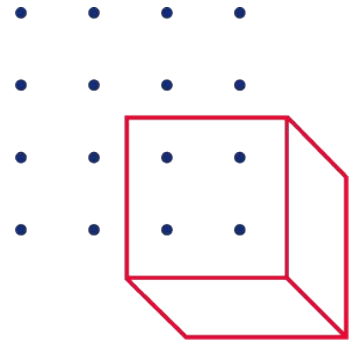
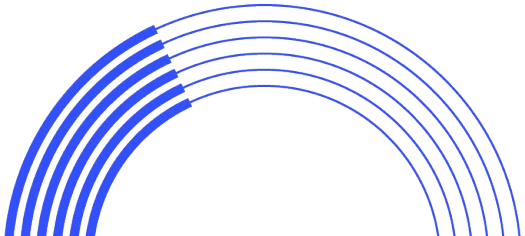
PingCAP



Distributed Database

2021/06/17

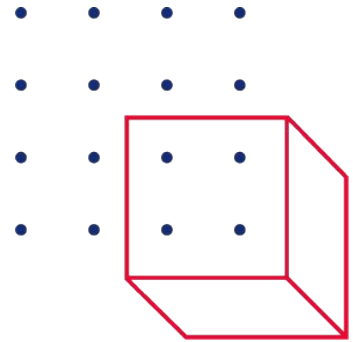


PingCAP

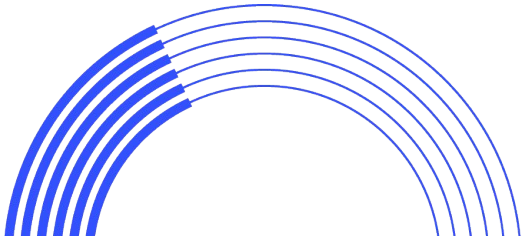
Agenda

1. The Evolution of Distributed Database
2. TiDB Architecture
3. How to Participate in This Area

PingCAP



The Evolution of Distributed Database



Overview

PingCAP

1. Sharding Middleware
2. NoSQL
3. NewSQL/HTAP
4. Cloud Native

Sharding Middleware

PingCAP

To solve the capacity bottleneck in single node:

- combine multi database nodes together into a logical database
- route queries to the correct node

Common problems:

- application developers need design the schema and queries carefully, resharding is painful
- lacking some features, for example, cross node join (depends on the implementation)

Typical systems: Vitess, MyCat, DRDS, ShardingSphere, etc.

NoSQL

PingCAP

To solve the scalability problem and achieve high agailability:

- no-relational data model
- low latency, high throughput, eventually consistancy

Common problems:

- No SQL API, not ideal for complex applications
- No strong transactional semantics, not ideal for some mission-critical applications

Typical systems: BigTable, HBase, Cassandra, MongoDB, DynamoDB

NewSQL

PingCAP

To provide scalability, flexibility, high availability, ACID transaction, and SQL support:

- Deliver the scalability and flexibility promised by NoSQL systems
- Providing ACID transaction and SQL support

Common problems:

- Transaction latency is High without fine-grained optimization

Typical systems: Spanner, F1, CockroachDB, TiDB

Some of them are working on HTAP functionality

Cloud Native

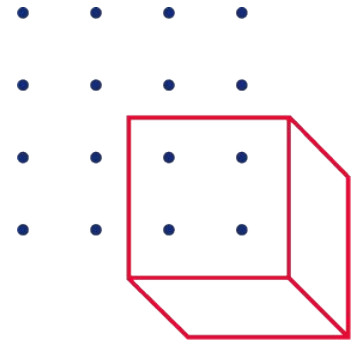
PingCAP

To provide scalability, flexibility, high availability, ACID transaction, and SQL support:

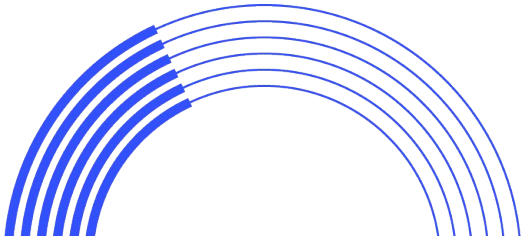
- Building database components based on cloud services
- Scale storage and compute independently

Typical systems: Snowflake, Aurora, Cloud Spanner

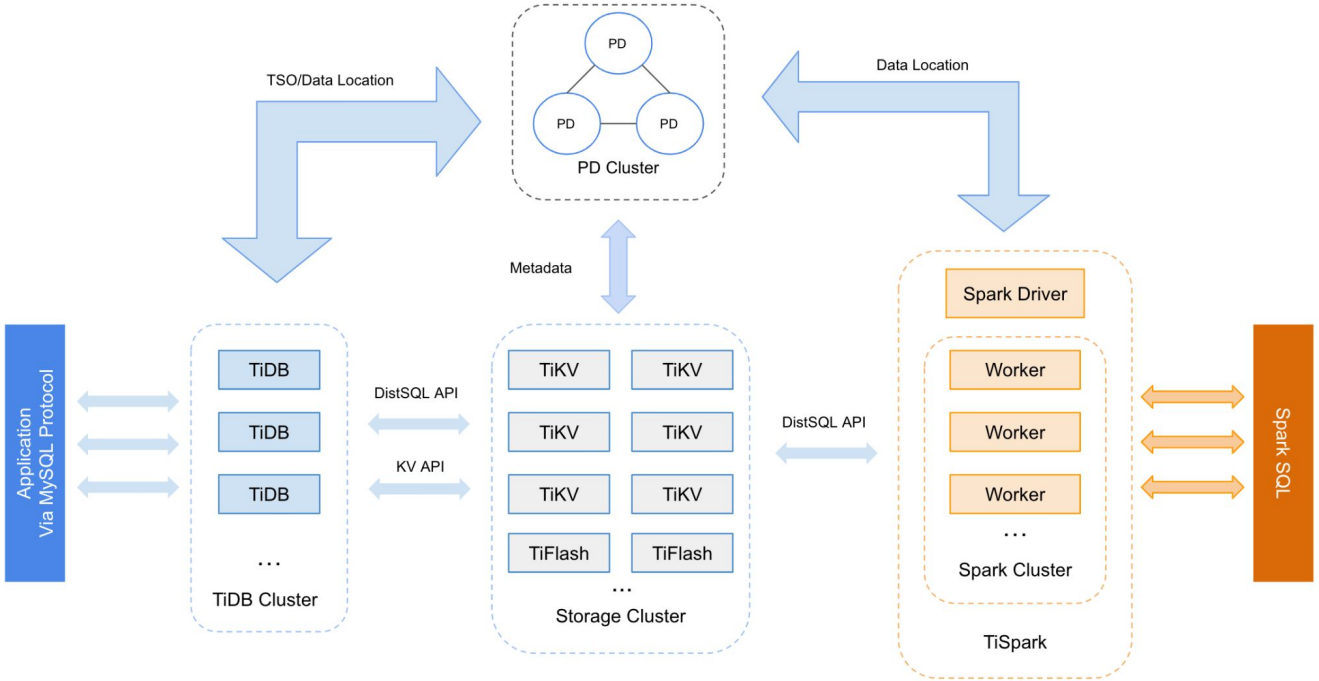
PingCAP



TiDB Architecture



Overview

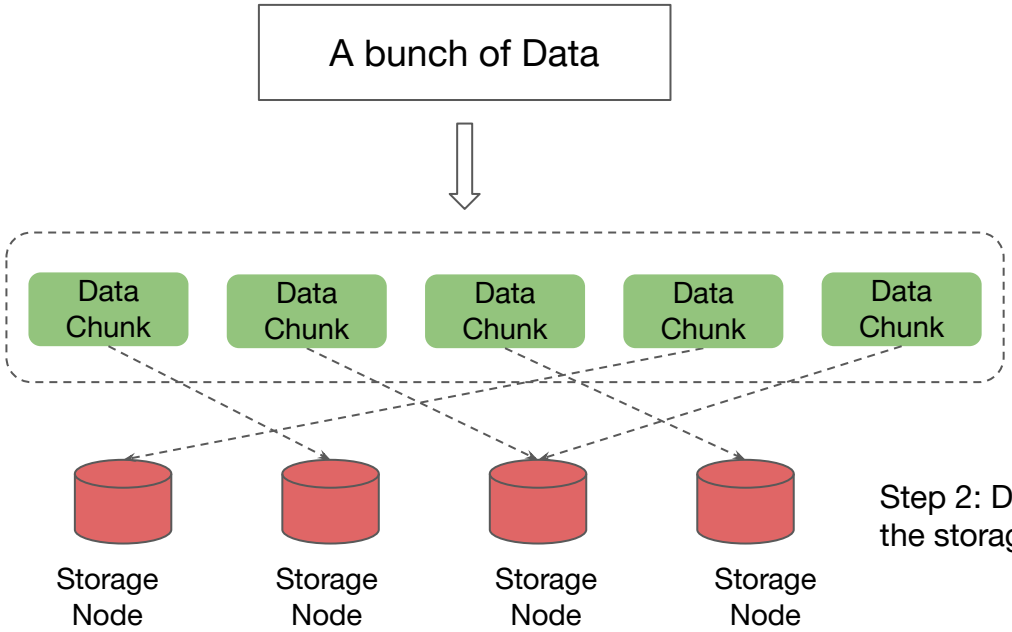

PD
PD
PD
PD Cluster
TSO/Data Location
Data Location
Metadata
Application
Via MySQL Protocol
TiDB
TiDB
TiDB
...
TiDB Cluster
DistSQL API
KV API
TiKV
TiKV
TiKV
TiKV
TiKV
TiKV
TiFlash
TiFlash
...
Storage Cluster
DistSQL API
Spark Driver
Worker
Worker
Worker
...
Spark Cluster
TiSpark
Spark SQL

TiDB Design Goals

PingCAP

- **Horizontal Scalability**
  - Scale-out process should be transparent to the application layer
- **High Availability**
  - Self-healing thanks to modern consensus algorithm like Raft/Multi-Paxos
- **ACID** Compliance
  - Distributed cross-row transaction support
- **MySQL** Protocol && Dialect
  - MySQL syntax & ecosystem tools

To achieve Horizontal Scalability



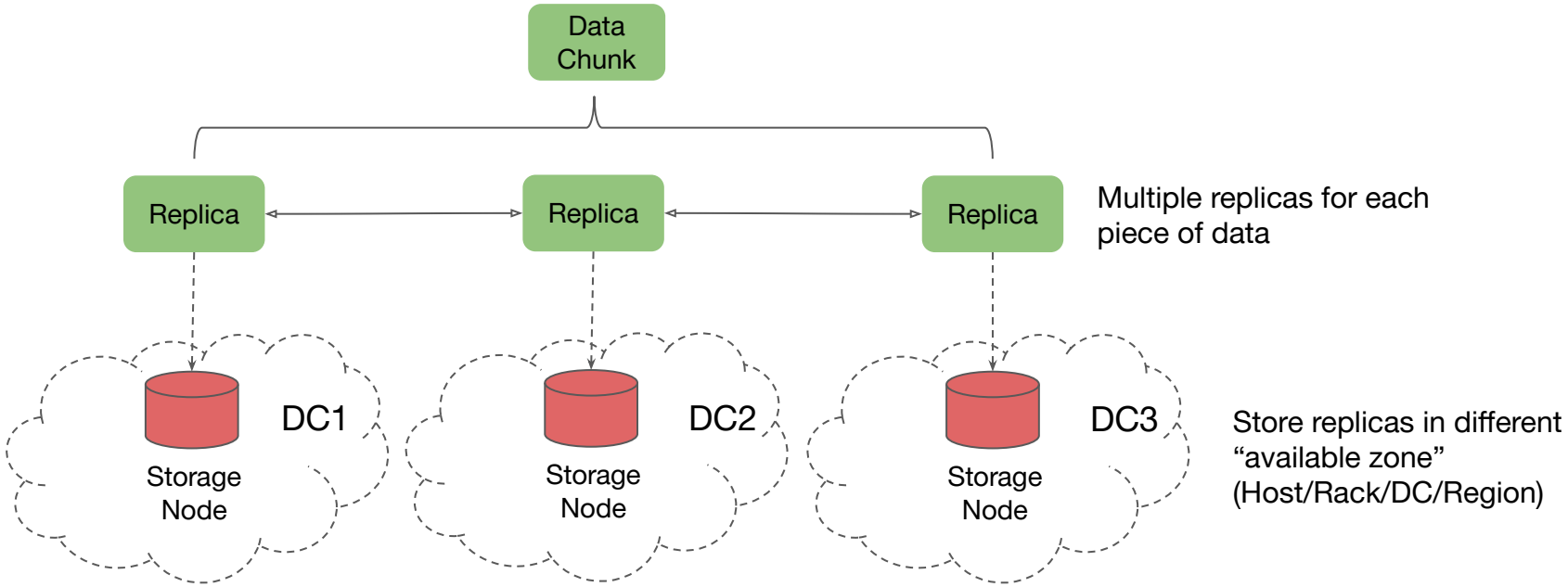
Step 1: Shard/Split data into smaller chunks

Step 2: Distribute the chunks among the storage nodes

How to decide chunk size?

How to do balance load?

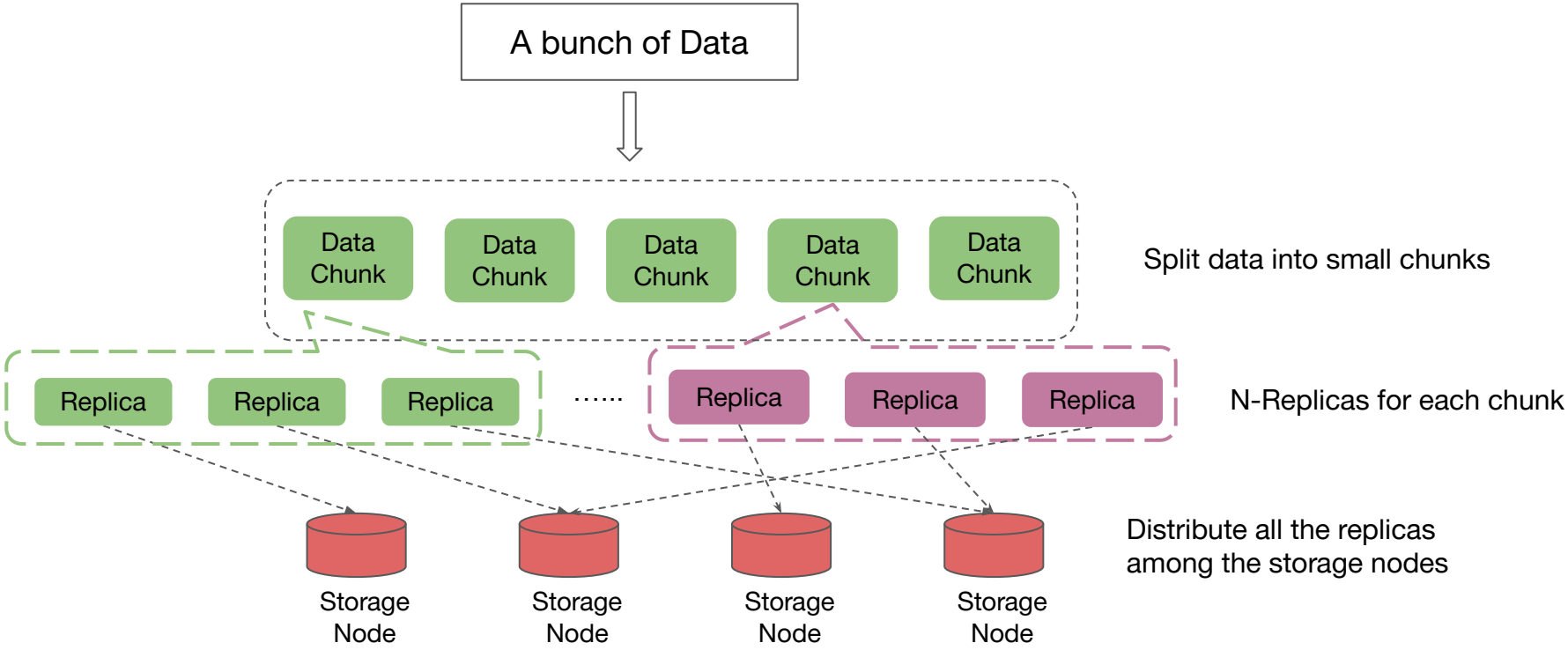
To achieve High Availability



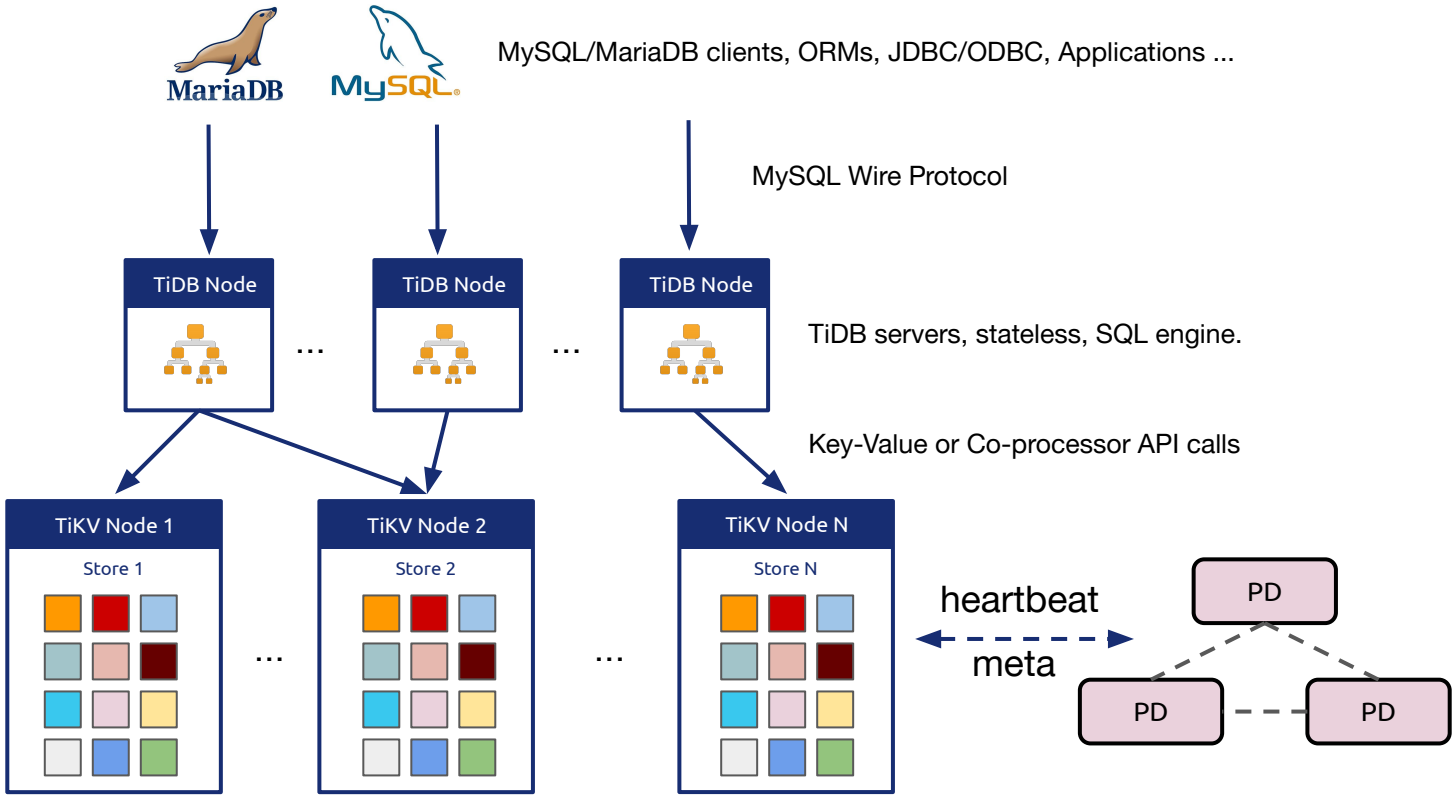
How to make data consistent among replicas?

What will happen if one replica is gone?

Combine Scalability and HA together



TiDB Architecture



Query Execution

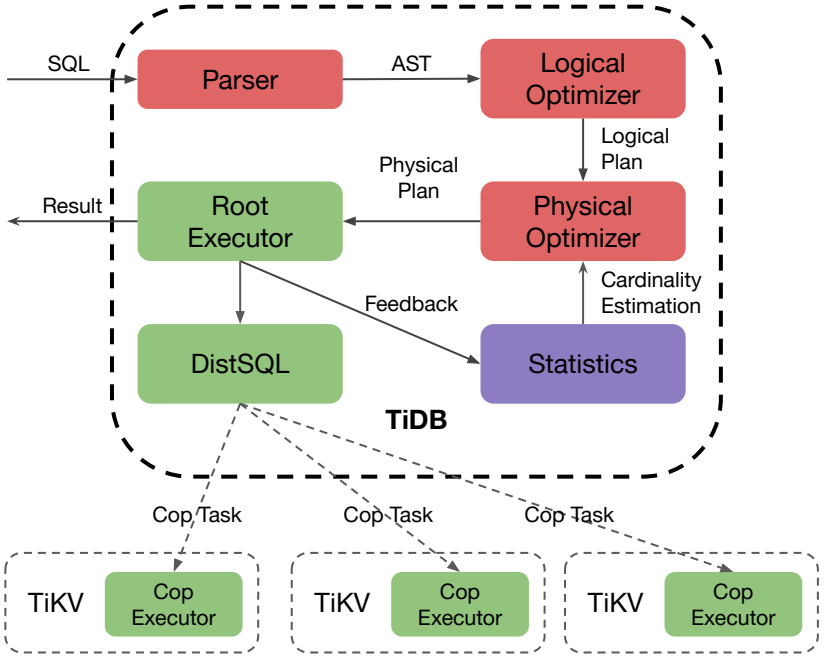
PingCAP

Query Optimization

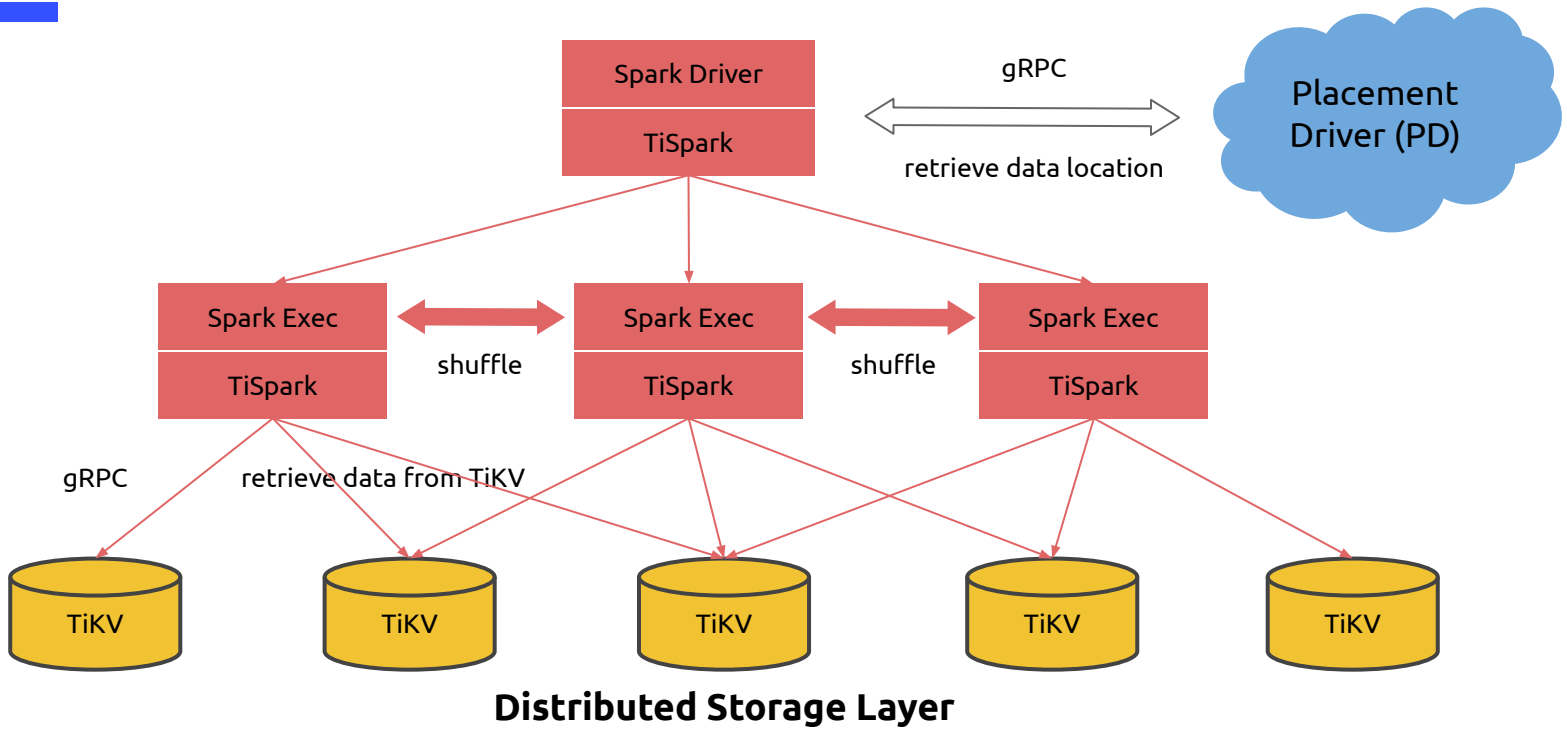
- Logical Optimizer
- Physical Optimizer
- Statistics

Query Execution

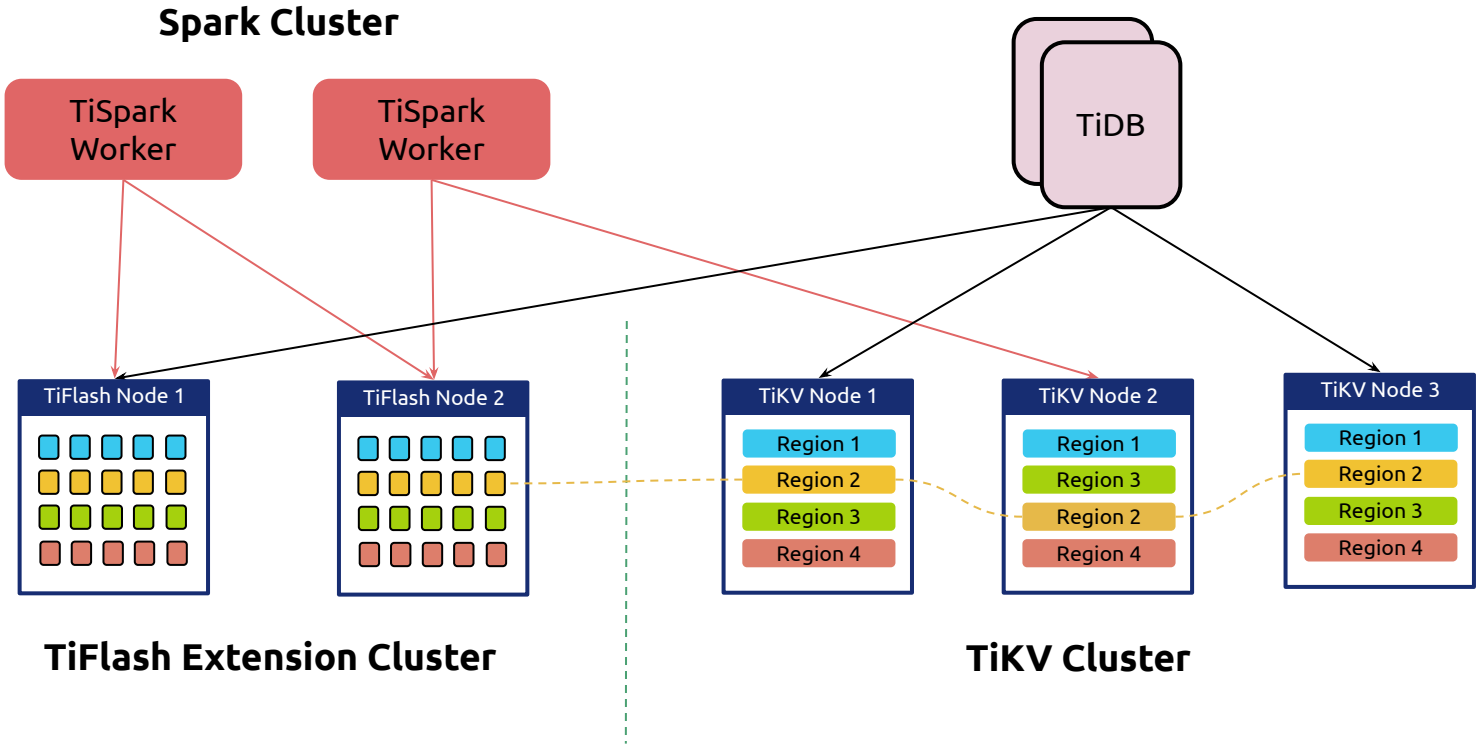
- Root Executor
- DistSQL
- Coprocessor Executor



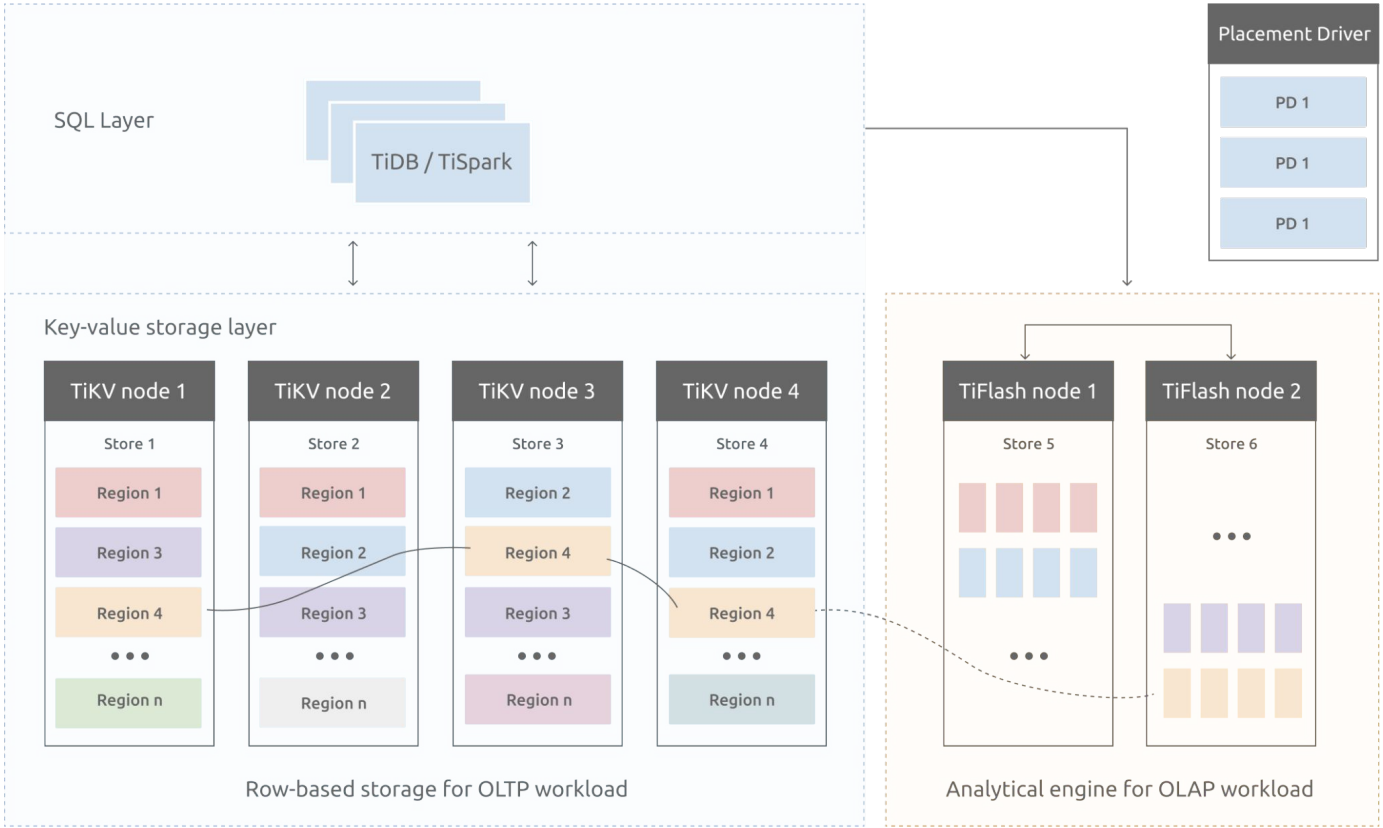
TiSpark


Spark Driver
TiSpark
gRPC
retrieve data location
Placement Driver (PD)
Spark Exec
TiSpark
shuffle
Spark Exec
TiSpark
shuffle
Spark Exec
TiSpark
gRPC
retrieve data from TiKV
TiKV
TiKV
TiKV
TiKV
TiKV
Distributed Storage Layer

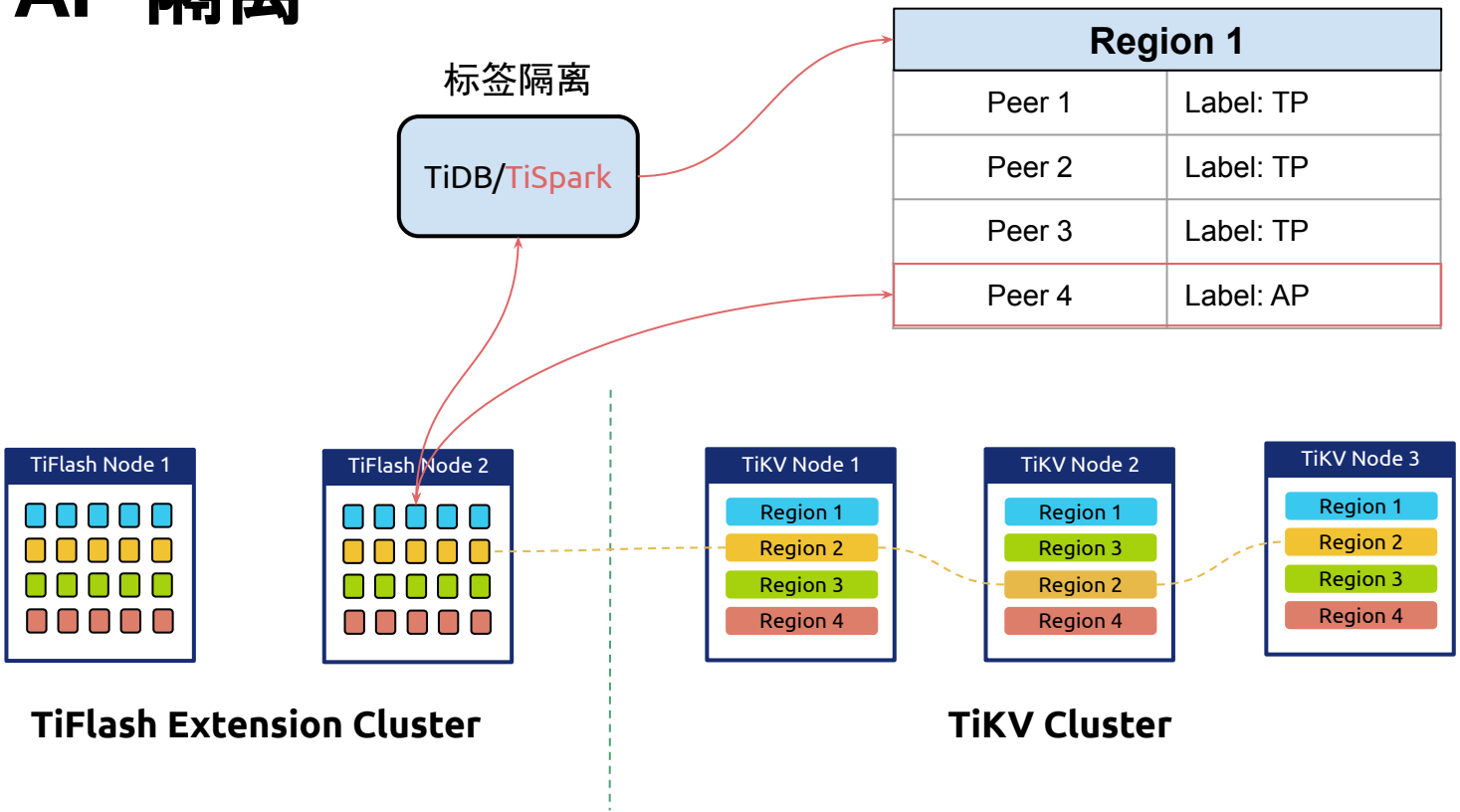
TiFlash 4.0


Spark Cluster
TiSpark Worker
TiSpark Worker
TiDB
TiFlash Node 1
TiFlash Node 2
TiKV Node 1
Region 1
Region 2
Region 3
Region 4
TiKV Node 2
Region 1
Region 3
Region 2
Region 4
TiKV Node 3
Region 1
Region 2
Region 3
Region 4
TiFlash Extension Cluster
TiKV Cluster

TiFlash 5.0 - MPP Support

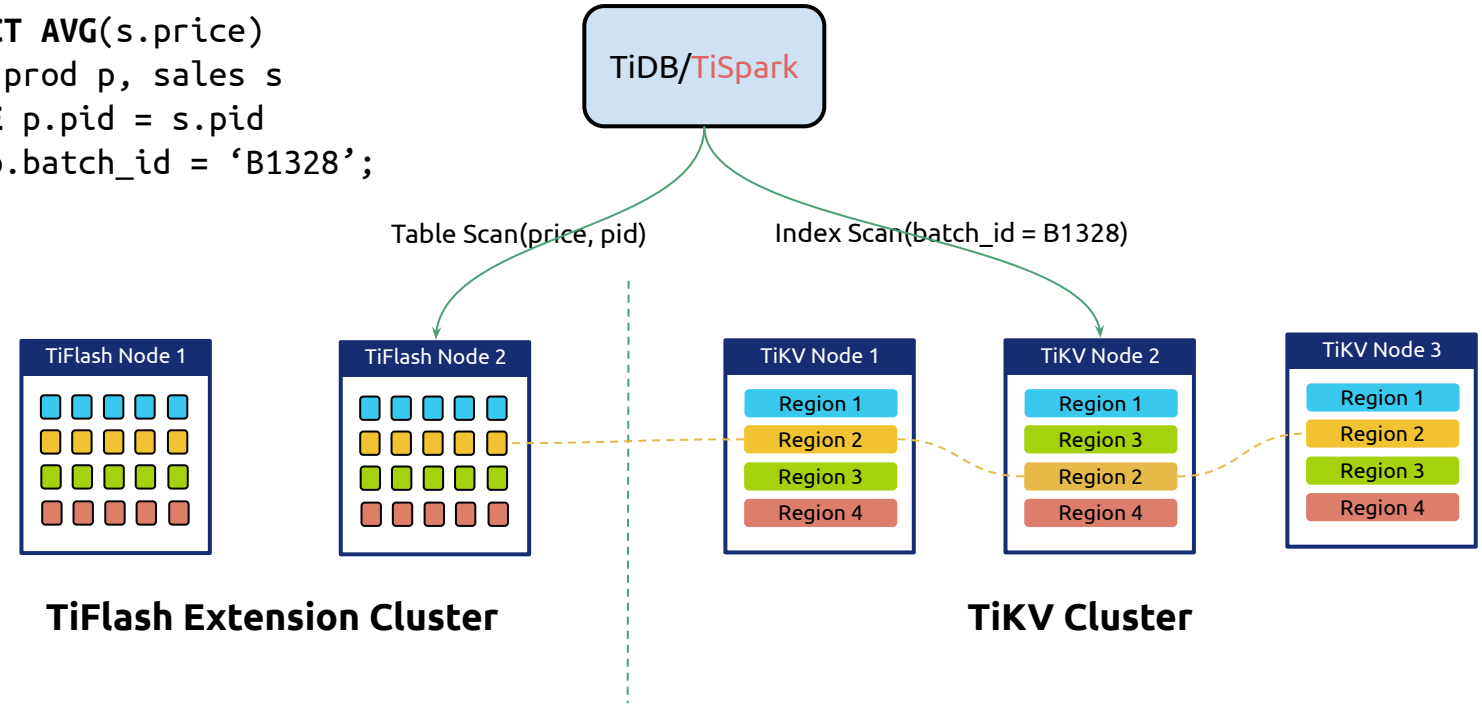


TP、AP 隔离

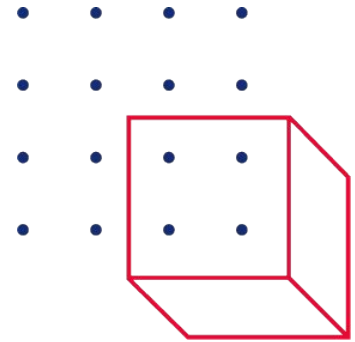

标签隔离
TiDB/TiSpark
Region 1
Peer 1
Label: TP
Peer 2
Label: TP
Peer 3
Label: TP
Peer 4
Label: AP
TiFlash Node 1
TiFlash Node 2
TiKV Node 1
Region 1
Region 2
Region 3
Region 4
TiKV Node 2
Region 1
Region 3
Region 2
Region 4
TiKV Node 3
Region 1
Region 2
Region 3
Region 4
TiFlash Extension Cluster
TiKV Cluster

TP、AP 融合

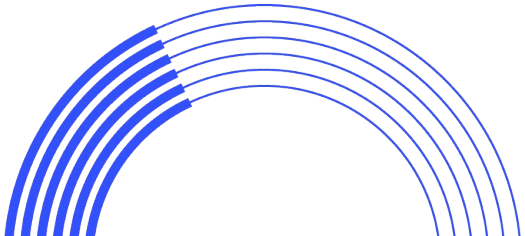
```
SELECT AVG(s.price)
FROM prod p, sales s
WHERE p.pid = s.pid
AND p.batch_id = 'B1328';
```



PingCAP



How to Participate in This Area



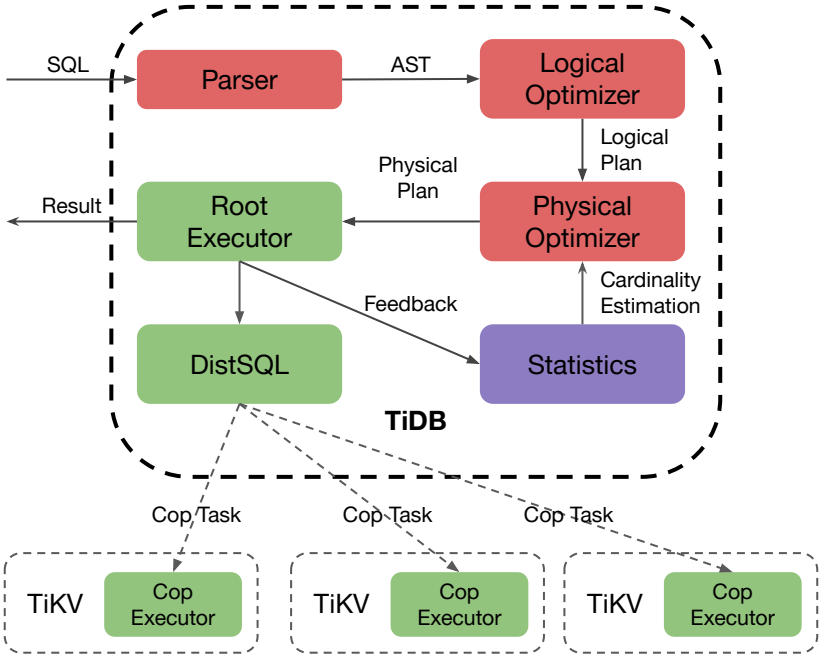
Suggestions

从一个小模块开始，成为细分领域的专家

逐渐扩展多个小模块，成为它们的细分领域专家

PingCAP

Suggestions



Suggestions