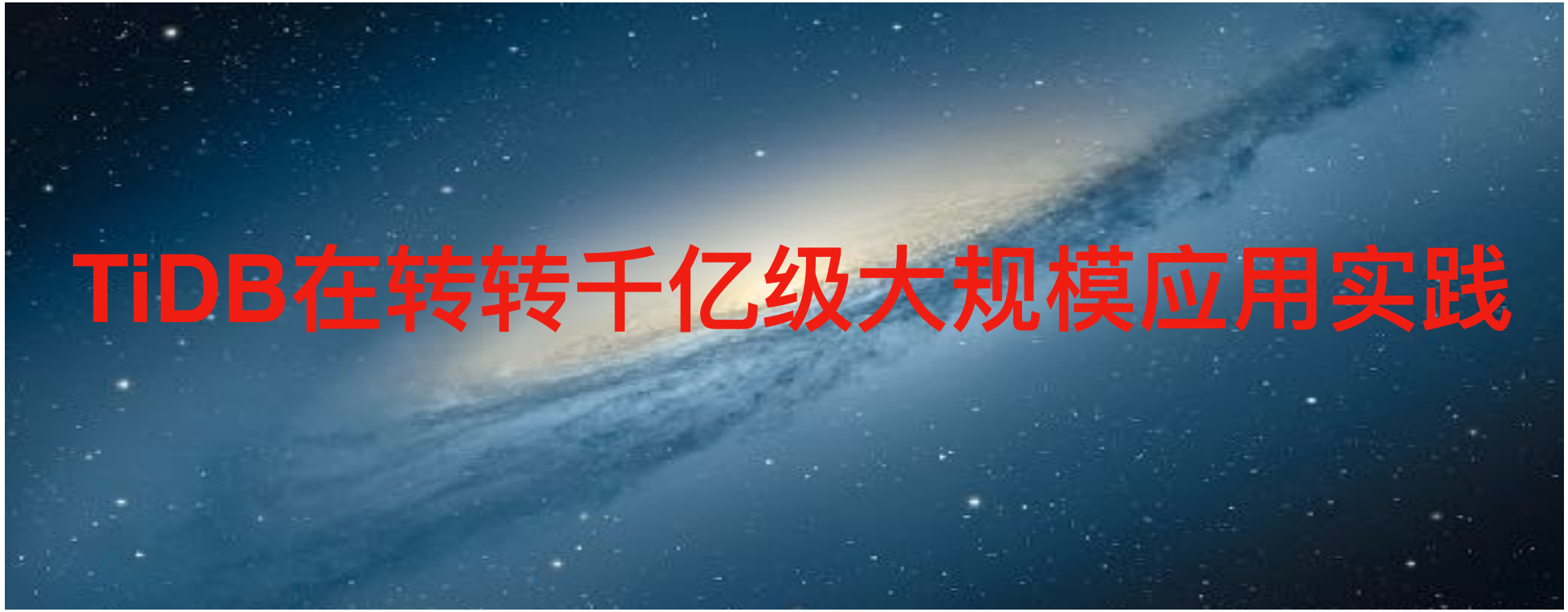


演讲人:孙玄

关于我

转转公司首席架构师

转转公司架构算法部负责人

前58集团技术委员会主席

前58集团高级系统架构师

前百度高级工程师

毕业于浙江大学

代表公司多次对外分享

架构之美（beautyArch）公众号作者

关于我

QCon 全球软件开发大会

DTCC 2016中国数据库技术大会

Artificial Intelligence Conference

SDCC 中国软件开发者大会

TOP1 技术型企业案例研究智库

Strata+Hadoop World

WOT World Of Tech

程序员 Programmer

分享要点

1. 转转业务场景
2. TiDB替代传统分库分表
3. TiDB应用实践经验
4. 接入现状/微信钱包流量暴增表现/规划

二手交易平台-转转

二手交易平台-转转

业务场景


转转有书
男生，家有小公主：3岁11个月，加入转转1019天，他有4个好友在转转。图书正版保证，品相无明显硬伤，经过专业清洁消毒。现在买书，超低折扣，满29包邮！
+ 关注
实人认证
他的交易
全部 200
12
买到宝贝
999+
卖出宝贝
新***侠
好评!
用户


消息中心
通知助手
留言消息
李正_tfr
赵加琪
[图片消息]
锦B_1yy
已买
买家136****3232f
消息


看推荐
逛附近
转转有书
逻辑工作法 （日）西村克己 图书 工作学习 励志 工...
黄晓指h
苹果 iPhone8 金色 64G 其他版本 有配件 有包
商品


我买到的
全部
待付款
待发货
待收货
交易成功
共1件商品 实付款：¥139.00
联系卖家
共1件商品 应付款：¥3204.00
联系客服
交易

使用TiDB代替传统分库分表

- 面临问题

✓**MySQL+MongoDB**

✓大数据量性能瓶颈

✓业务层Sharding

✓业务成本

✓业务逻辑复杂性增加（多维度Mapping以及性能）

✓运维成本

✓DDL

✓故障切换时间长&高可用方案

使用TiDB代替传统分库分表

- 面临问题总结

✓数据量大，如何快速水平扩展存储

✓大数据量下，如何快速DDL

✓业务层分库分表造成业务逻辑非常复杂

✓常规MySQL主从故障转移造成业务不可用

✓开源高可用运维方案不友好

使用TiDB代替传统分库分表

- 为什么选择NewSQL

✓RDBMS->NoSQL->NewSQL

✓Sharding语义下推

✓业务开发简单

✓运维成本降低

✓TiDB

TiDB应用实践经验

- **TiDB特点**

✓开源分布式NewSQL关系型数据库

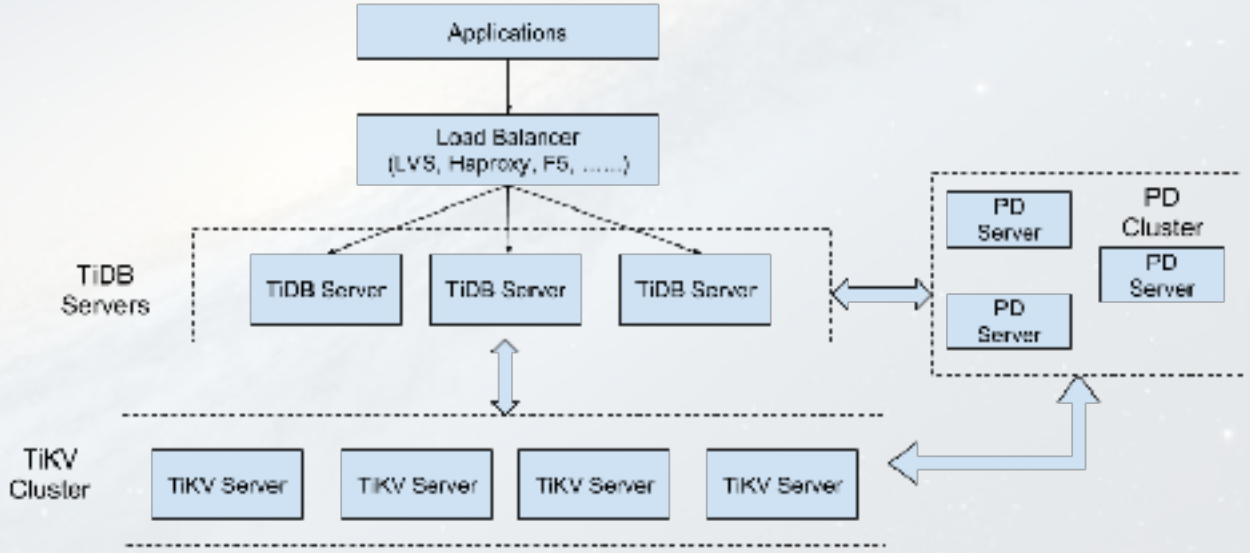
✓自动水平伸缩

✓强一致性分布式事务

✓基于Raft算法的多副本复制

✓高度兼容MySQL协议

✓……



TiDB应用实践经验

- **TiDB测试-功能测试**

  - ✓支持绝大数据MySQL语法
  - ✓存储过程 / 自定义函数 / 触发器等除外
  - ✓基于MySQL业务无缝迁移

TiDB应用实践经验

- **TiDB测试-压力测试**

✓硬件环境

✓3台CPU密集型物理服务器，启动TiDB Server及PD Server

✓3台IO/CPU密集型PCIe物理服务器，启动TiKV Server

TiDB Server 节点服务器	
服务器型号	Huawei RH1288 V3
CPU	Xeon CPU E5-2630 v3 2 x 8 cores
内存	2400MHz*16GB, 128GB
磁盘	SAS 1.5T
网卡	Intel I350 Gigabit, 1Gb/s
操作系统	Linux CentOS 7.4

✓软件环境

✓数据库版本：TiDB-v1.1.0

✓压测工具：Sysbench-1.0.11

✓监控：TiDB 配套监控（prometheus、grafana）、Zabbix

Tikv 节点服务器	
服务器型号	HP ProLiant DL380 Gen9
CPU	Xeon CPU E5-2630 v4 2 x 10cores
内存	2400MHz*16GB, 128GB
磁盘	PCIe 1.5T
网卡	Intel I350 Gigabit, 1Gb/s
操作系统	Linux CentOS 7.4

TiDB应用实践经验

- **TiDB测试-压力测试**

  - ✓测试方案

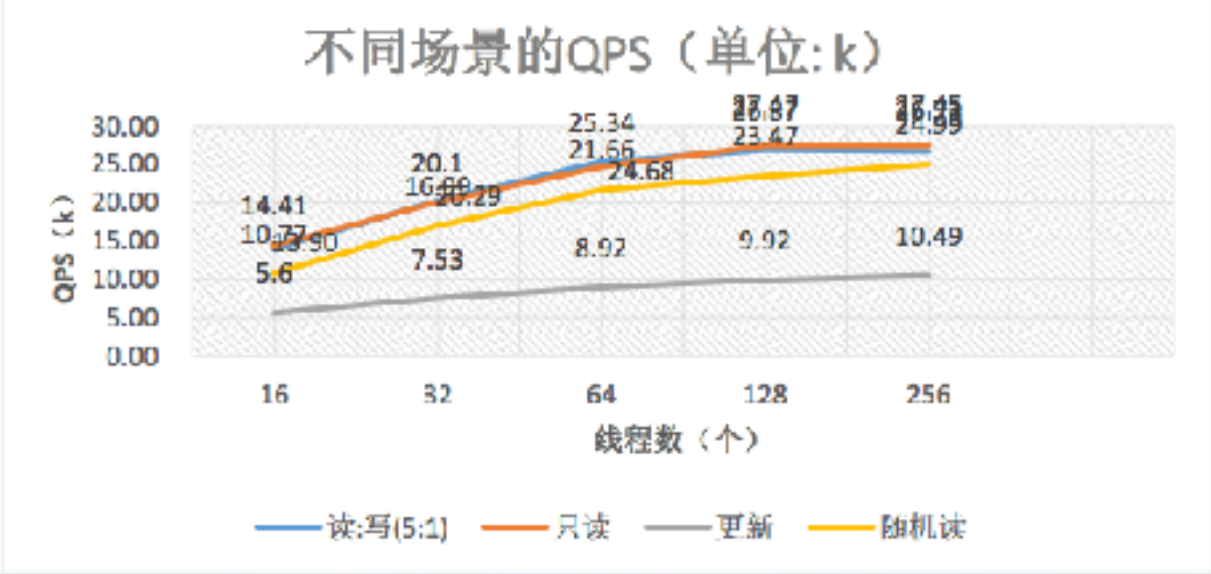
    - ✓使用sysbench-1.0.11，调整线程数，请求读写比例，模拟不同场景请求

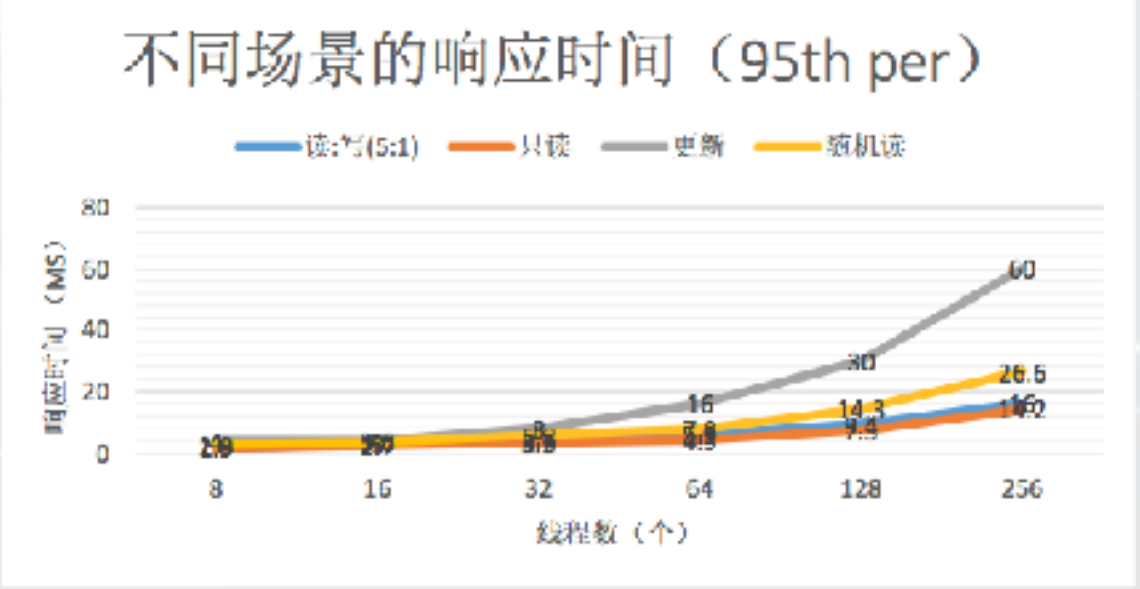
    - ✓测试数据大小200GB

TiDB应用实践经验

- TiDB测试-压力测试

✓测试结果





TiDB应用实践经验

- TiDB测试-压力测试

  - ✓场景建议

    - ✓适合线上业务混合读写场景

    - ✓适合顺序写场景：数据归档、操作日志、摊销流水等

TiDB应用实践经验

- **TiDB预上线方案**

✓TiDB挂载到线上MySQL，作为MySQL从库同步线上数据

✓首先将业务部分读流量切换到TiDB，做好预判，逐步灰度读流量到100%

✓然后切换业务部分写流量到TiDB，做好预判，逐步灰度流量到100%

TiDB应用实践经验

- 线上业务接入

✓第一个接入TiDB转转消息服务

✓消息是转转最重要基础服务，保证买卖双方有效沟通/交易

✓联系人列表、个人消息表、系统消息表

✓数据量&访问量大

✓几十T&千亿

✓MySQL分库分表

✓二次拆分

TiDB应用实践经验

- 线上业务接入步骤

  - ✓测试

    - ✓判断TiDB业务场景满足

    - ✓性能

  - ✓同步数据

  - ✓切流量

TiDB应用实践经验

- 线上业务接入步骤

✓测试验证

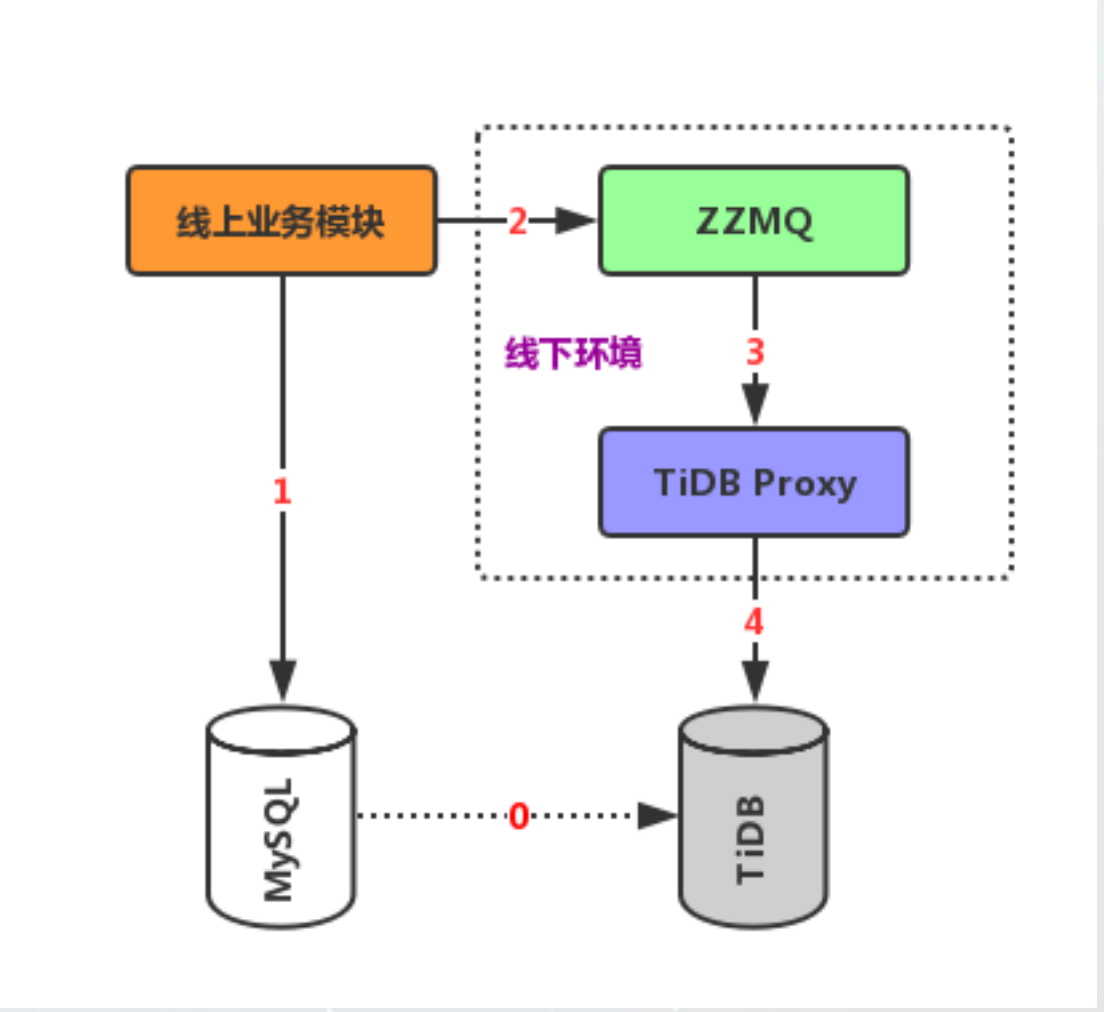
✓引流线上数据和流量到线下

✓大量功能和性能验证

✓日志、耗时

✓DBA数据抽样正确性验证

✓TiDB完全满足消息业务需求



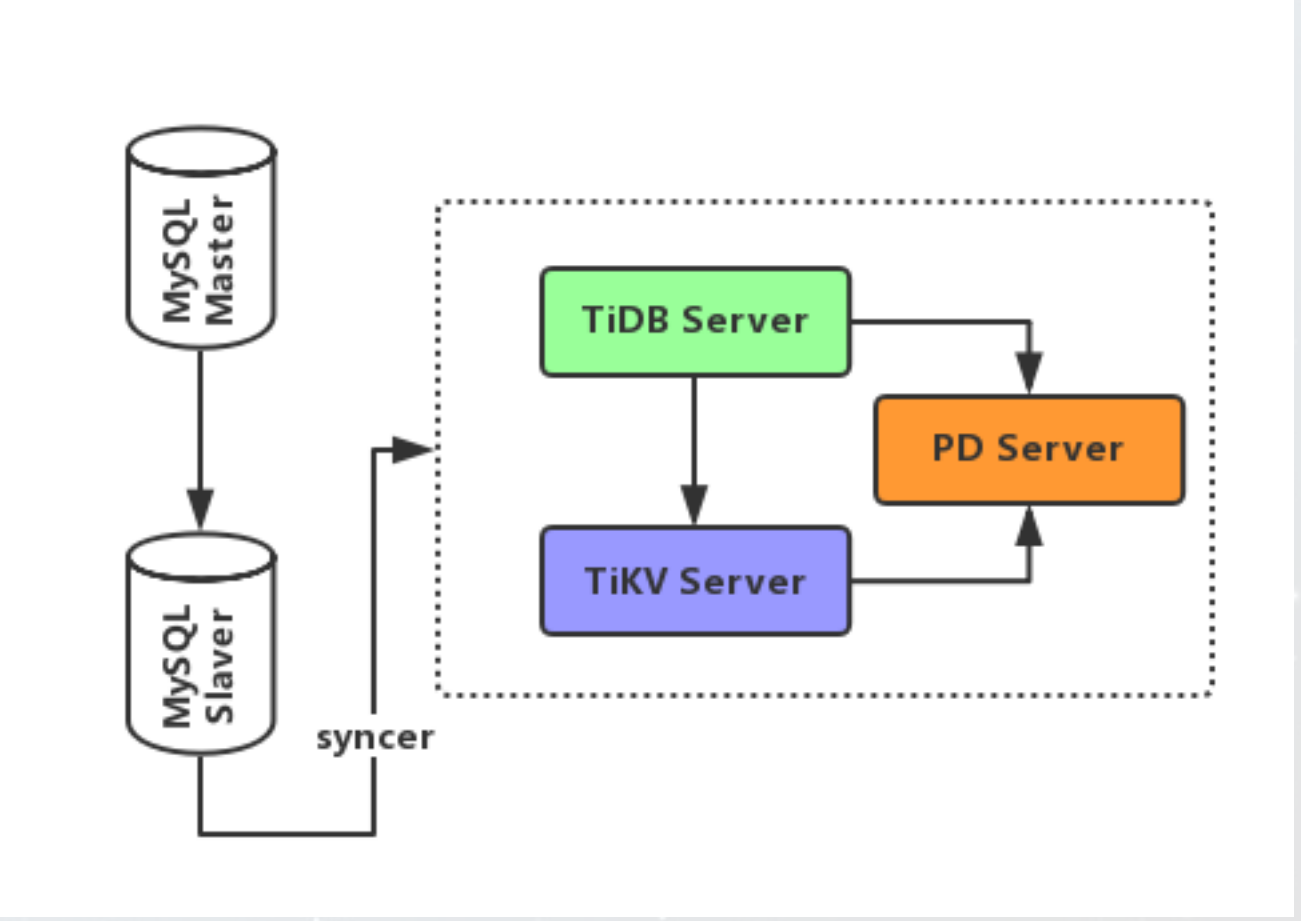
TiDB应用实践经验

- 线上业务接入步骤

✓同步数据

✓TiDB集群作为MySQL实例从库

✓MySQL单实例1024表同步TiDB一张大表



TiDB应用实践经验

- 线上业务接入步骤

✓切流量

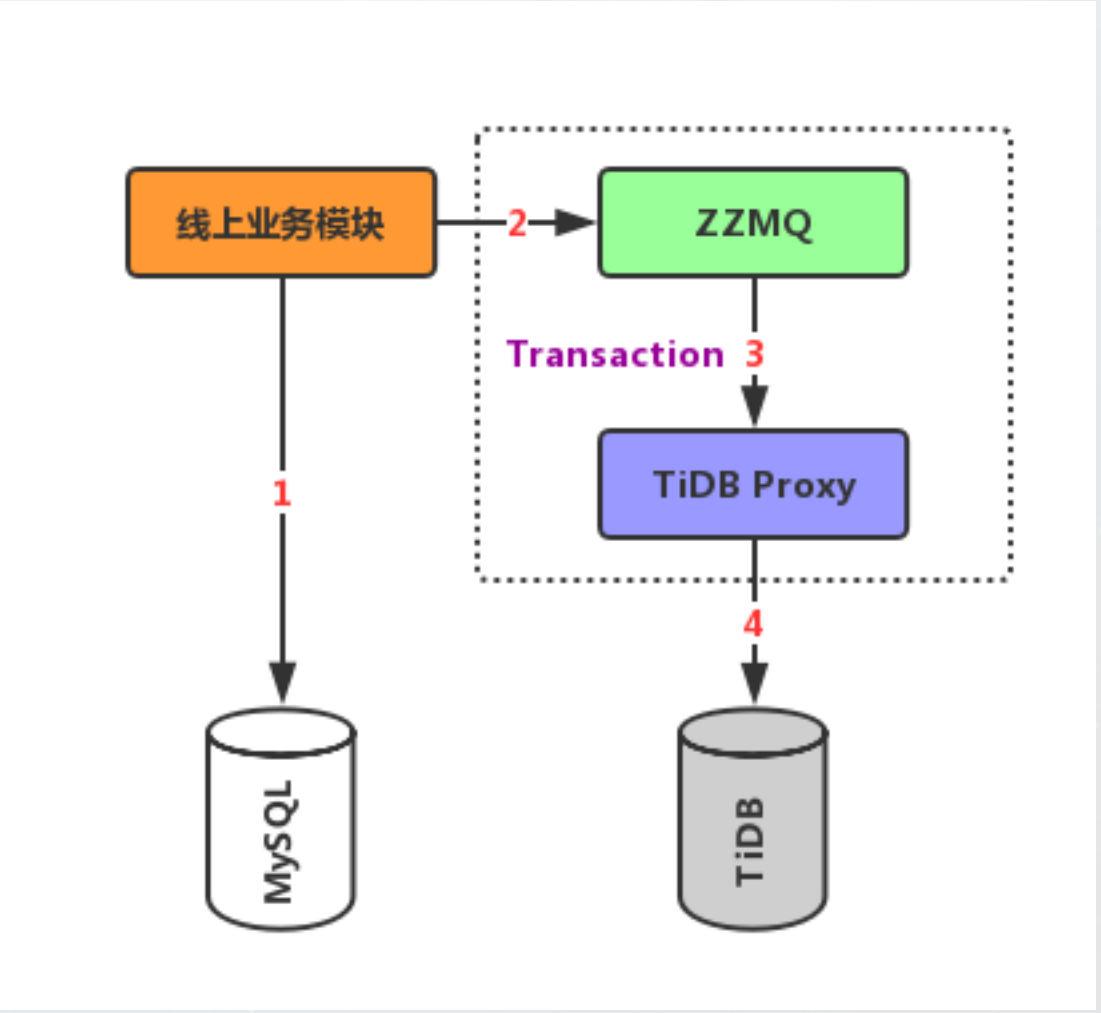
✓第一步将读流量灰度切到TiDB，观察一周，

逐步灰度至100%

✓第二步断开TiDB与MySQL主从同步，

业务双写（确保业务随时回滚到MySQL）

✓第三步业务停止MySQL写入



TiDB应用实践经验

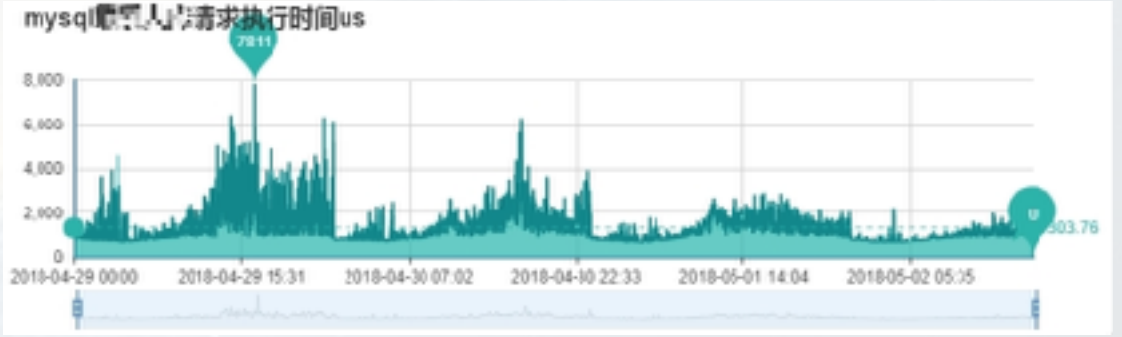
- 线上业务效果对比

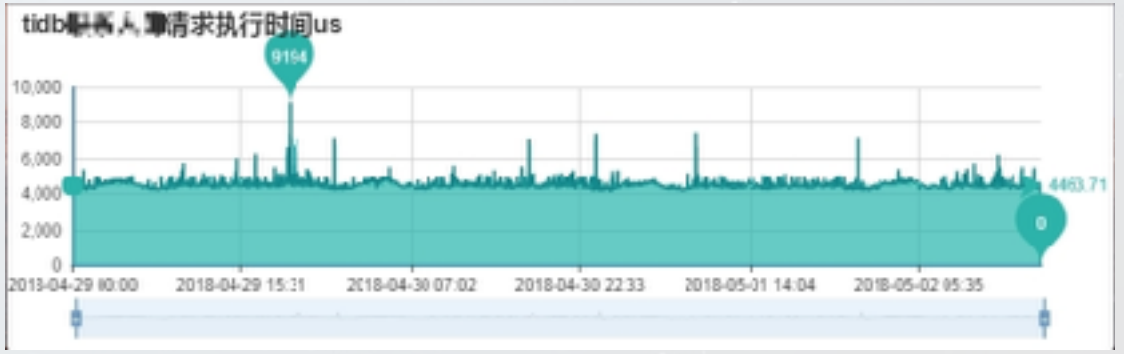
✓数据库请求延时情况

✓TiDB 整体响应延时非常稳定，不受业务流量高峰影响

✓ MySQL 波动很大

✓ 扩展性，TiDB/TiKV通过无缝实例提升吞吐量





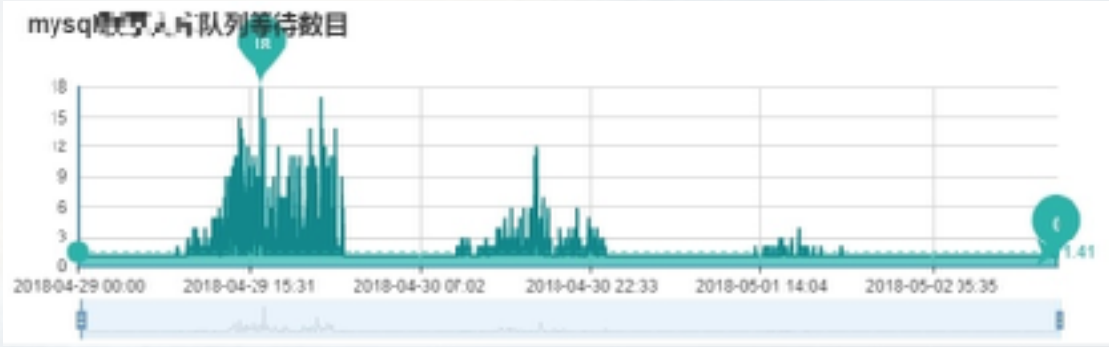
TiDB应用实践经验

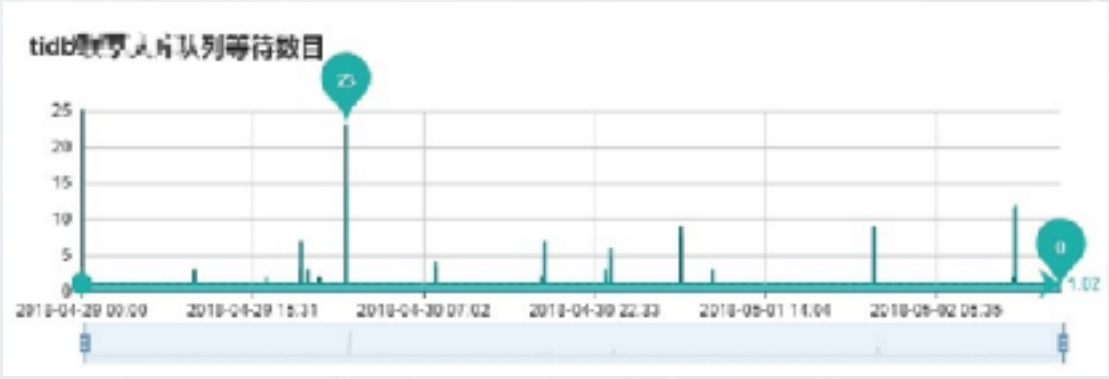
- 线上业务效果对比

✓业务请求队列等待情况

✓TiDB 恒为1

✓MySQL抖动大





TiDB应用实践经验

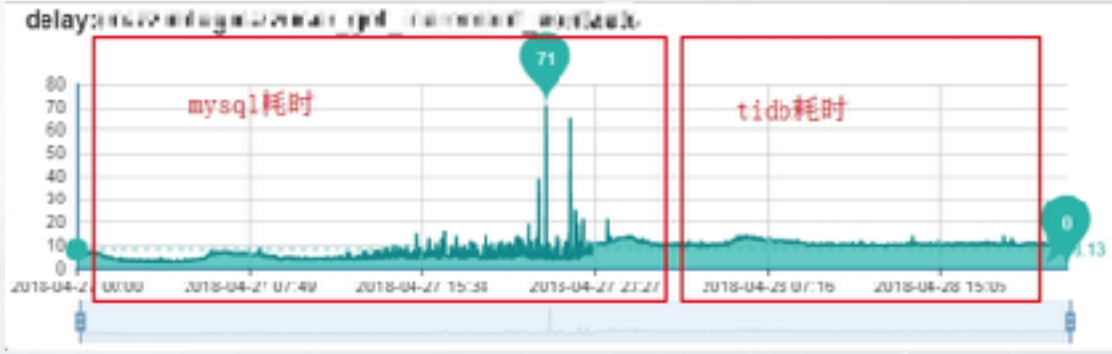
- 线上业务效果对比

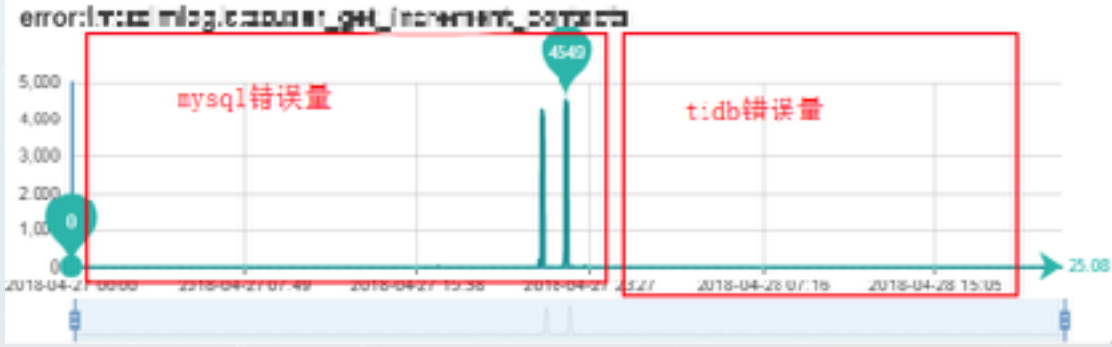
✓业务延时和错误量情况

✓接入 TiDB 后业务服务接口耗时稳定无抖动

✓且没有发生丢弃的情况

✓下图错误由数据访问层服务队列堆积发生请求丢弃





TiDB应用实践经验

- **遇到问题及其解决方案**

  ✓整体非常稳定

  ✓问题第一时间得到官方团队帮助并解决掉

TiDB应用实践经验

- 问题一：锁机制差异

✓起因

✓token字段是唯一索引

✓大量INSERT请求，token为空

✓影响

✓请求延时高（几十秒）

✓原因

```
CREATE TABLE `user_dev_token` (
  `uid` bigint(20) unsigned NOT NULL,
  `token` varchar(128) NOT NULL,
  `timestamp` bigint(20) unsigned NOT NULL,
  PRIMARY KEY (`uid`),
  UNIQUE KEY `UK_TOKEN` (`token`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8
```

```
INSERT INTO user_dev_token(uid,token,timestamp)
VALUES（942279400,'65E1A305D76D9E28875D',
1530081302499) ON DUPLICATE KEY UPDATE
token='65E1A305D76D9E28875D',timestamp=1530
081302499;
```

```
INSERT INTO user_dev_token(uid,token,timestamp)
VALUES（942279400,'',1530081302499) ON
DUPLICATE KEY UPDATE
token='',timestamp=1530081302499;
```

TiDB应用实践经验

- 问题一：锁机制差异

✓解决方案

```
$ grep 'limit' *.yml | grep retry
tidb.yml:   # retry-limit: 10
```

✓关闭retry_limit

✓业务优化（过滤使用默认值token=‘’请求）

✓Bug 2.0 RC 5已修复，insert on duplicate key update 语句性能提升10倍

TiDB应用实践经验

- 问题二：Truncate大表

✓起因

✓DBA 按照 MySQL 运维经验，对一个近 T 的表做了 Truncate 操作

✓操作后，起初数据库表现正常

✓几分钟后，TiKV Server负载变高

✓影响

✓业务超时

TiDB应用实践经验

- 问题二：Truncate大表

```
tikv.yml:  use delete range: false
```

✓原因

✓大表Truncate操作触发了频繁回收 Region的 BUG （delete-region）

✓解决方案

✓TiDB 2.0 版本已经修复

TiDB应用实践经验

- 问题三：Order by createTs Desc

✓起因

✓ 例如：获取商品最近评论的前10条

✓ SELET comment FROM commentTable WHERE infoid = 88 order by createTs desc limit 10;

✓ createTs为bigint类型

✓影响

✓业务耗时增加

TiDB应用实践经验

- 问题三： Order by createTs Desc

✓原因

✓默认排序是ASC，DESC需要排序

✓解决方案

✓不优雅的解决方法：将 createTs 转换为负值（0 - createTs）， 默认可走ASC

✓TiDB 2.0.4 release 版本已经优化

TiDB应用实践经验

- **问题四：update操作row Affect**

  - ✓**起因**

    - ✓ 业务多线程update记录状态，均返回成功，其中未生效语句，也返回成功

●问题四：update 操作 row Affect

时间轴	Transaction 1	Transaction 2	备注
T1	Begin; Query OK, 0 rows affected (0.00 sec)	Begin; Query OK, 0 rows affected (0.01 sec)	
T2	update t1 set status = 3 where id = 10086 and status = 2 ; Query OK, 1 row affected (0.00 sec)	update t1 set status = 4 where id = 10086 and status = 2 ; Query OK, 1 row affected (0.00 sec)	MySQL 中 Query OK, 0 row affected (0.00 sec)
T3	Commit; Query OK, 0 rows affected (0.01 sec)		
T4		Commit; Query OK, 0 rows affected (0.01 sec)	

TiDB应用实践经验

- **问题四：update 操作 row Affect**

  - ✓**原因**

    - ✓MySQL悲观锁，TiDB乐观锁，锁冲突TiDB重试导致返回结果不正确

  - ✓**解决方案**

    - ✓设置一个 system variable，禁用 TiDB 的自动重试

TiDB应用实践经验

- 问题五：部署PD多实例问题

  ✓起因

    ✓DBA升级集群时， 误将另一个集群的PD-server关停

  ✓影响

    ✓集群不可用时长1~2min

TiDB应用实践经验

- 问题五：部署PD多实例问题

✓原因

✓被关停集群使用的是systemd管理方式（不支持PD多实例，默认pd.service）

✓部署多PD实例时（手动启动PD，不会更新pd.service），默认的systemd只管理已部署的PD集群

✓再使用ansible调整新部署集群时（更新PD），会kill老的PD集群

✓解决方案

✓建议独立环境部署PD

✓多PD实例建议使用supervise管理方式

接入现状

- 已接入情况

✓已接入11套集群

✓IM

✓Spam

✓Trade 2套

✓欢乐送

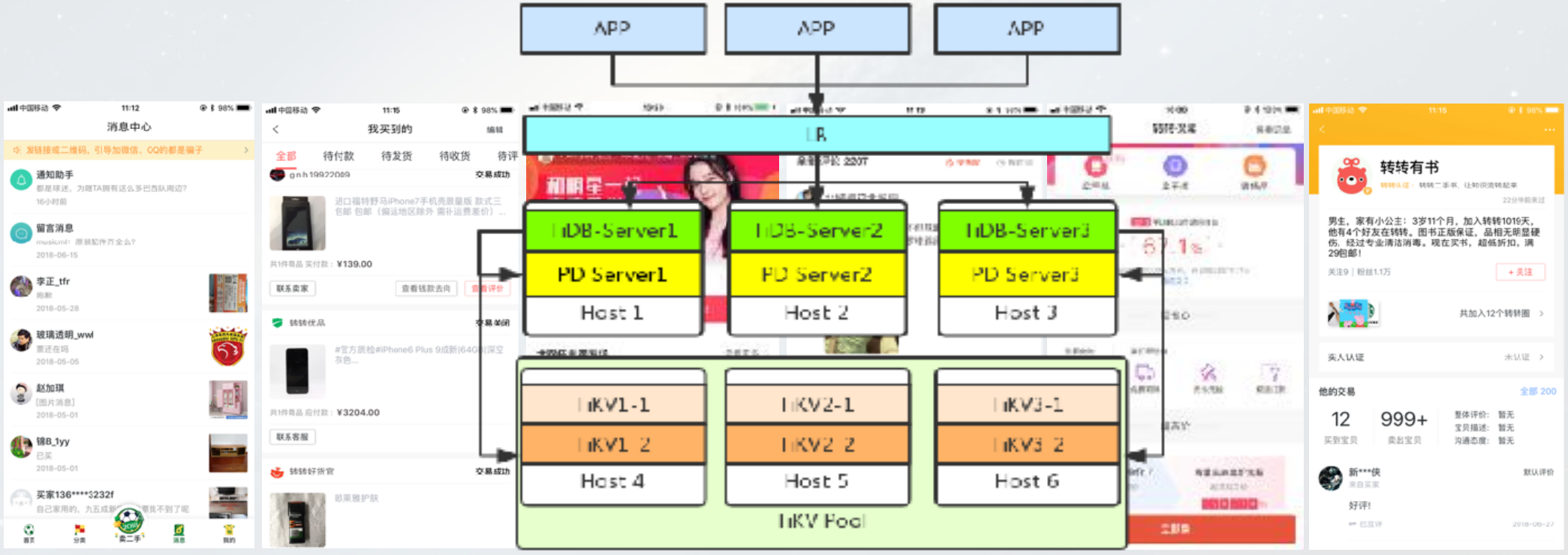
✓留言

✓保卖

✓用户 2套

✓商品

✓商业



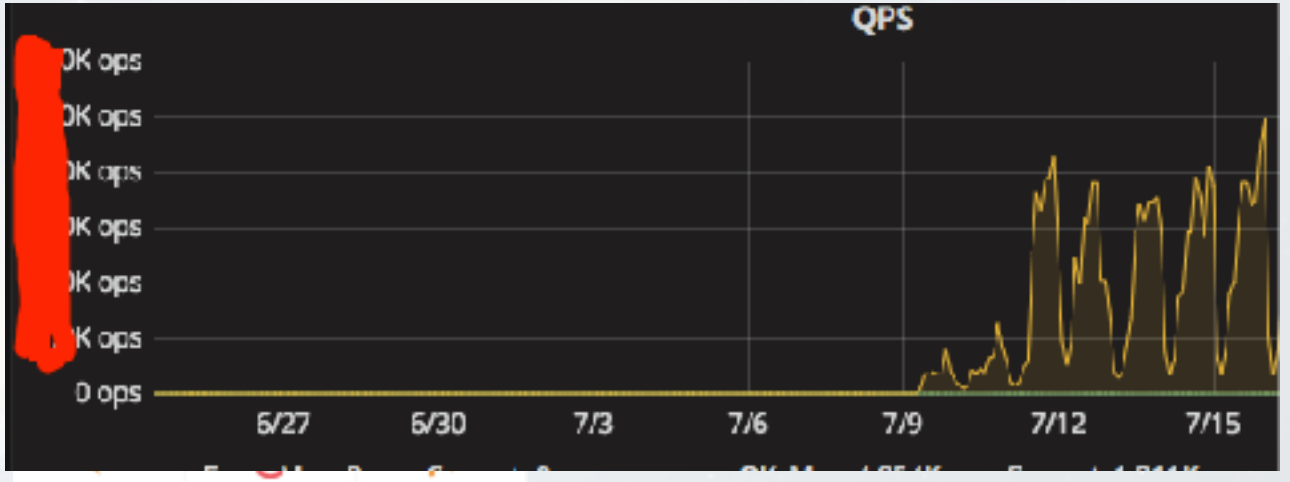
微信钱包小红点流量暴增表现

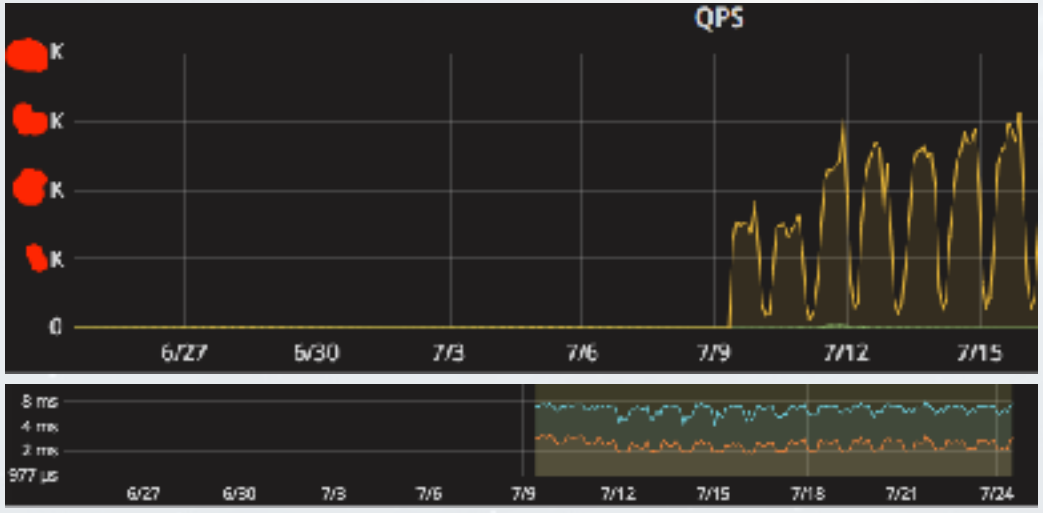
- 微信钱包接入

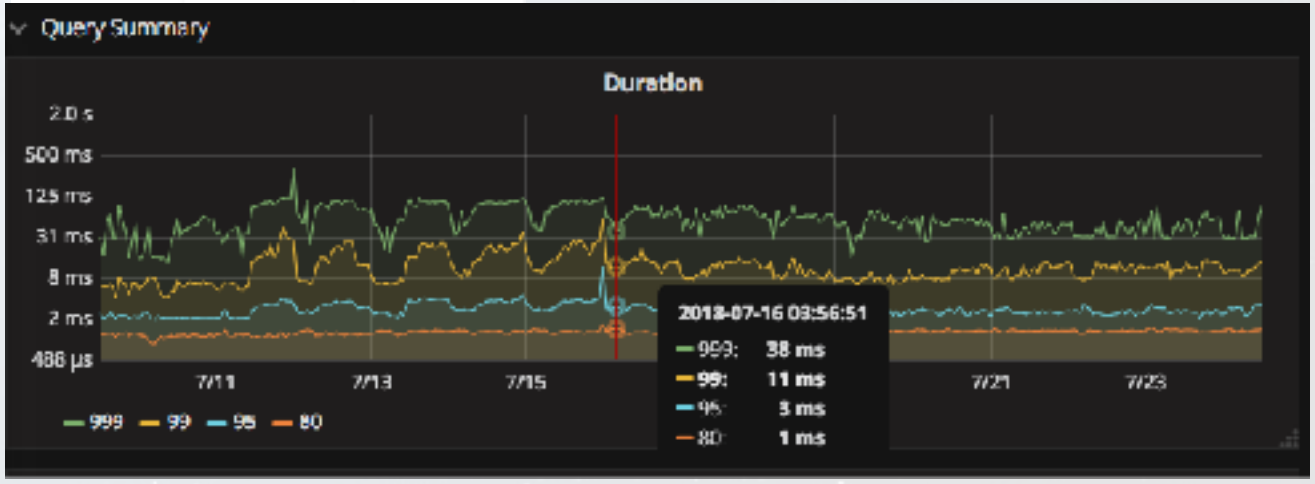
✓线上表现

✓业务流量增大近10倍

✓响应延迟基本没变化







接入规划

- 未来接入

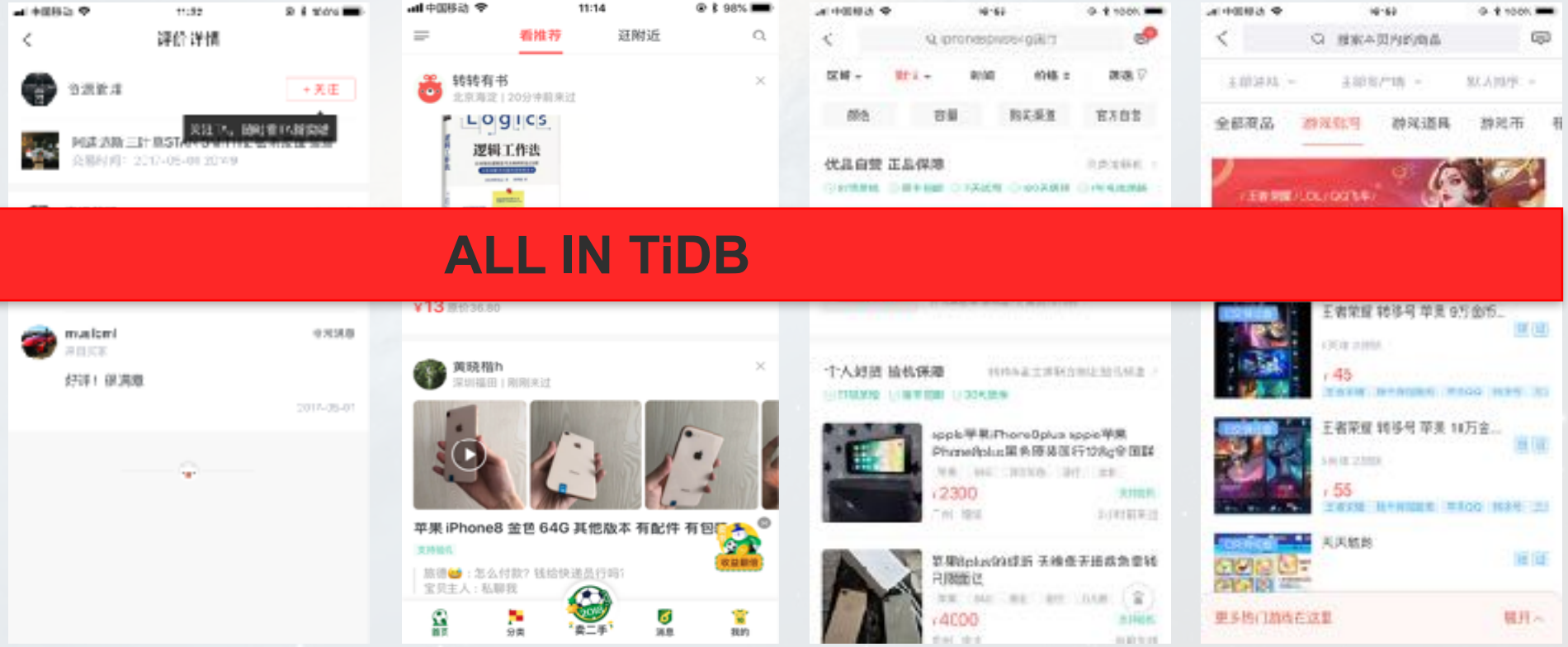
✓计划接入

✓评论

✓推荐/搜索

✓垂直业务线

✓……



ALL IN TiDB

要点回顾

1. 转转业务场景
2. TiDB替代传统分库分表
3. TiDB应用实践经验
4. 接入现状/微信钱包流量暴增表现/规划

欢迎关注本人公众号“**架构之美**”