Vitess: Stateful Storage for K8s

Toliver Jue
Engineer, PlanetScale
What is Vitess?

- Sharding middleware for MySQL
- Massively scalable
- HA - Five-9’s Availability
- Cloud-native
Vitess stats

- Started 2010
- 8,000+ stars
- 16,000+ commits

Marquee adopters
- 100+ contributors
- 800+ Slack members
- 1000+ forks
Adoption

Top Charts

1. Cash App
   Send, Spend Money, Buy Bitcoin

2. YouTube: Watch, Listen, Stream
   Videos, Music and Live Streams

3. BURGER KING® App
   Coupons & Mobile Order

4. Instagram
   Photo & Video
Key adopters

3000+ Databases, 18000+ Tablets

JD.COM

25% Migrated to Vitess

slack

Cash App fully runs on Vitess

Square

All of advertising campaign management

Pinterest
The Community
Stateful Storage on Kubernetes ➞ Vitess
I'm always going to recommend people exercise extreme caution when running stateful workloads on Kubernetes. Most people who are asking "can I run stateful workloads on Kubernetes" don't have much experience with Kubernetes and often times the workload they are asking about.
ONE DOES NOT SIMPLY
MOVE MYSQL TO KUBERNETES
Vitess: History

- Started 2010
- Solved YouTube’s scaling problems
- Designed for Borg → Designed for K8s
Production Workloads

Oldest, since 2016

Hundreds of keyspaces

3000+ Databases, 18000+ Tablets
64-shards (on largest DB)

Migrates between clouds
MySQL 8.0
MYSQL MERELY ADOPTED THE CLOUD

VITESS WAS BORN IN IT
Small Workloads

- App server
- App server
- App server

Vitess

- Connection pooling
- Deadlines
- Hot row protection
- Row count limit
- Blacklisting

MainDB
Small Workloads

Scalability Guardrails

- Connection pooling
- Deadlines
- Hot row protection
- Row count limits
- Blacklisting
Growing Workloads

Vitess

app server
app server
app server

Replica routing
Load balancing
Master promotion with Orchestrator

Master
Replica
Growing Workloads

Vitess

Unified View
Keyspace Subsetting

app server

DB2
Replica

DB1
Replica
Growing Workloads

Sharding agnostic
Massive Workloads

Multi-zone Isolation

app server

Vitess

app server

app server

DB1:S1

DB1:S2

DB1:S3

DB2

Replica

Replica

Replica

Zone 1

Zone 2

Zone 3

Multi-zone Isolation
Demo
Sharded

- product
  - pid
  - description
- customer
  - cid
  - name
- orders
  - oid
  - cid
  - pid
  - mname
- merchant
  - name
select * from product

select pid, sum(price) as amount from orders

select * from orders where in_keyrange(mname, 'unicode_loose_md5', '-80')

customer

cid

cid

name

orders

oid

oid

pid

pid

mname

mname

merchant

product

id

id

select * from product
Use cases

- Materialized Views
- Real-time Rollups
- Resharding
- Backfilling of lookup vindexes
- Schema deployment
- Data migration
- Change notification
What’s next?

● Sign into slack channel at vitess.io
● Join WeChat group: (FIXME link/qr code)
● Bring up a cluster on laptop or kubernetes
  ○ https://vitess.io/docs/tutorials/
● Read code
  ○ https://github.com/vitessio/vitess
● Square Cash blog posts on sharding with Vitess
● 🎨 @toliver
● 🦅 @vitessio @planetscaledata