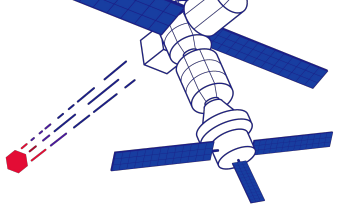
TiDB

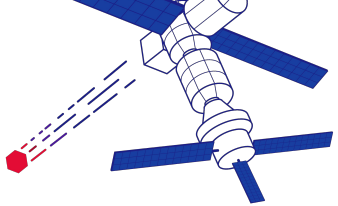
TiDB HTAP 介绍

HTAP 介绍



- HTAP = Hybrid Transactional / Analytical Processing
  - 由 Gartner 提出
  - 混合交易 / 分析型处理
  - 这个词汇用来描述同时具备以上两种处理能力的数据库

为何需要 HTAP



- TP 和 AP 分别有完全不同的技术设计理念
- 想一下，如果你需要分析你的在线订单数据
  - 需要搭建 TP 和 AP 两套平台，分别处理订单交易以及交易数据分析
  - 通过 ETL 作业将数据从 TP 数据库搬运到 AP 数据库
  - 搬运本身非常耗时且难于维护
    - 当数据量超过单机时，技术复杂度会陡然上升

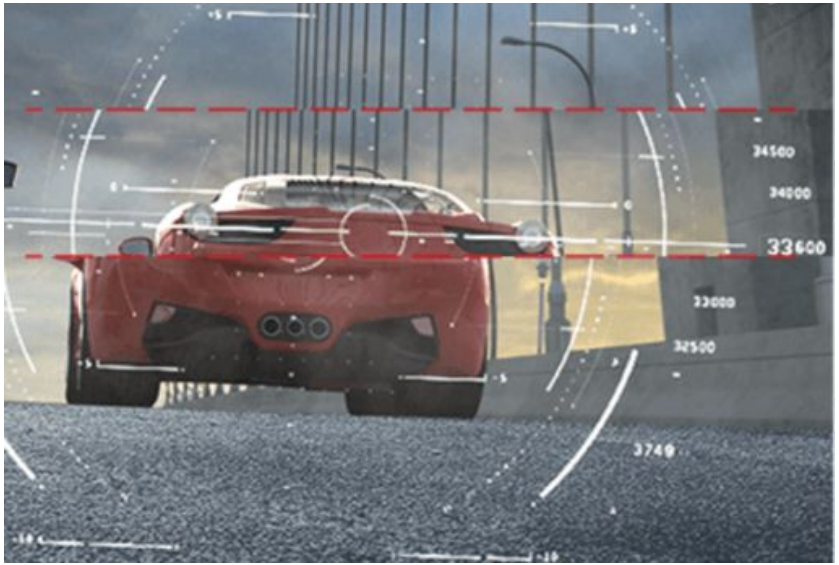
复杂性

OR

新鲜度

OR

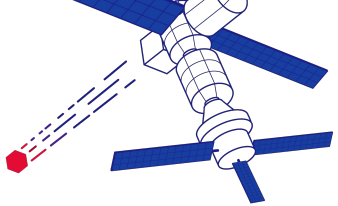
一致性


34560
34200
33600
33000
32500
3749

OR

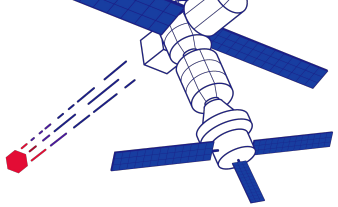

34560
34200
33600
33000
32500
3749

交易型数据库为何不擅长分析？



- 行存批量读取慢
  - 列存是分析场景的标配，配合向量化引擎如虎添翼
  - 行存在查询很少数据时对比列存有绝对优势
  - 但大量扫表且并非选中全部列时则 IO 效率极低

交易型数据库为何不擅长分析？



行存

id	name	age
0962	Jane	30
7658	John	45
3589	Jim	20
5523	Susan	52

SELECT * from emp where id = 7658;

SELECT avg(age) from emp;

列存

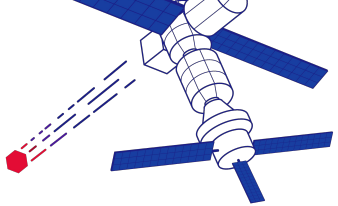
id
0962
7658
3589
5523

name
Jane
John
Jim
Susan

age
30
45
20
52

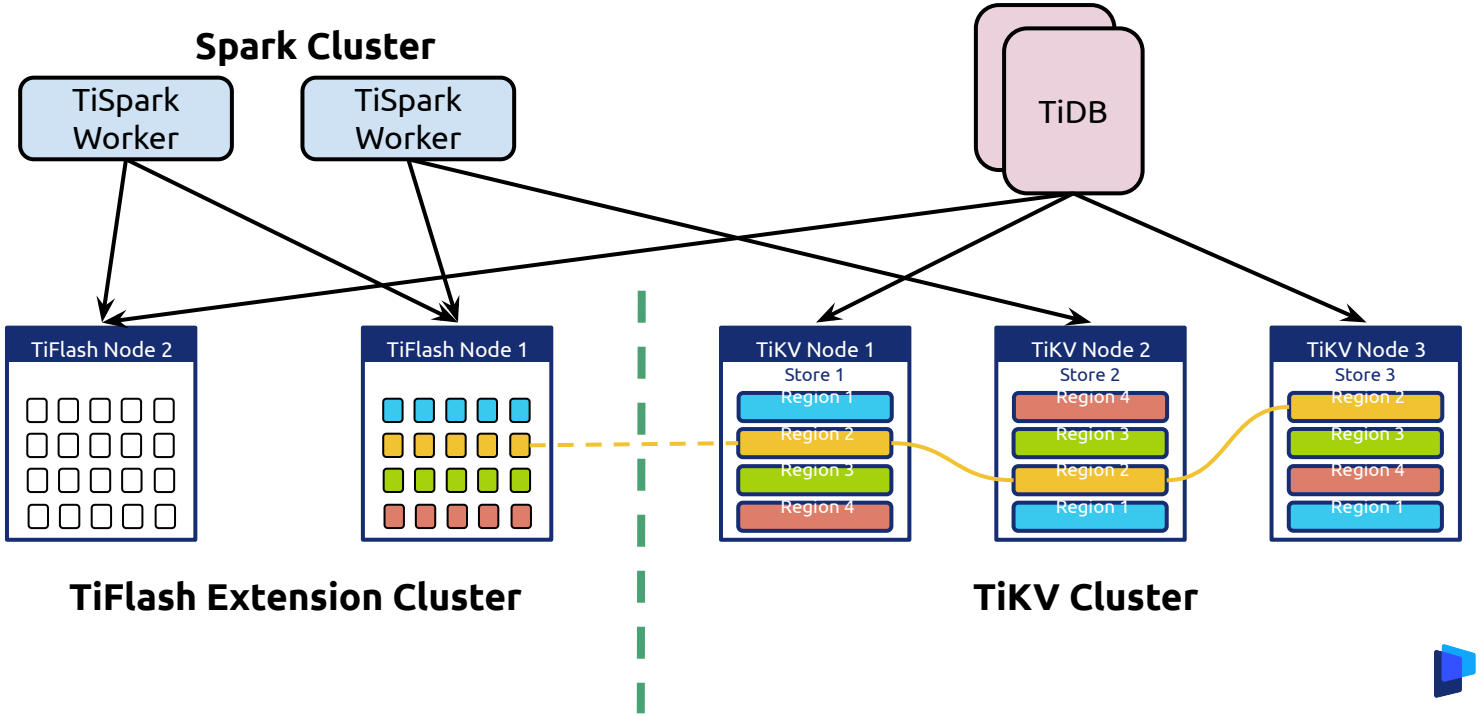
TP / AP 干扰

交易型数据库为何不擅长分析？



- TP 和 AP 互相影响
  - 分析场景使用资源的方式倾向于同时投入尽可能多的资源以迅速完成计算
  - 分析场景存储方式也倾向于暴力扫描而非精确索引
  - 很多用户在进行 AP 查询时甚至需要很小心调整并发和读取速度，防止干扰，就算引擎可以算得快，也戴上了枷锁

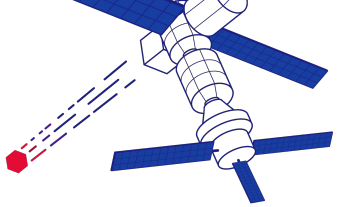
TiDB HTAP 架构

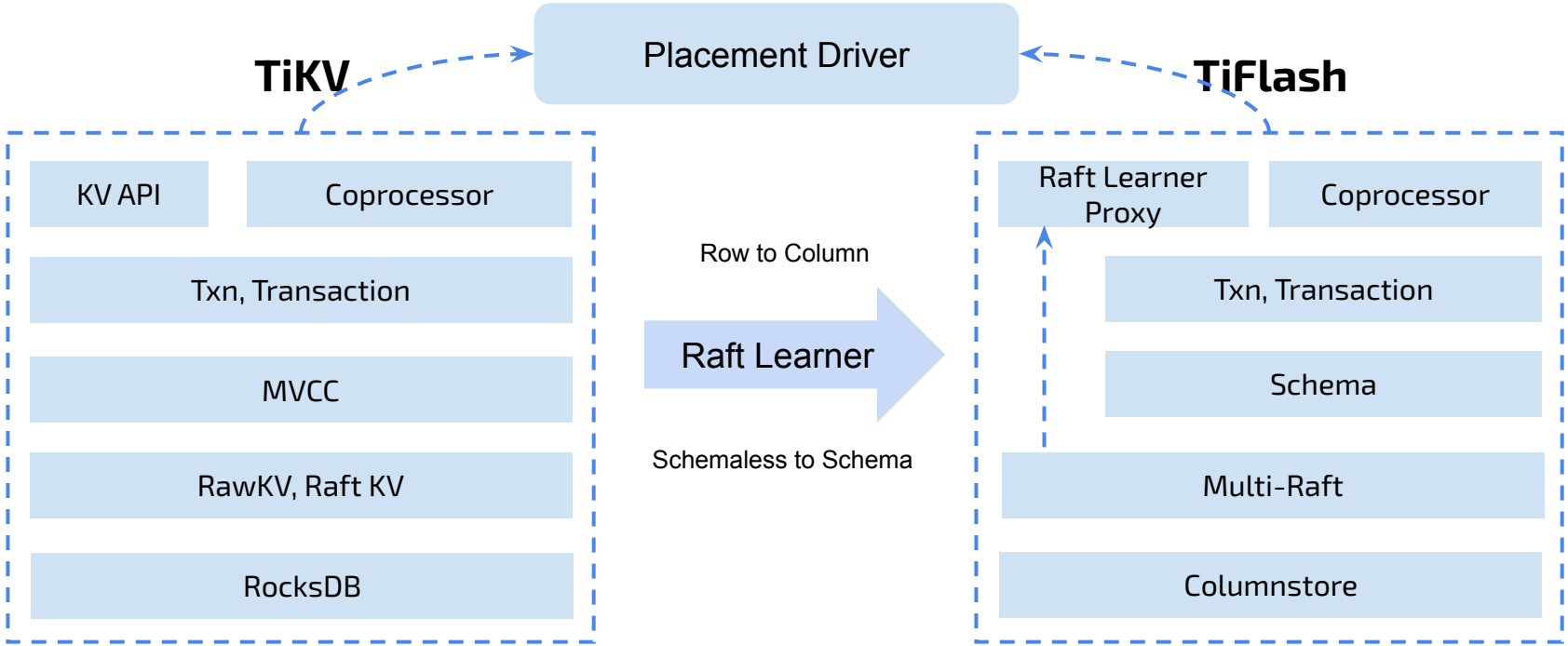

Spark Cluster
TiSpark Worker
TiSpark Worker
TiDB
TiFlash Node 2
TiFlash Node 1
TiKV Node 1
Store 1
Region 1
Region 2
Region 3
Region 4
TiKV Node 2
Store 2
Region 4
Region 3
Region 2
Region 1
TiKV Node 3
Store 3
Region 2
Region 3
Region 4
Region 1
TiFlash Extension Cluster
TiKV Cluster

PingCAP

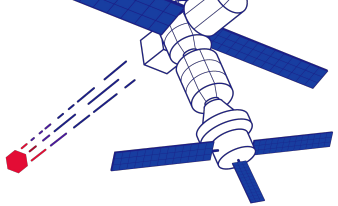
TiFlash

TiFlash 架构设计



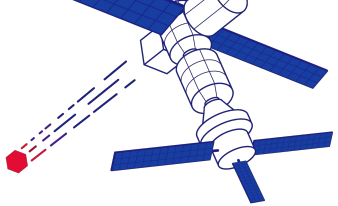


TiFlash 特色



- 高性能，支持实时更新的列存引擎
  - 高速实时基于主键的 Update 支持
- 以 Learner 角色接入 Multi-Raft 体系
  - 负载均衡，更快更方便 扩展
  - 强一致的读取
- 配合 TP 场景使用：TP 数据实时高速可查无干扰
- 配合 AP 场景使用：可自动大幅提升查询速度

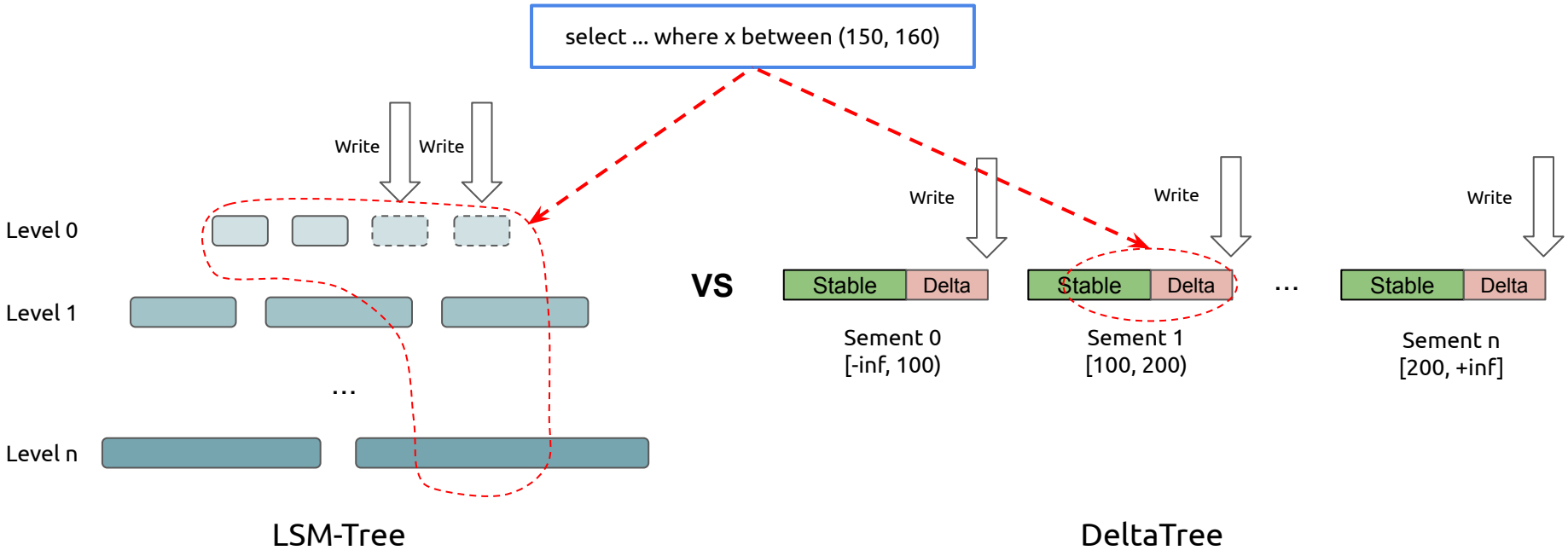
更快的查询响应



- 与 TiKV 不同的存储设计，更高效的批量读取性能
  - 分析场景特化的存储结构设计，更低的读取消耗

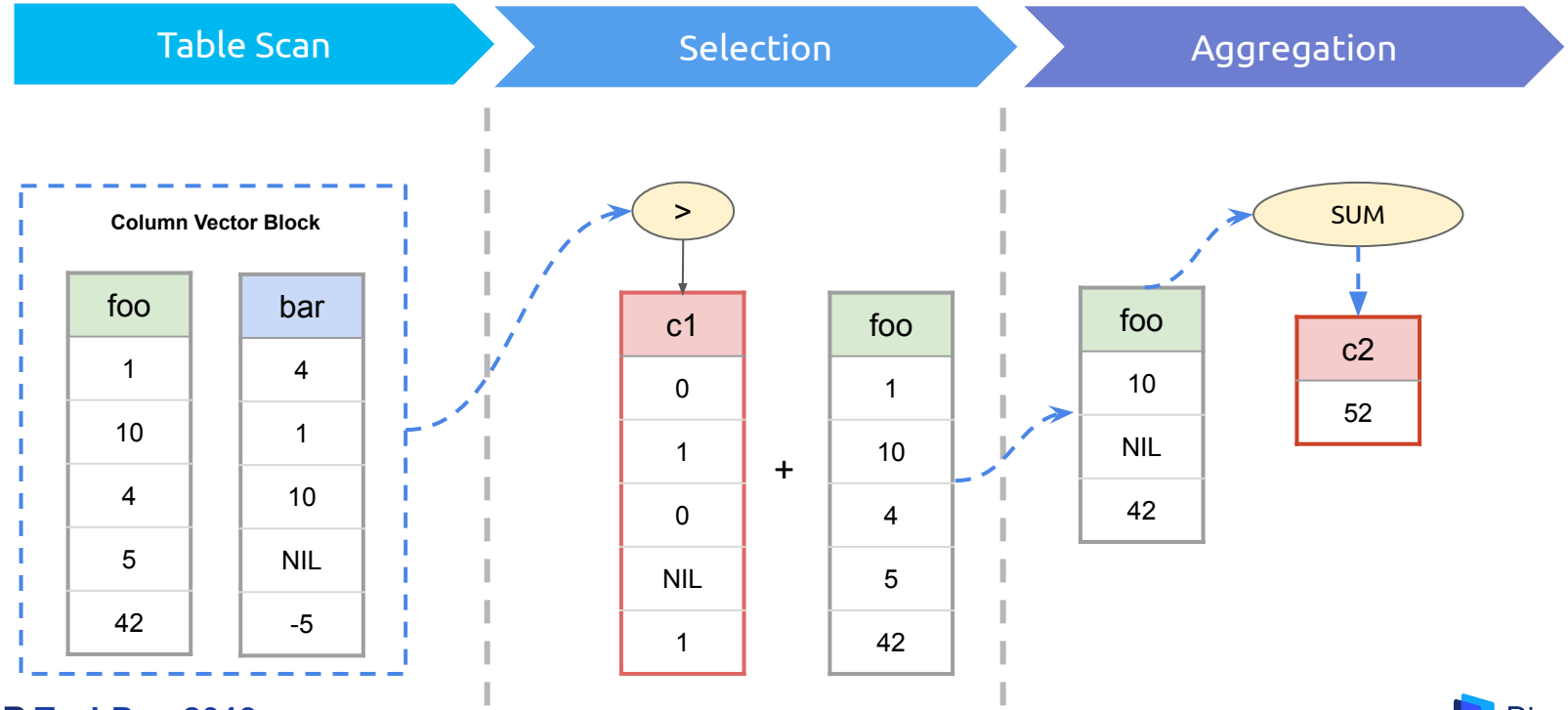
- 配合源于 ClickHouse 的极致向量化计算引擎
  - 更少的废指令，SIMD 加速

更快的查询响应 - 存储

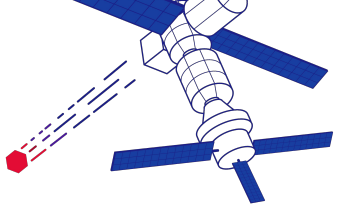

select ... where x between (150, 160)
Write
Write
Level 0
Level 1
...
Level n
LSM-Tree
VS
Write
Write
Write
Stable
Delta
Stable
Delta
...
Stable
Delta
Sement 0
[-inf, 100)
Sement 1
[100, 200)
Sement n
[200, +inf]
DeltaTree

更快的查询响应

SELECT SUM(foo) FROM Table WHERE foo > bar

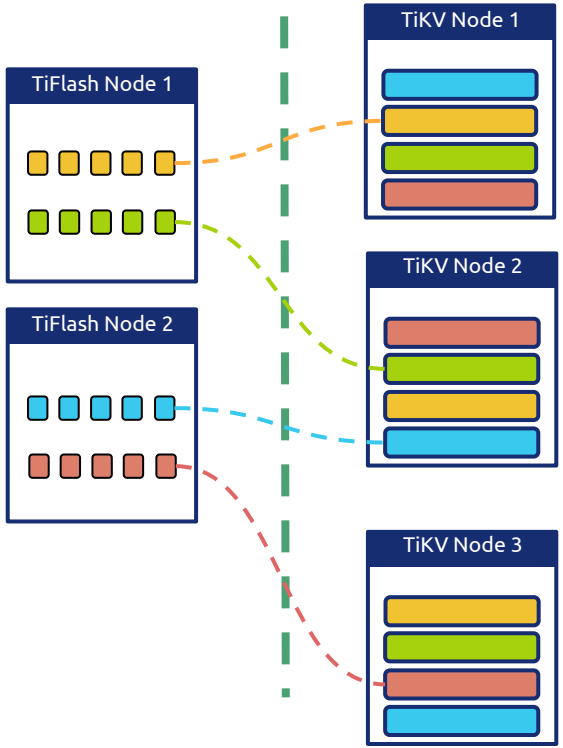


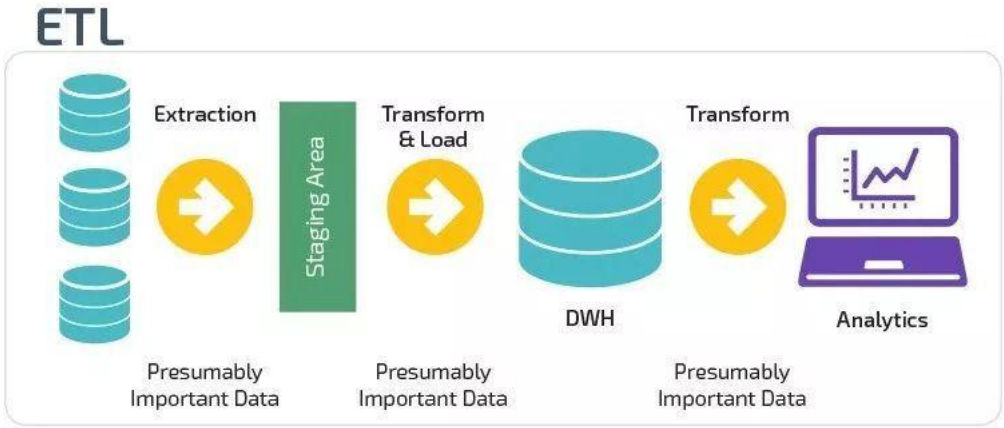
更快获取一致的最新数据



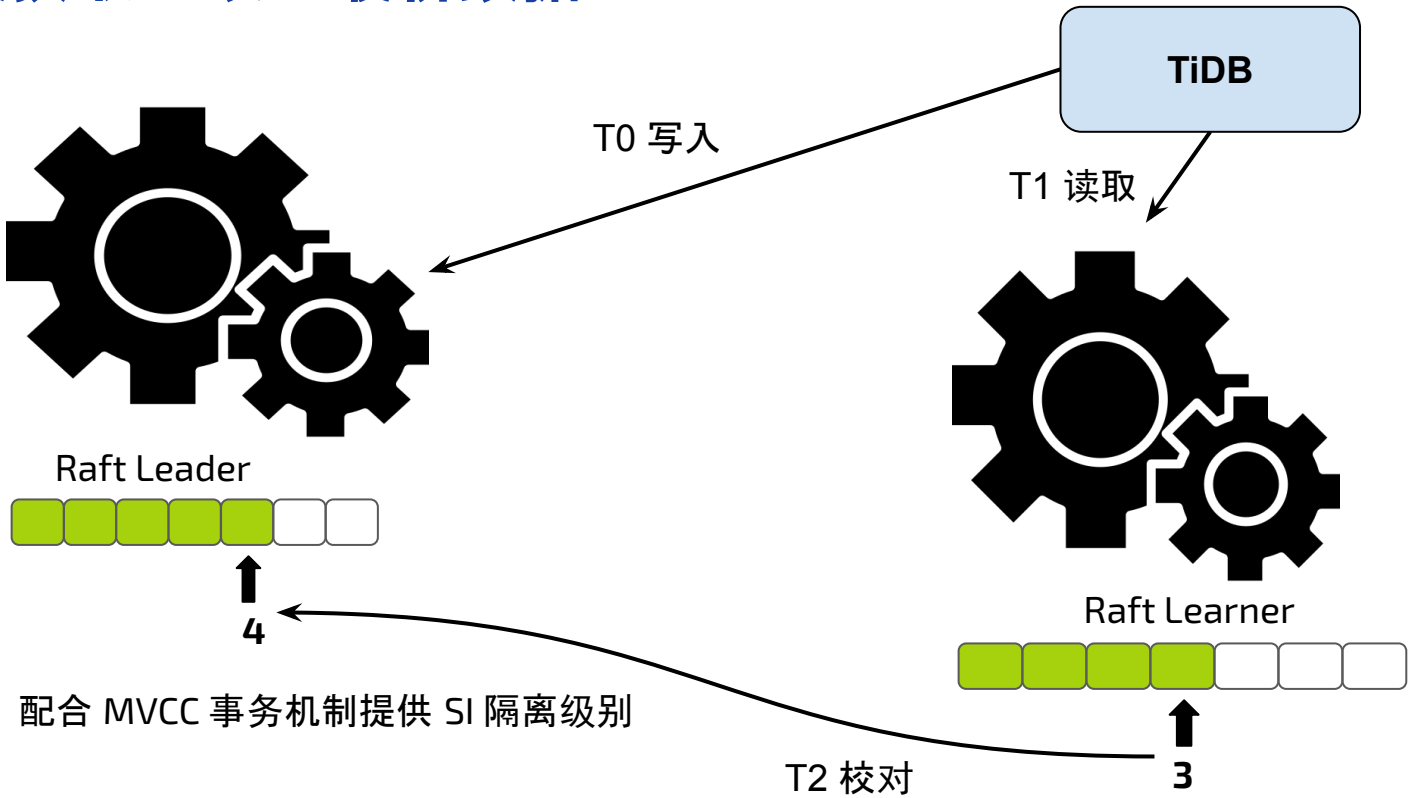
- 实时，且自动容错 + 负载均衡的数据复制
  - 实时到账且对 TP 压力非常小
  - TiFlash 节点稳定性无关 TiKV
  - AP 查询对 TP 性能影响非常小
  - 自带负载均衡，可多对多高效复制
  - 复制协议本身自带容错和数据恢复
  - 更快速的数据获取可以让业务决策更有效
    - 更快的物流周转，更精确的风控，更敏捷的商业策略调整

更快获取一致的最新数据

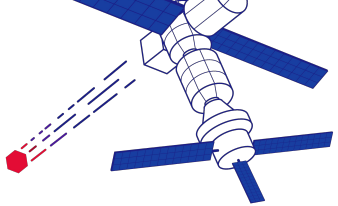




更快获取一致的最新数据

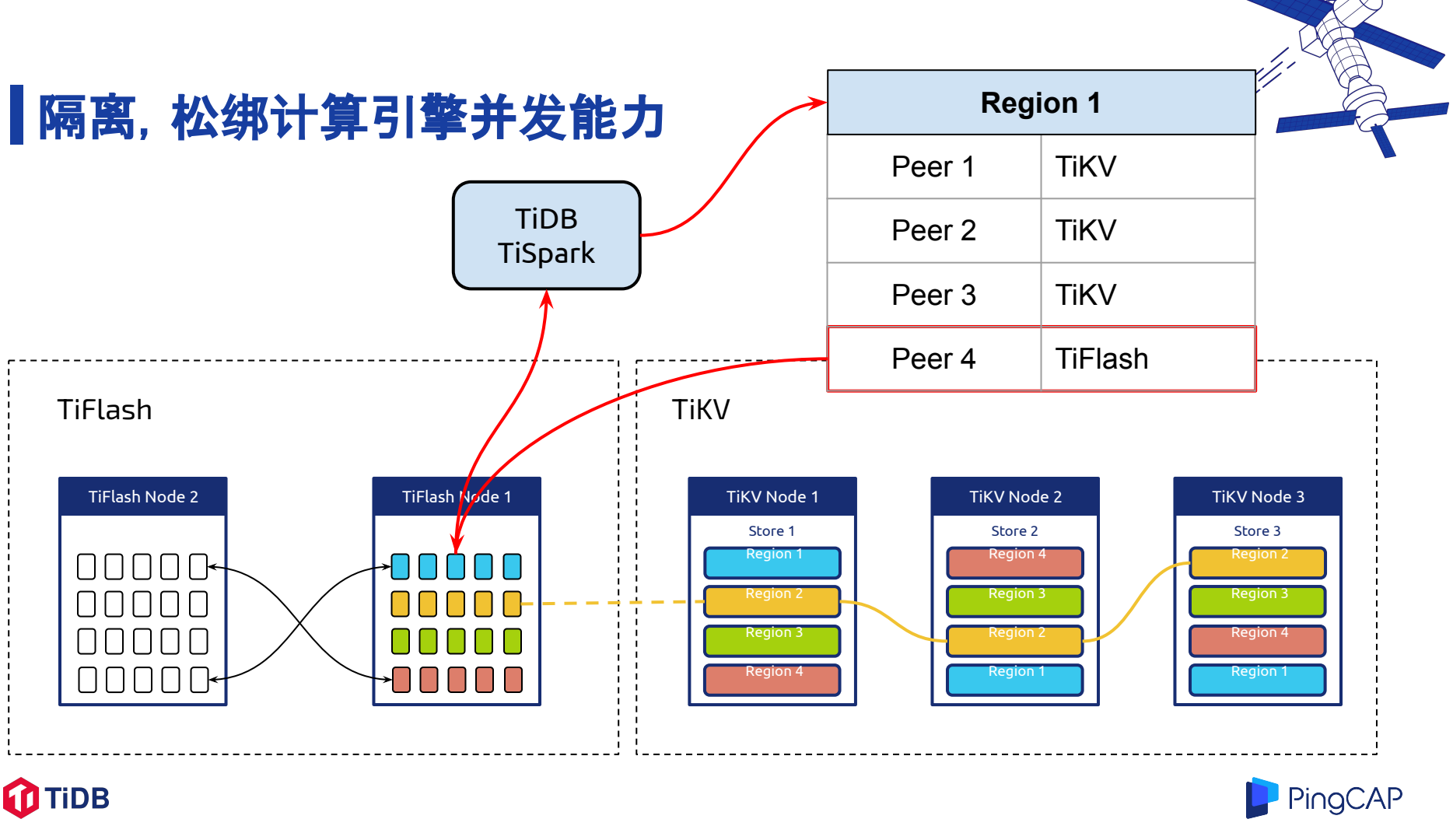


隔离，松绑计算引擎

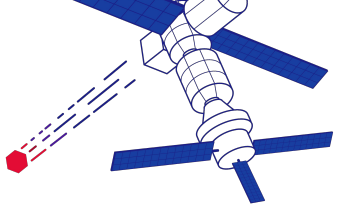


- 拥有良好的隔离能力，分析型查询完全不会干扰在线业务
- 不论是 TiDB 还是 TiSpark 都可以开足火力读取数据
- 业务不再需要担忧实时统计分析查询引起抖动
- 列存的引入也让用户无需为了避免扫表而创建各种索引减慢写入
- 同一套数据同一个平台完成分析和交易两种业务

隔离，松绑计算引擎并发能力



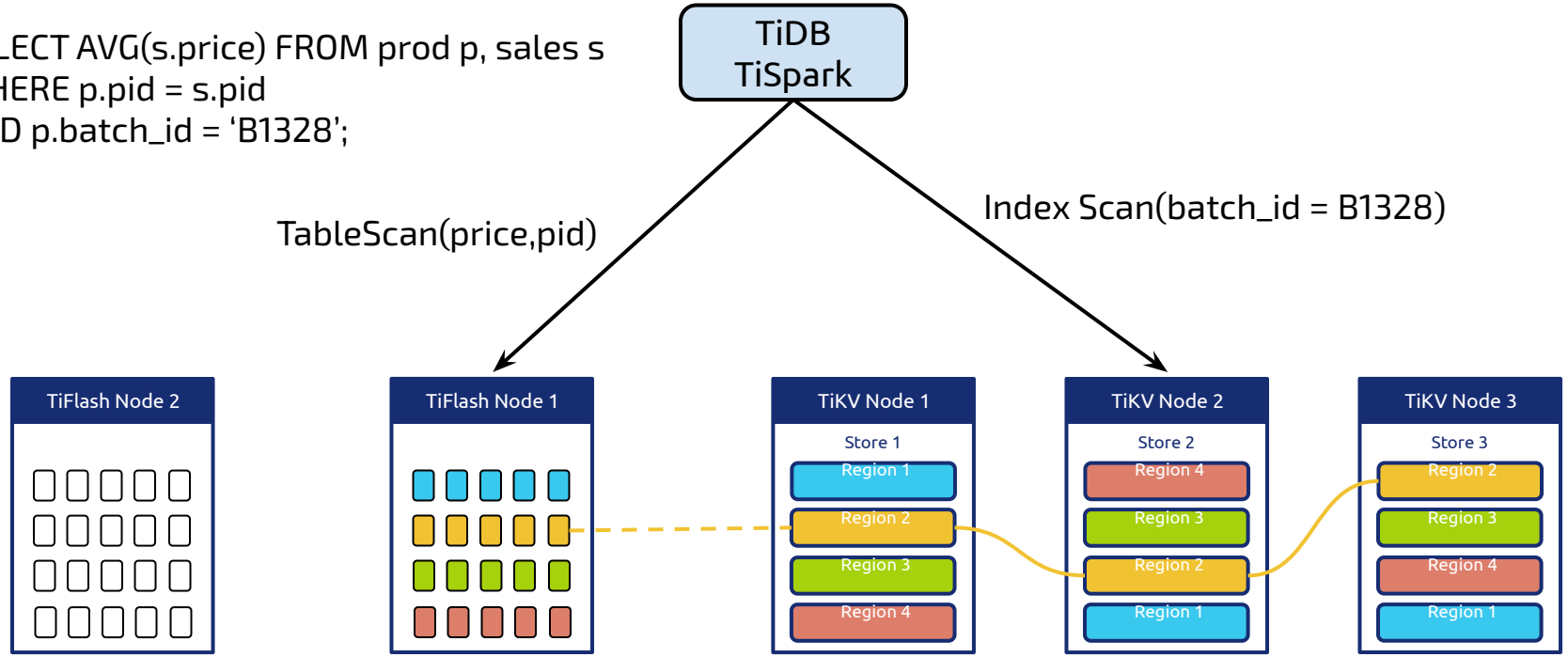
行列配合达到最优速度



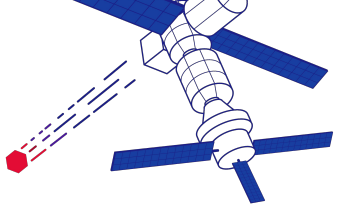
- 不开启隔离时，TiFlash 也可以作为列存索引使用
  - 作为一张表的特殊索引
    - 通过统计信息确定选中了多少数据
    - 列存少量读取代价高，但大量扫表代价低；行存+索引相反
  - 和行存及其现有索引机制联合加速
- 统一由优化器根据代价模型评估
  - 批量扫描且无行存索引或者索引无法有效 过滤数据时
  - 表过宽读取放大严重时

行列配合达到最优速度

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

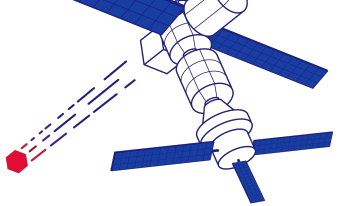


行列配合达到最优速度



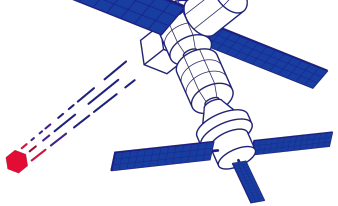
- 行列混合加速，而不仅仅是列存
  - 统一平台满足高并发短查询混合中低并发实时分析和跑批，报表等业务
  - 甚至同一查询可同时使用行列进行加速达到最大效果
  - 例如物流数据平台
    - 大量高并发低延迟点查追踪包裹或者客户信息
    - 同时进行亚秒级的多维分析，例如分省统计某种货物的发单率
    - 由于数据实时到账且一致，两种用法数据完全不会打架
    - 同一套平台，无需切换

更快的业务接入速度



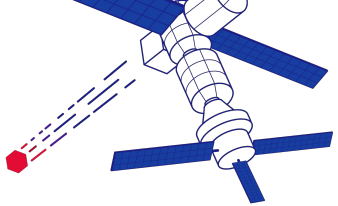
- 亲身经历：某游戏接入 Hadoop 平台进行分析
  - 3 天时间架设数据从分布式数据 库传输变更日志的管道
  - 2 天时间编写将日志入库 Hive 的作业
    - 由于传输管道本身无法保 证一致性，作业本身需要去重，且分区入 库需要校对时间边界
  - 还需要编写数据校对代码，以保证及时发现传输错误，又过去几天
- TiFlash 让用户更早接入服务
  - 快到仅仅需要一条命令开启同步

更快的业务接入速度



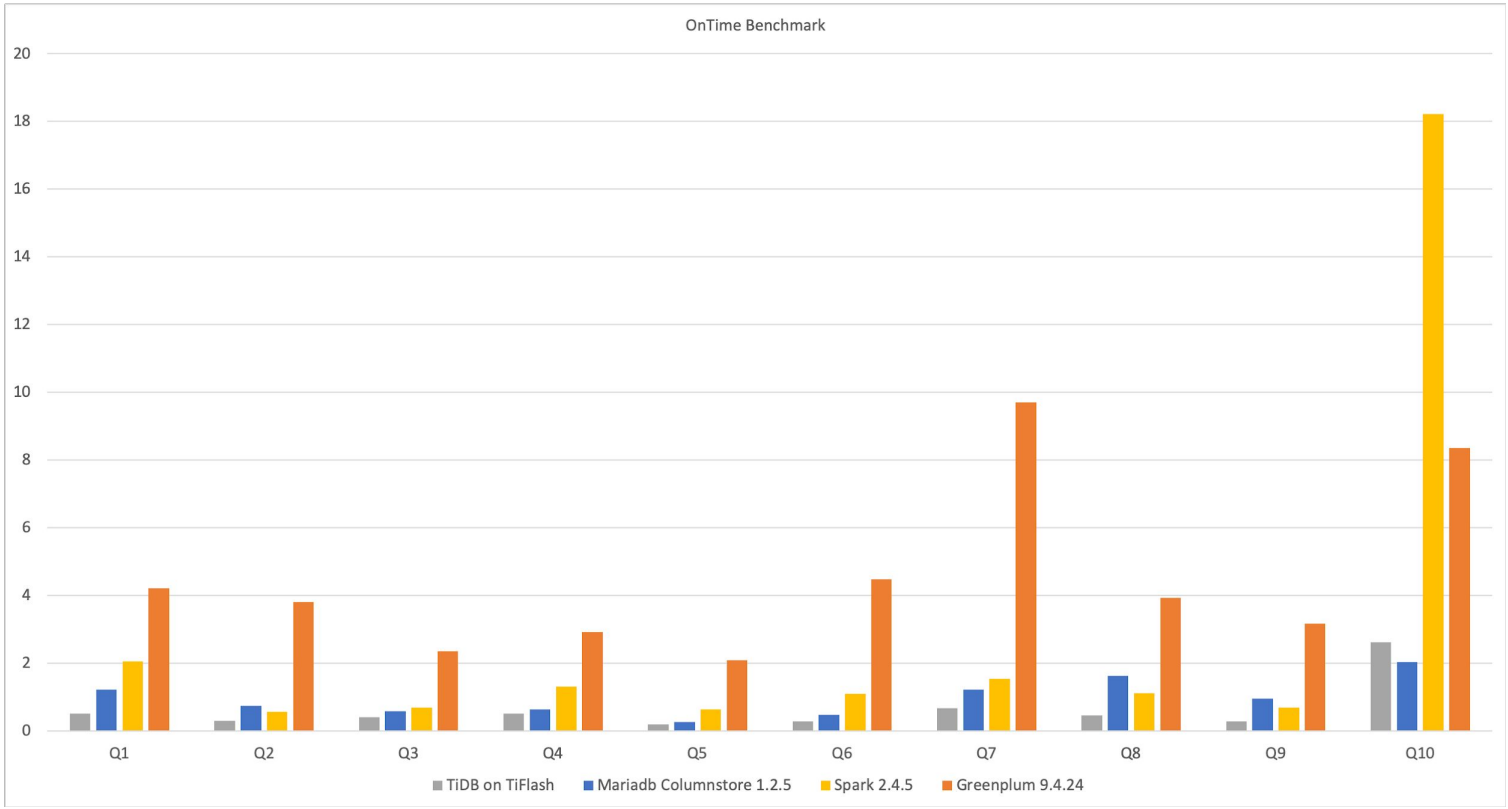
```
mysql> ALTER TABLE orders SET TIFLASH REPLICA 2;
```

硬件部署

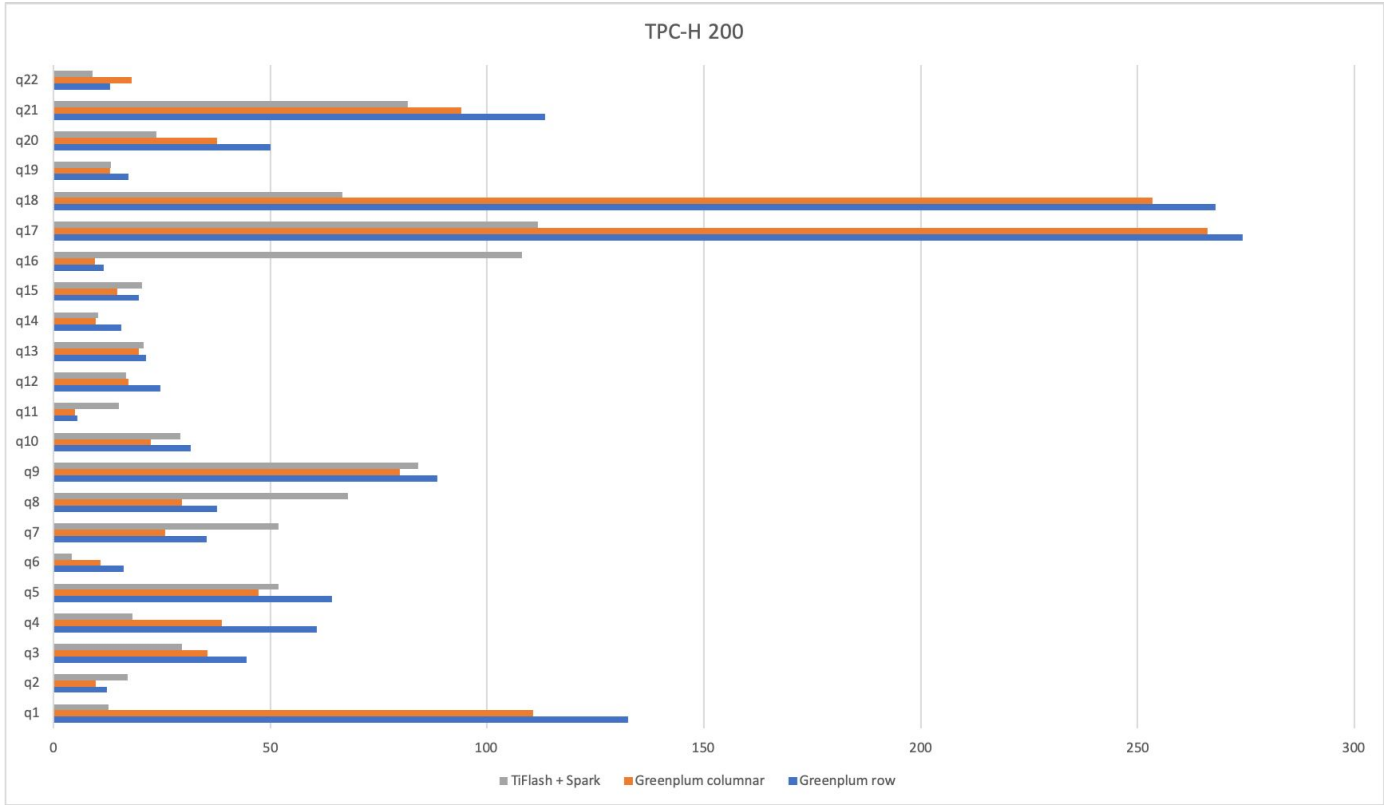


- 最低 8C 16G，可以使用 HDD，需要 10% 容量为 SSD 作为缓冲区
- 推荐 2 副本，如果完全没有 HA 需求也可以 1 副本
  - 1 副本可以保证数据不丢（只要行存数据仍存活）
- 如果不是 TP 场景可以选择和 TiKV 联合部署，但是不要共享磁盘
- 占用空间暂时按照 TPC-H 类测试大约是 2:1（单副本是 KV 的一半）

Benchmark OnTime



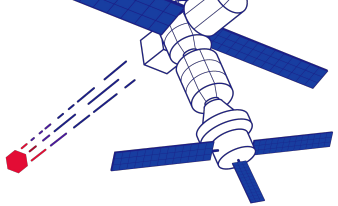
Benchmark TPC-H



PingCAP

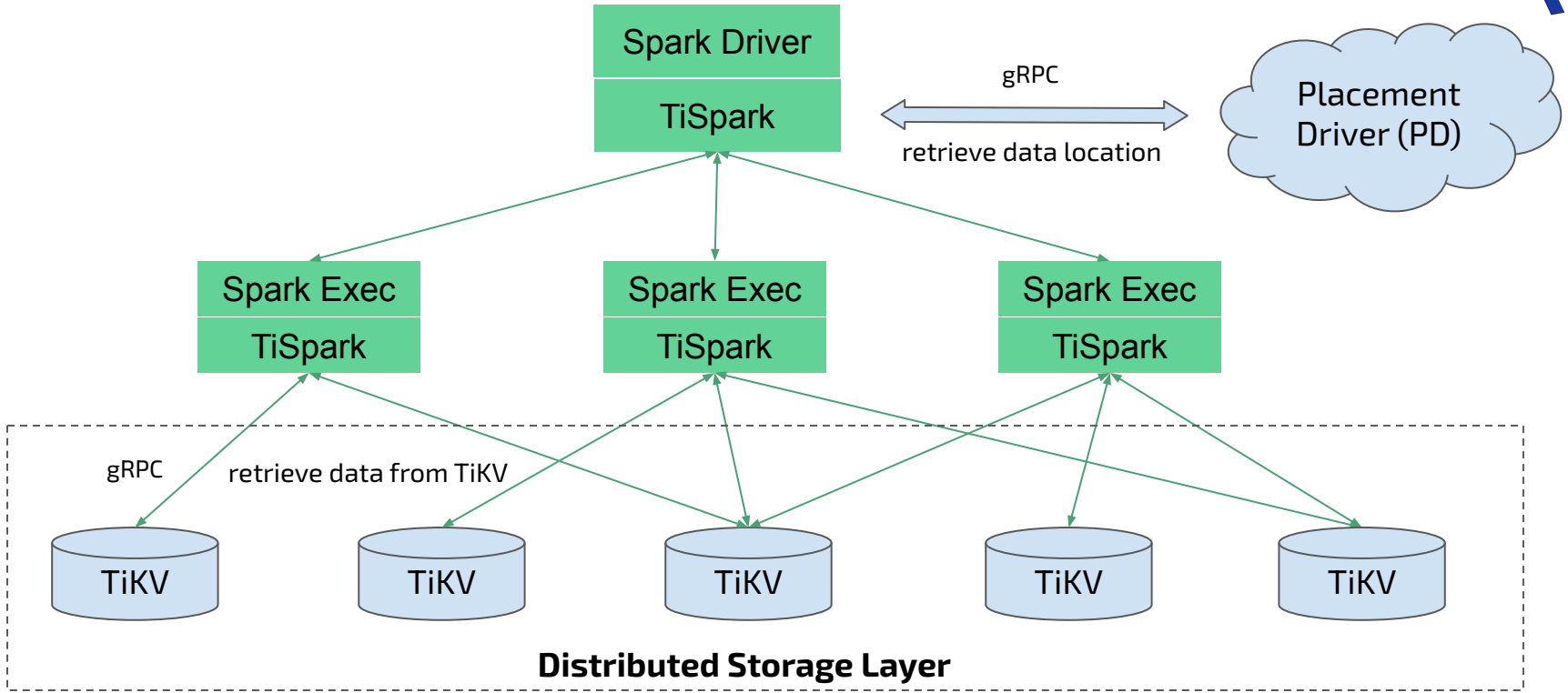
TiSpark

TiSpark



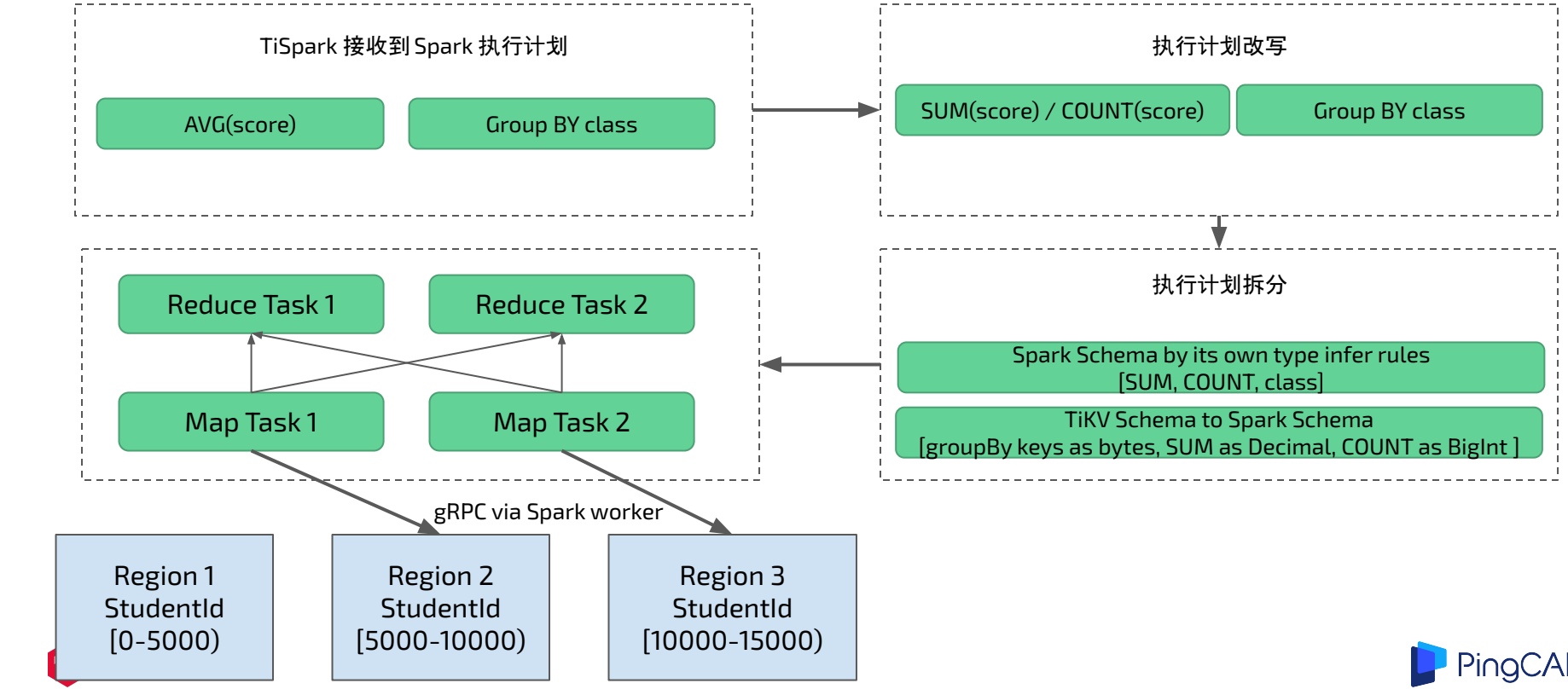
- TiSpark 是运行于 TiDB 存储层上的 Apache Spark 连接器
- 对接 Apache Spark 生态
  - Apache Zeppelin，访问 Hive 仓库，机器学习，R 等等
- 为 TiDB 提供分布式计算框架
  - 突破如大表 Join 的单机资源限制
- 和 TiDB 一样支持复杂计算下推以及索引
- 带有事务支持的分布式批量写入

TiSpark

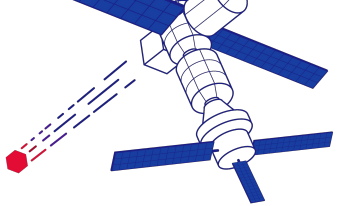

Spark Driver
TiSpark
gRPC
retrieve data location
Placement Driver (PD)
Spark Exec
TiSpark
Spark Exec
TiSpark
Spark Exec
TiSpark
gRPC
retrieve data from TiKV
TiKV
TiKV
TiKV
TiKV
TiKV
Distributed Storage Layer

下推 - 聚合处理

```
select class, avg(score) from student
.....
 group by class ;
```



适用场景



- 大型查询加速
  - 在重型复杂分析场景比 TiDB 有用更好的性能和扩展性
- 联合大数据平台
  - TiSpark 可以同时读写大数据平台和 TiDB 两边的数据，衔接快慢数据层
  - 使用现有的 Spark 自有分析能力和第三方生态工具，例如 PySpark，R，MLlib 或者 Apache Zeppelin
- ETL 场景

PingCAP

Thank You!

Any Questions ?