Vitess

How to Migrate a MySQL Database to Vitess

Morgan Tocker

Community Development Manager @ PlanetScale

Sugu Sougoumarane

Co-creator, CTO @ PlanetScale

What is Vitess?

Cloud Native Database

Massively Scalable

HA

Based on MySQL

Graduation!
Vitess
CLASS OF 2019
VITESS.IO
Vitess

Vitess Stats

Started
2010

Marquee
Adopters

100+
Contributors

★★★★★
8,000+
Stars

17,000+
Commits

1000+
Slack Members

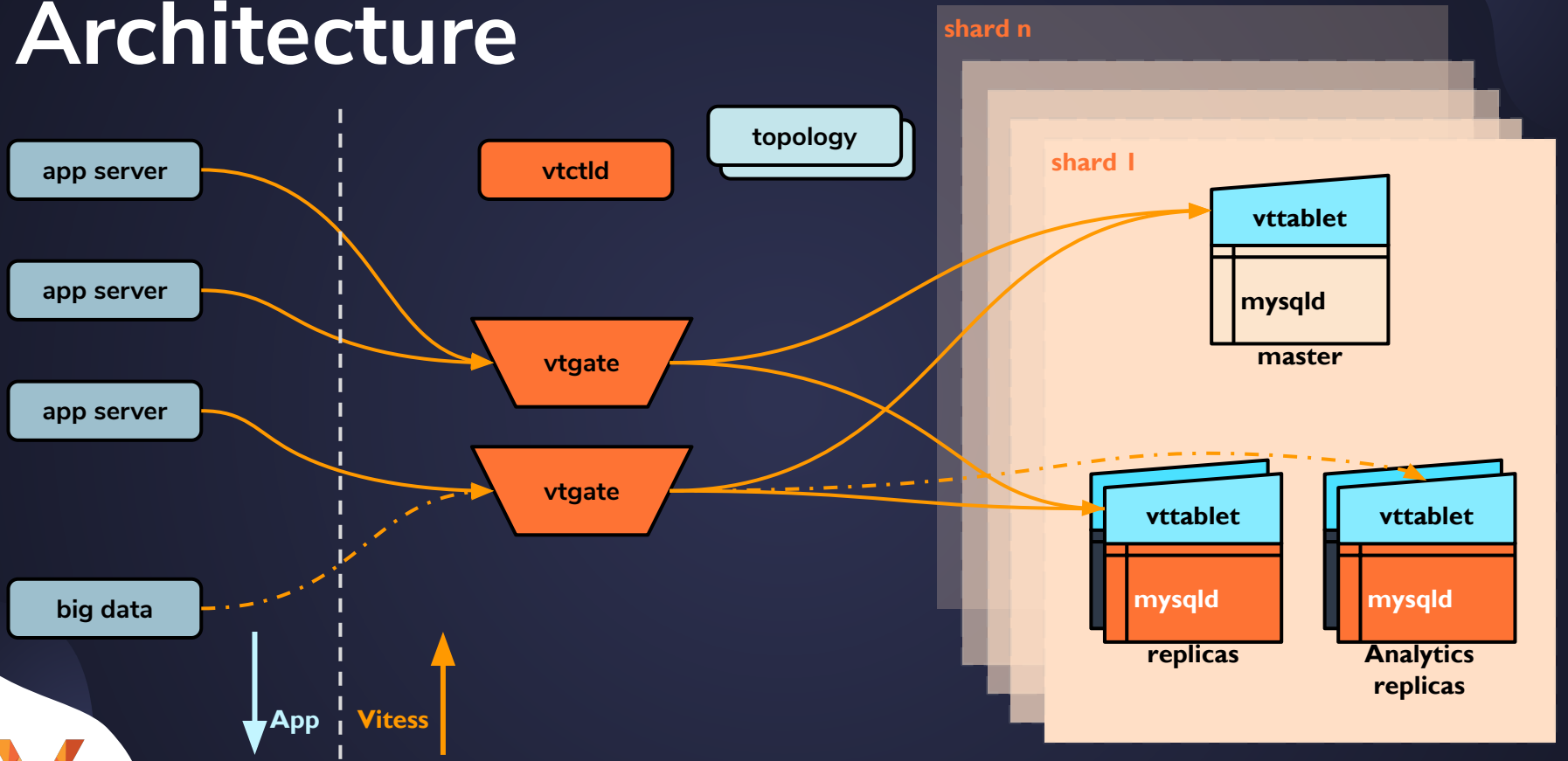
1000+
Forks

Vitess

Key Adopters

Vitess

Architecture



Vitess

VITESS.IO

Migration Plan (Stage 1)

Test Vitess for Query Compatibility

Test Application for added Latency

Backout if any issues discovered

Test Queries

- Best to start in development and QA
- Some applications have user-generated queries
- I still want to verify in production

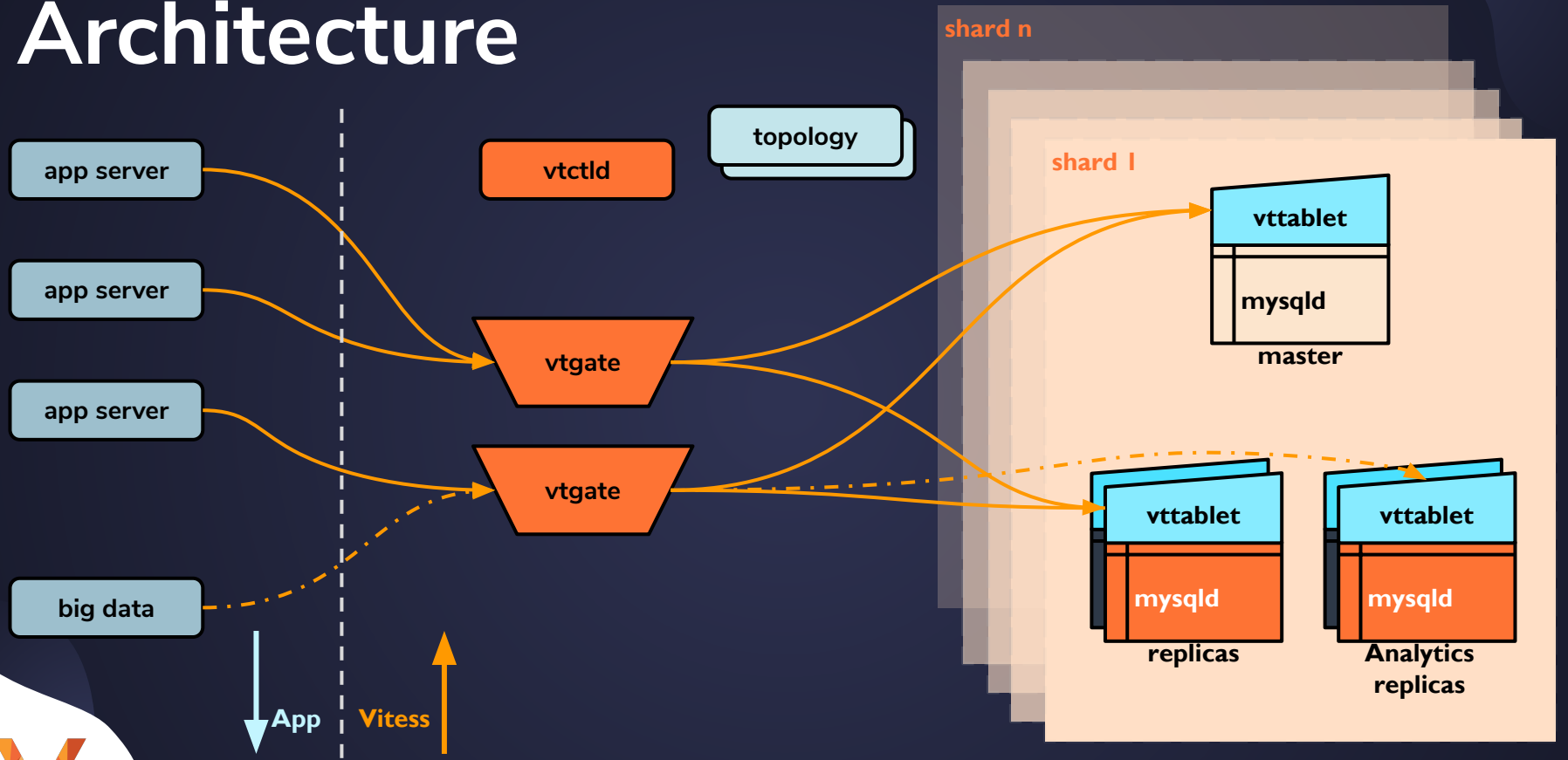
Added Latency

- Vitess requires one more network hop than MySQL (VTGate Proxy)
- Simple back of napkin math: +1-2ms on each query
- Should be within tolerable threshold for most Apps
    - Edge cases are N+1 pattern, typically not well designed apps.

Backout Plan

- We've verified in Dev/QA that our App works with Vitess
- No observable errors or problems
- Good ops practice is to use a Canary
  - Migrate just 5% of our traffic to Vitess
  - Rollback if any issues
  - Works great with Vitess!

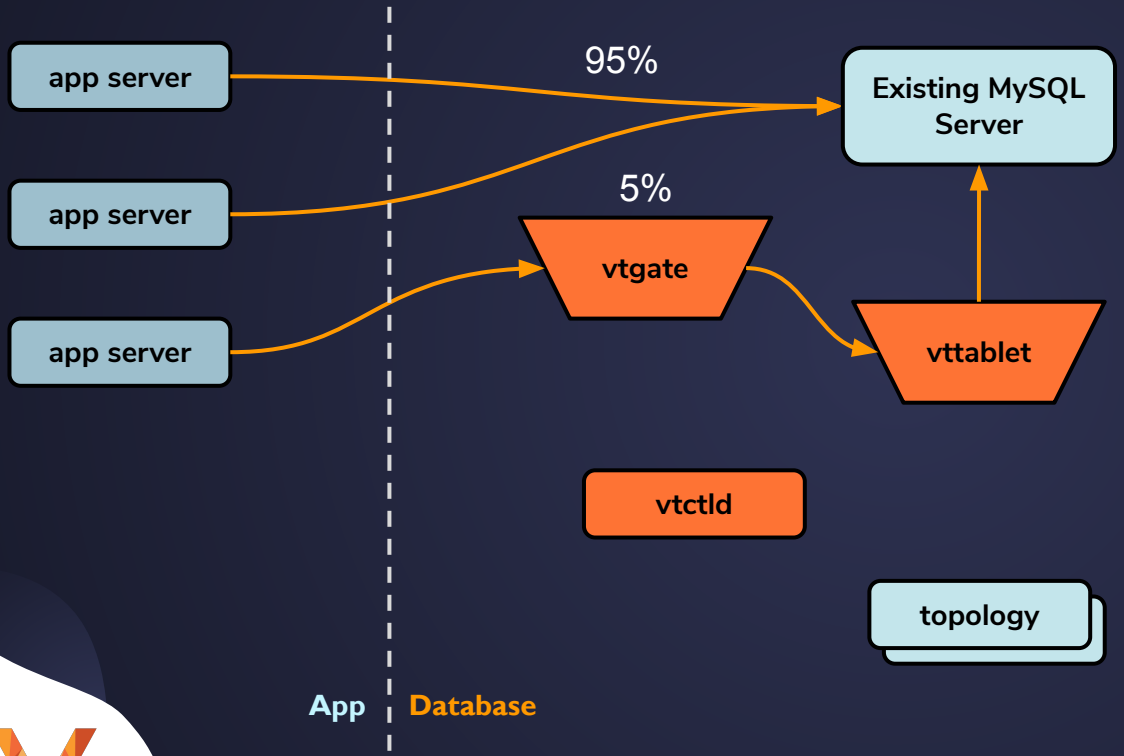
Architecture



Vitess

VITESS.IO

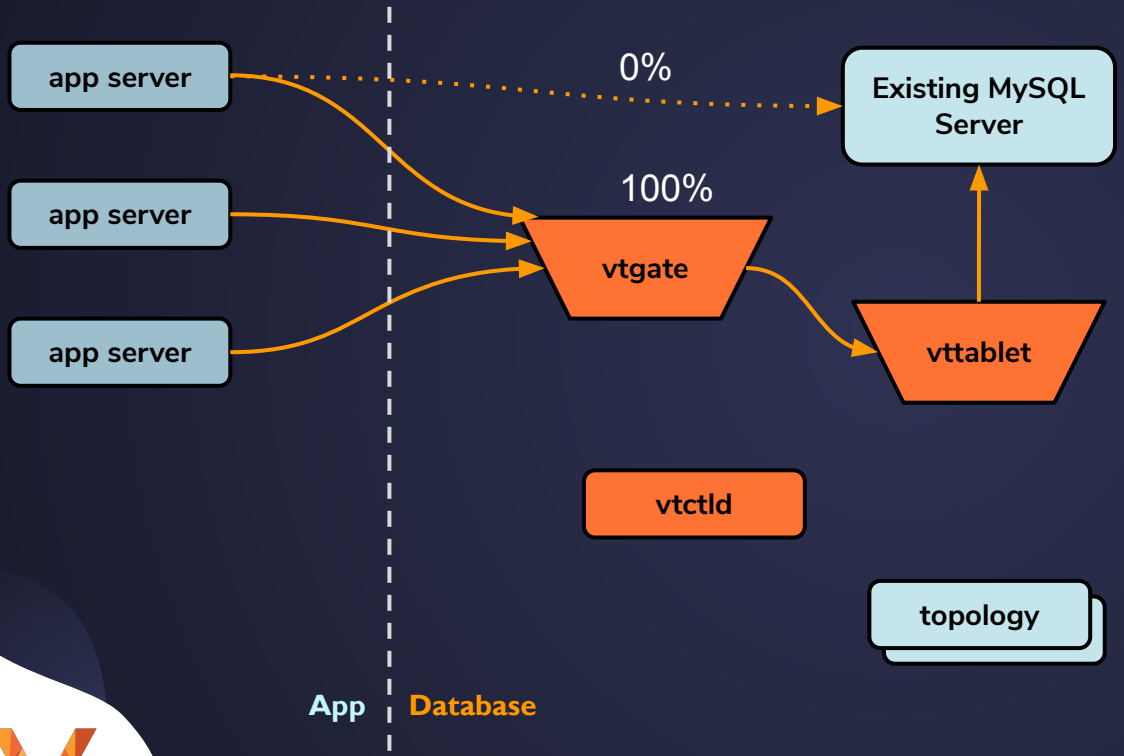
Canary Deployment



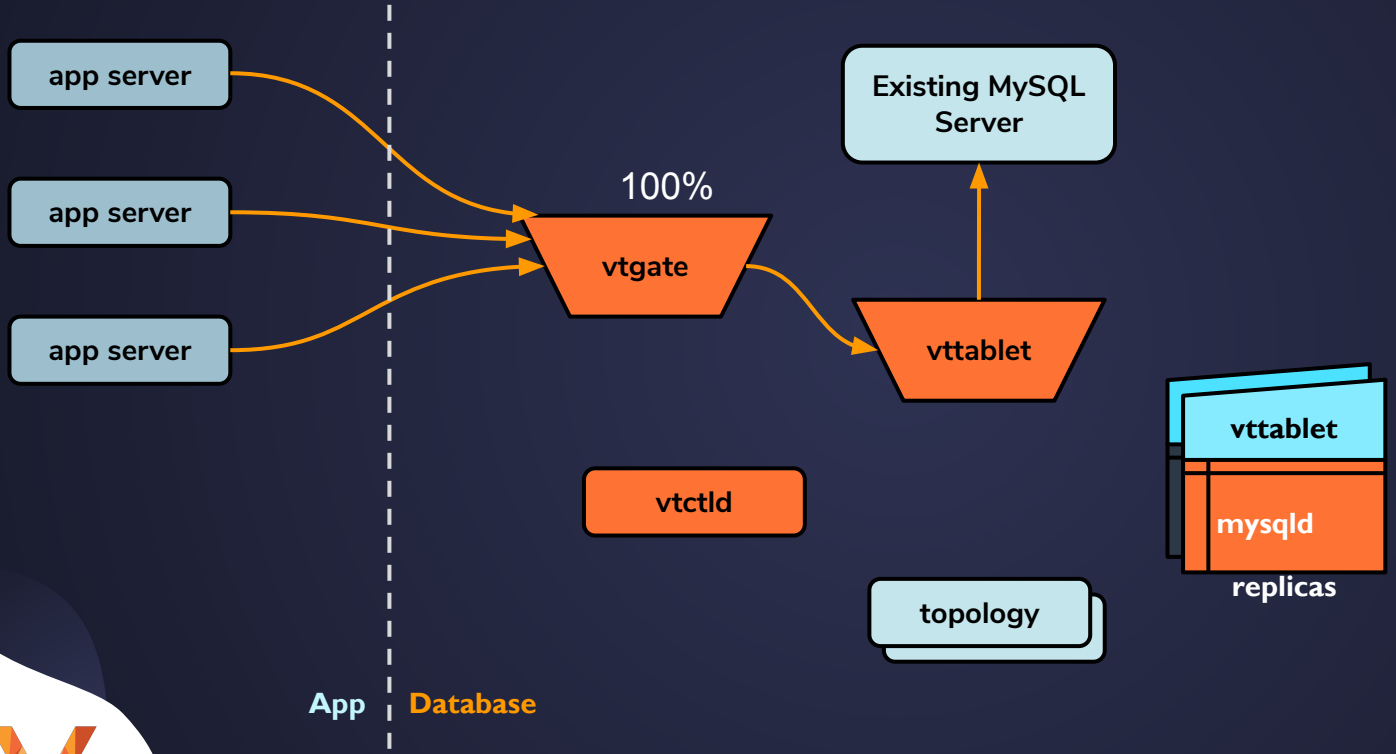
VTGate

- Is a stateless process
- Does not care if you route directly to MySQL
- Does not care if you route to another VTGate

Phase 2: Prerequisite

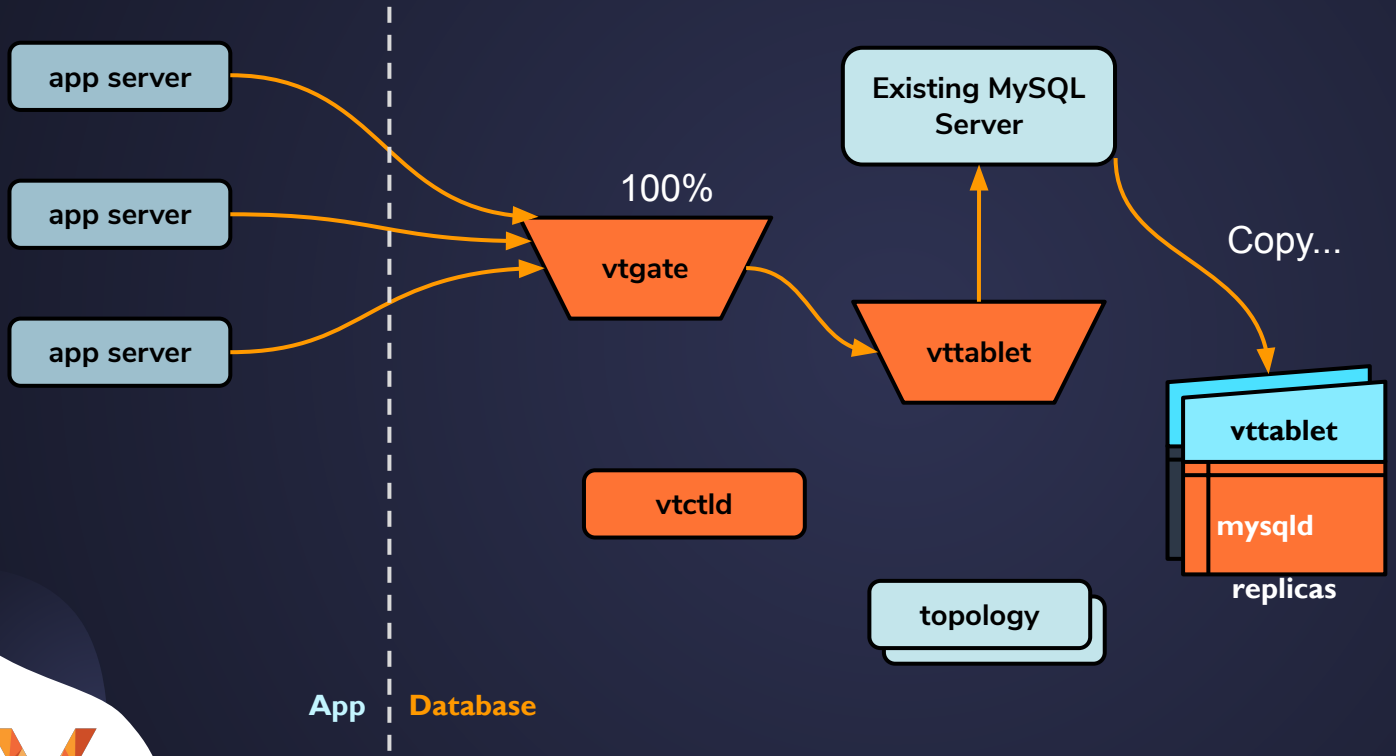


Phase 2: Add a New Tablet



Vitess

Phase 2: Table Migration

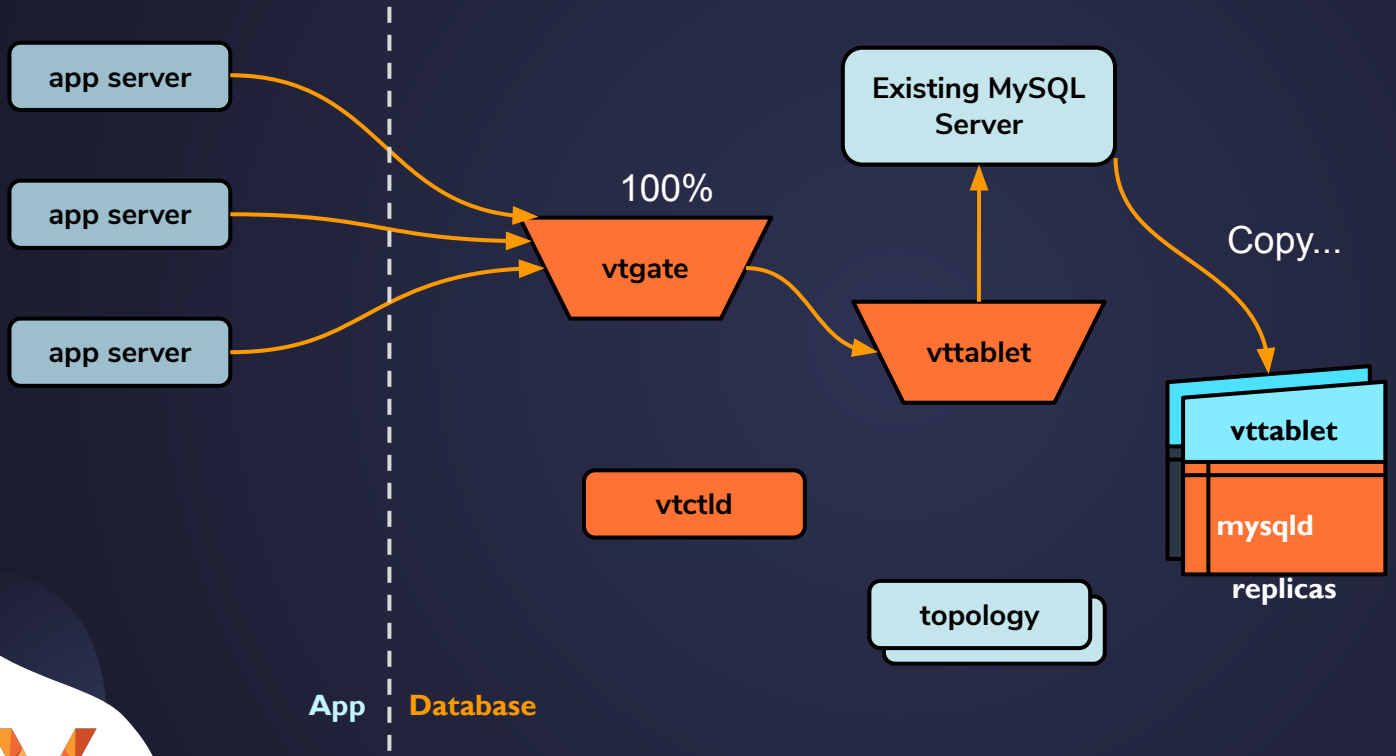


Vitess

Table Migration

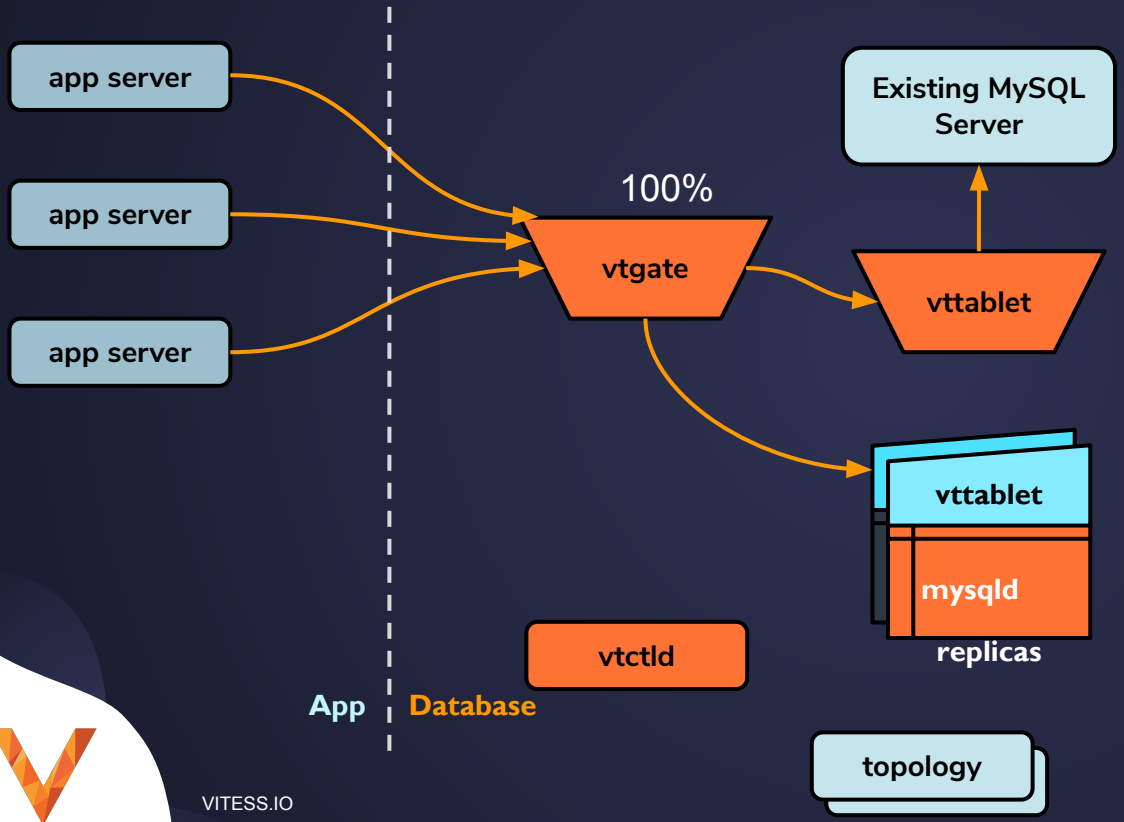
- Also called “Vertical Split Clone”
- Requires your source MySQL Server to have enabled:
  - Binary Logging with GTIDs
  - Row-based Replication
  - Matching Character Set (utf8)
- Copy phase is completely online
- Final cut-over will take a couple of seconds of blocking

Phase 2: Completion



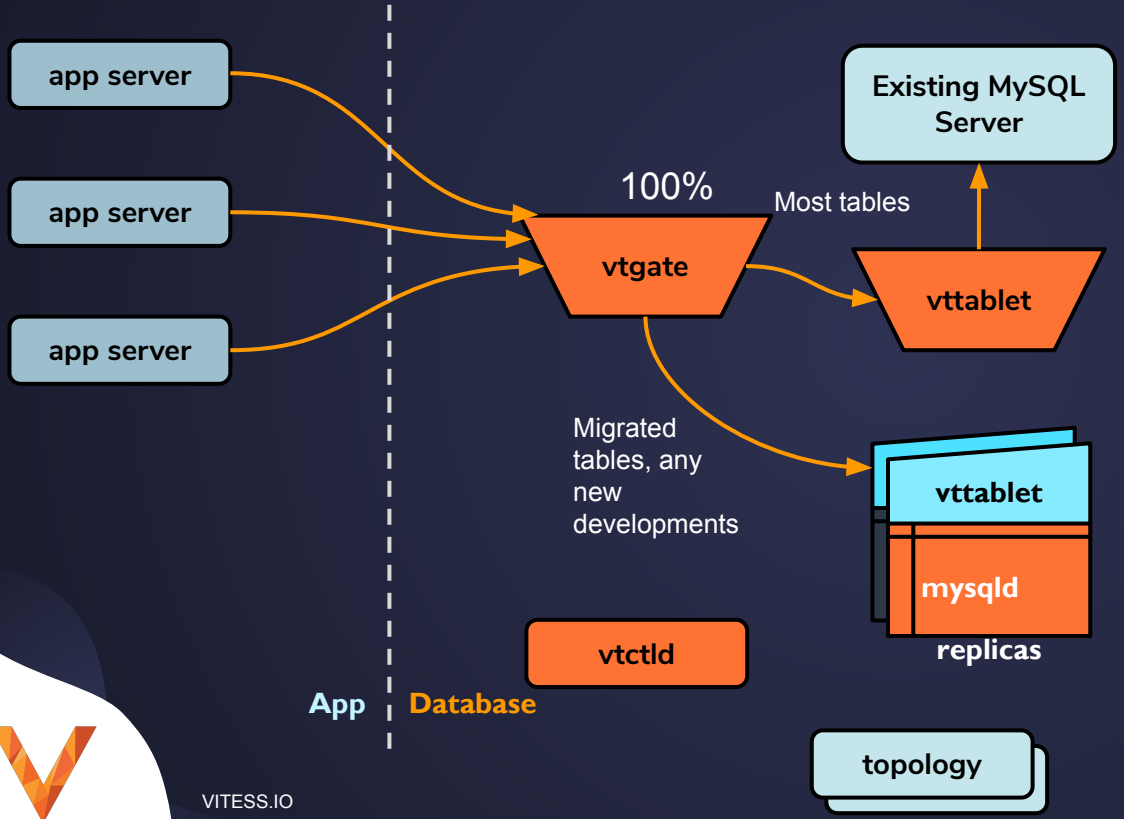
Vitess

Phase 2: Completion



Vitess

Phase 2: Completion



Vitess

Our Setup

- Legacy MySQL is one Keyspace
  - In Vitess terminology “The tablet server uses an externally managed MySQL”
- New Infrastructure is a single unsharded Keyspace
  - We could have just as easily migrated to a sharded keyspace
- We can still join between tablets in each keyspace
- It is recommended to keep updates contained within a single keyspace

Vitess

VITESS.IO

What's Next

Vitess.io

Do the tutorial

Bring up a cluster

slack
Join Vitess Slack

console.planetscale.com

Don't leave your data behind.

We’re hiring!
planetscale.com/careers

Other Vitess talks at KubeCon

Scaling Resilient Systems: A Journey into Slack's Database Service

Rafael Chacon & Guido Iaquinti, Slack

Tuesday 11:50am - 12:25pm (Ballroom Sec 20AB)

How to Migrate a MySQL Database to Vitess

Sugu Sougoumarane & Morgan Tocker, PlanetScale

Tuesday 11:50am - 12:25pm (6D)

Building a Database as a Service on Kubernetes

Abhi Vaidyanatha & Lucy Burns, PlanetScale

Tuesday 2:25 pm - 3:00pm (30ABCDE)

Vitess: Stateless Storage in the Cloud

Sugu Sougoumarane, PlanetScale

Tuesday 3:20 - 3:55pm (11AB)

Growth and Design Patterns in the Extensions Ecosystem

Eric Tune, Google

Wednesday 11:50am - 12:25pm (31ABC)

Geo-partitioning with Vitess

Deepthi Sigireddi & Jitendra Vaidya, PlanetScale

Wednesday 2:25 pm - 3:00pm (6D)

Gone in 60 Minutes: Migrating 20 TB from AKS to GKE in an Hour with Vitess

Derek Perkins, Nozzle