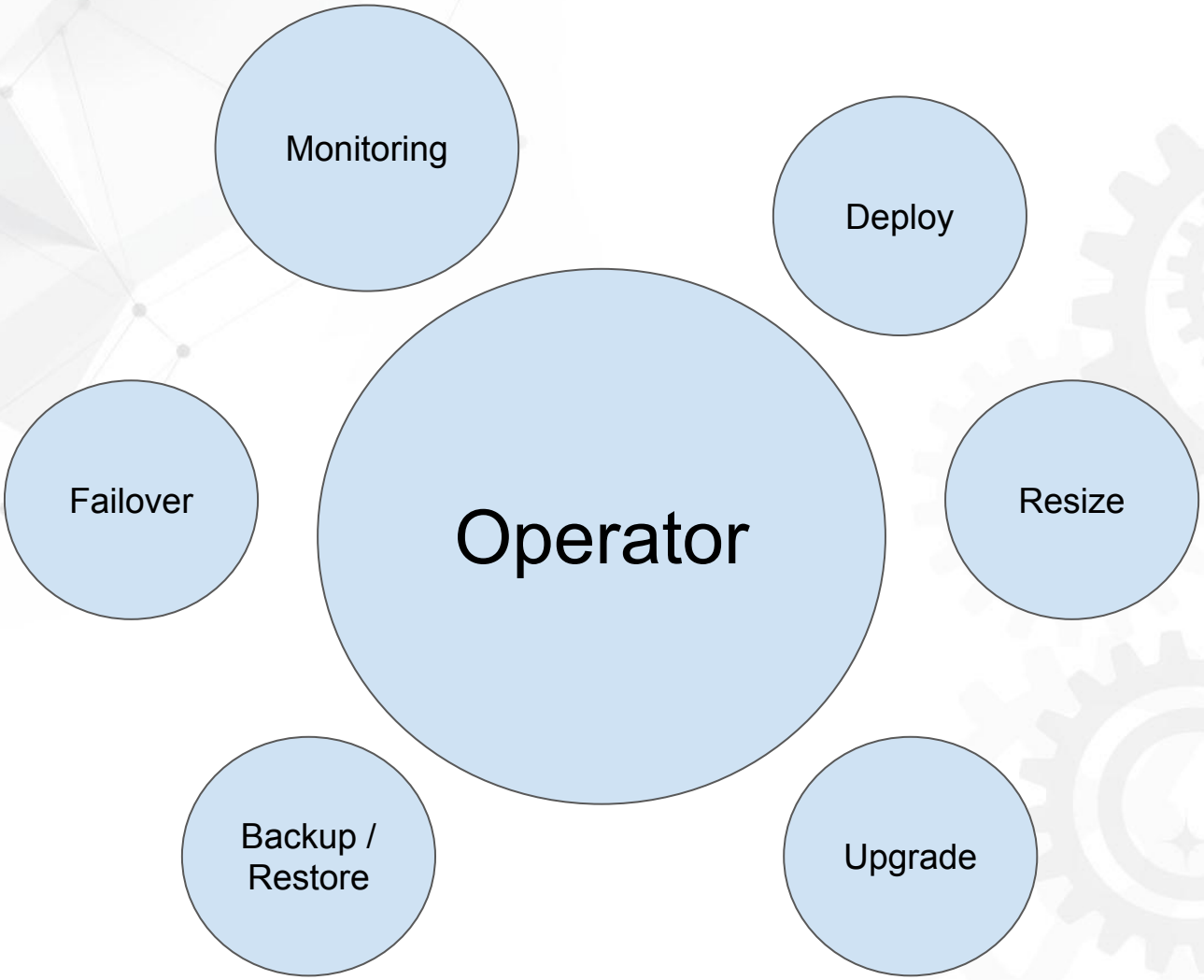
Operator saves the day

An **Operator** is software that encodes domain knowledge and **extends** the Kubernetes API through the ***third party resources*** mechanism, enabling users to create, configure, and manage applications.

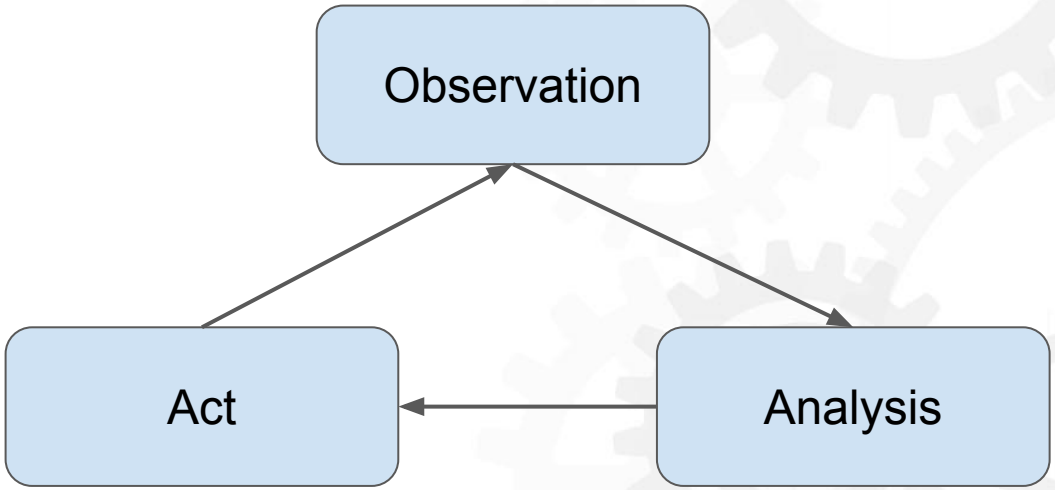
--- CoreOS

An Operator represents human operational knowledge in software, to reliably manage an application.

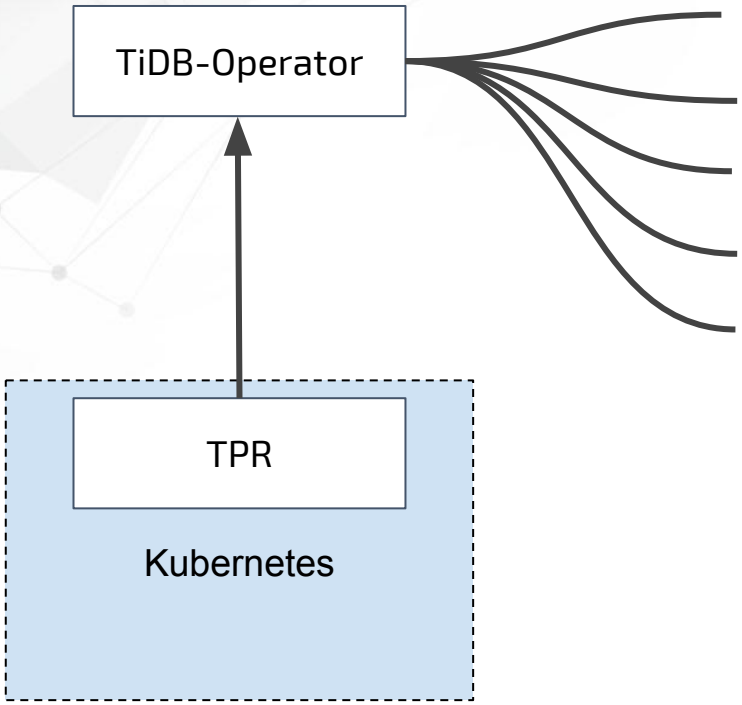
CoreOS


Monitoring
Deploy
Failover
Operator
Resize
Backup /
Restore
Upgrade

‘Self-driving’ mode

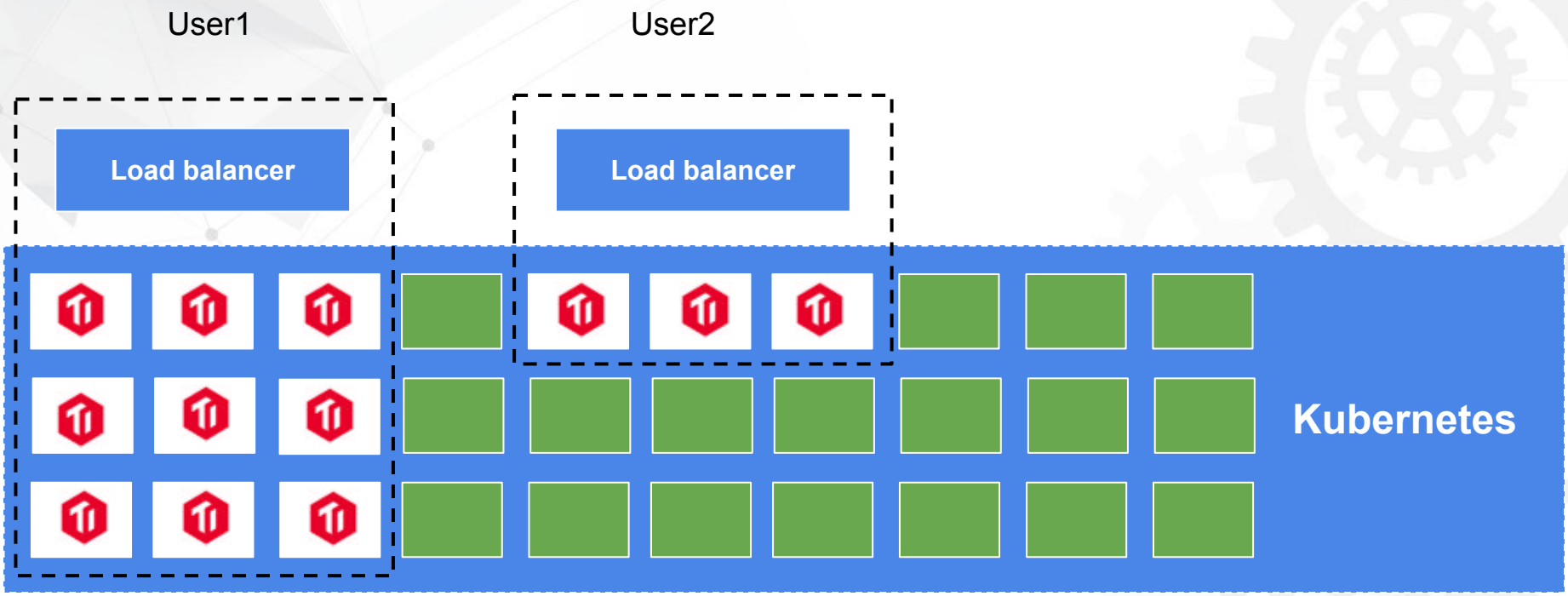


TiDB-Operator



- Create
- Rolling update
- Scale out
- Failover
- Backup/Restore

TiDB-Operator



The hardest part: Local storage resource

[Storage] Add support for persistent local storage #43640

New issue

Open msau42 opened this issue on 25 Mar · 4 comments

msau42 commented on 25 Mar • edited Member

Based on kubernetes/community#306

This issue is meant to track work items and help collaboration between the community on adding support for persistent local storage.

Note: Priority of these features can change based on the number of collaborators.

v1.7
General

- [ ] Add alpha feature gate for local persistent storage (Owner: **@msau** PR: #44640)
- [ ] Add e2es to verify the overall behavior (Owner: **@msau42, @jeffvance** PR: #44897)
- [x] Support for Local SSDs in GCE clusters (Owner: **@vishh** PR: #43726)

Volume plugin

- [ ] Add LocalStorage as a volume type in API (Owner: **@msau42** PR: #44640)
- [ ] Basic volume plugin that can mount/unmount LocalStorage PV to pods (Owner: **@msau42** #44897)
- [ ] Node e2e tests for mount/unmount (Owner:)

StorageClass changes

Assignees
vishh
msau42

Labels
None yet

Projects
None yet

Milestone
No milestone

Notifications
Subscribe
You're not receiving notifications from this thread.

4 participants

v1.9

- [ ] Dynamic provisioning for node local PVs
- [ ] Support for block level storage

 5

Chill out, bro…

Try to create a TPR to manage local storage resources

Under the hood

1. **Create a ConfigMap: tidb-storage**

```
nodes:
  - name: "172.17.4.101"
    directories:
      - "/tikv-storage-dir-1"
      - "/tikv-storage-dir-2"
  - name: "172.17.4.102"
    directories:
      - "/tikv-storage-dir-3"
      - "/tikv-storage-dir-4"
```

Under the hood

2. **Create a TPR: tidb-volume.pingcap.com/v1, Like:**

```
name: "172-14-4-101-tikv-dir1"
state: "binded"
podName: "tikv-1"
```

3. **Create a controller: volume-controller, notify configuration change of tidb-storage, generate tidb-volume resources**

Under the hood

4. **Add storage attribute to tidb-operator, so that tidb-operator would assign local storage resource to specific tikv instance**

```
...
storage:
  - "172.17.4.101:/tikv-storage-dir-1"
  - "172.17.4.102:/tikv-storage-dir-2"
  - "172.17.4.103:/tikv-storage-dir-4"
...
```

5. **Add a DaemonSet: tidb-storage-ds, maintain the lifetime of hostPath, when a tikv instance is offline, tidb-storage-ds would reclaim the storage resource.**

Open source....coming soon

pingcap / tidb-operator Private

TiDB-Operator

Tutorial

Create TiDB-Operator

```
$ kubectl create -f example/tidb-operator.yaml

$ kubectl get po
NAME                              READY     STATUS    RESTARTS   AGE
tidb-operator-1774570901-9vp2n    1/1       Running   0          3s
tidb-operator-139385347-k8lcp     1/1       Running   0          3s

$ kubectl get thirdpartyresource
NAME                        DESCRIPTION              VERSION(S)
tidb-cluster.pingcap.com    Managed tidb clusters    v1
```

Create a test cluster

```
$ kubectl create -f example/test-cluster.yaml

$ kubectl get po
NAME                              READY     STATUS    RESTARTS   AGE
test-cluster-pd-0000              2/2       Running   0          2m
```

The future

- ‘Self-driving mode’ for everything
  - Circuit breaker and MT is still important, as lifesaver.
- DB as a Service / Serverless
- Local storage isn’t necessary, maybe
  - Or another way, in-storage computing
  - Or maybe, both

Wrap it up

- Distributed system operation matters
- Kubernetes is the OS for the datacenter, but on the storage side, things become complicated
- Operator builds the bridge between domain knowledge and kubernetes, it's kindof batch script for DCOS.
- TiDB-operator provides the ability to set up/manage large cluster
  - We solve the local storage problem, little hacky, but it works

THANKS

SequeMedia 盛拓传媒 IT168.com ITPUB ChinaUnix