Using Rust to Build
a Distributed Transactional Key-Value Database

LiuTang | tl@pingcap.com
About me

- Chief Architect at PingCAP
- TiDB and TiKV
- Open source projects
  - LedisDB
  - go-mysql
  - go-mysql-elasticsearch
  - rust-prometheus
  - ...

Agenda

● Introduction
● Hierarchy
  ○ Storage
  ○ Raft
  ○ Transaction
  ○ RPC Framework
  ○ Monitor
  ○ Test
● Combine them all
When we want to build a distributed transactional key-value database...
Consistency

- Performance
- ACID
- HA

- Scalability
- Stability
- Others…
A High Building,
A Low Foundation
Language
Let’s start from scratch!!!
RocksDB

 immutable memory table

 Memory Table

 WAL

 Memory Disk

 Flush

 Level 0

 SST

 Level 1

 SST

 SST

 ...... SST

 Level 2

 SST

 SST

 ...... SST

 Info Log

 Manifest

 Current
https://github.com/pingcap/rust-rocksdb
Raft
Multi-Raft
Multi-Raft - Scalability
Multi-Raft - Scalability

Raft ConfChange - AddNode
Multi-Raft - Scalability

Raft ConfChange - RemoveNode
https://github.com/pingcap/raft-rs
Transaction
Transaction

```rust
let mut txn = store.begin()
let value1 = txn.get(region1_key)
let value2 = txn.get(region2_key)
// do something with value
txn.set(region1_key, new_value1)
txn.set(region2_key, new_value2)
txn.commit()
// or txn.rollback()
```

How to keep consistency crossing multi-Raft Groups?
Transaction

1. Inspired by Google Percolator
2. Optimized Two Phase Commit (2 PC)
3. Multiversion Concurrency Control (MVCC)
4. Snapshot Isolation
5. Optimistic Transaction
gRPC

- Mode
  - Unary
  - Client streaming
  - Server streaming
  - Duplex streaming

- Using Futures to wrap the asynchronous C gRPC API

```javascript
let f = unary(service, method, request);
let resp = f.wait();
```
https://github.com/pingcap/grpc-rs
Prometheus

- Type
  - Counter
  - Gauge
  - Histogram

```rust
lazy_static! {
    static ref HTTP_COUNTER: Counter = register_counter!(
        "http_request_total",
        "Total number of HTTP request."
    ).unwrap();
}

HTTP_COUNTER.inc();
```
Testing
Testing - Failure Injection

// Ingest a failure
fn function_foo() {
    fail_point!("foo");
}

// Run and Trigger the failure
FAILPOINTS=foo=panic cargo run
https://github.com/pingcap/fail-rs
Architecture
Architecture

- Prometheus
- Client
- gRPC
- Txn API
- MVCC
- Raft
- RocksDB
https://github.com/pingcap/tikv
Beyond TiKV
A Distributed Relational Database
TiDB Applications

MySQL Drivers (e.g. JDBC)

MySQL Protocol

RPC

TiKV
A Distributed Analytical Database
TiSpark

Spark Cluster

Spark Driver

Worker
Worker
Worker

RPC

TiKV
Hybrid Transactional/Analytical Processing Database
Thank you!

https://github.com/pingcap/tidb
https://github.com/pingcap/tikv

We are hiring…
@China @Silicon Valley @Home