

中立安全·赋能产业

UDB-灵活可变的数据库服务实践

UCloud优刻得 技术总监 罗成对

云数据库UDB的发展

产品全景

运营现状

成长型业务的数据库诉求

业务特点分析

痛点与共性需求

UDB解决之道

UDB的“不变”、“变”与“可变”

UDB核心产品理念

“不变”、“变”、“可变”的取与舍

UDB赋能案例解析

All-in-One：自适应性能调优

抢红包：多维度弹性可扩展

618/双11：差异化保障与持续可用

云数据库UDB的发展



2019TTC

产品全景

分布式	UDDB			UDW	-	Sharding Cluster
高可用	高可用UDB					Replica Set
分支	MySQL 5.1、5.5、5.6、5.7	Percona 5.5、5.6、5.7	MariaDB 10.0	PostgreSQL 9.4、9.6、10.4	企业版 2012R1	MongoDB 2.4、2.6、3.0、3.2、3.4、 3.6、4.0
协议	MySQL			PostgreSQL	SQL Server	MongoDB
分类	SQL (RDBMS)					NoSQL

UDB的“不变”、“变”与“可变”



UDB核心产品理念



“不变”、“变”、“可变”的取与舍

2019TTC

UDB核心产品理念



UDB的“不变”——云数据库愿景

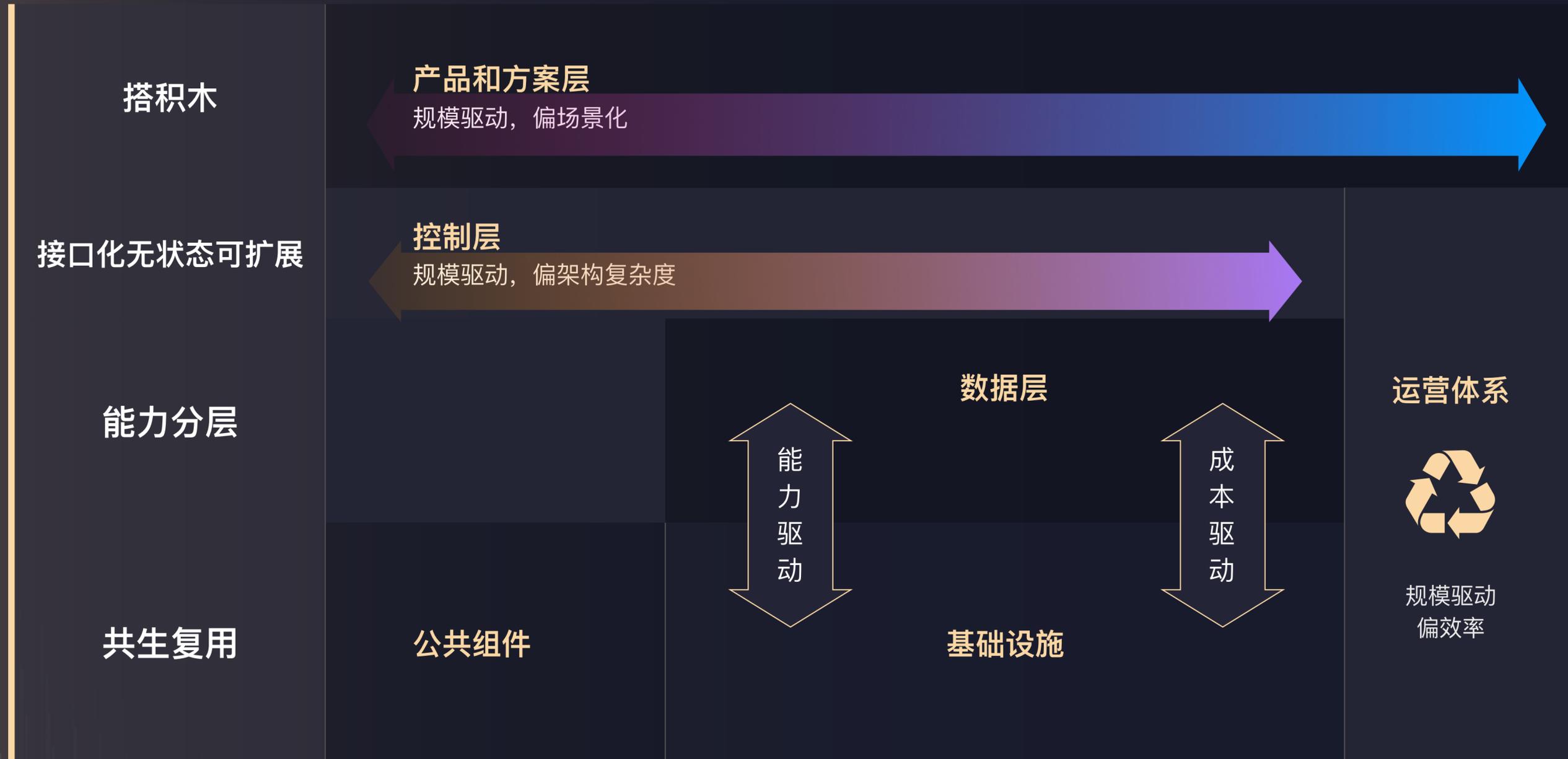
降低
获取门槛

降低
使用门槛

UDB的“不变”——云数据库能力模型



UDB的“不变”——云数据库能力模型



UDB的“变”——演进



自研产品/技术

产品	DBaaS (UDB)	CloudDBA
中间件	读写分离 (完全兼容MySQL协议)	数据传输UDTS
数据库	分布式数据库UDDB (高度兼容MySQL协议)	新一代分布式数据库UXDB (完全兼容MySQL 8.0协议)
存储	新一代超高性能分布式存储 (100TB、100万IOPS)	

UDB的“可变” — 差异化



升级高可用 ✕

升级高可用根据数据量大小需要一定时间，如果长时间处于升级高可用中或升级高可用失败状态请联系技术支持。当数据库状态为待切换时，请在操作项中选择切换至高可用完成切换。

数据库名称	magodo_test_cgroup	付费方式	年付 ▼
数据库类型	高可用版		1年 ▼
当前配置	数据库版本: mysql-5.6 内存: 2GB 硬盘: 50GB SSD机型		预计费用 ⓘ 240.00元

取消 确定

UDB的“取”

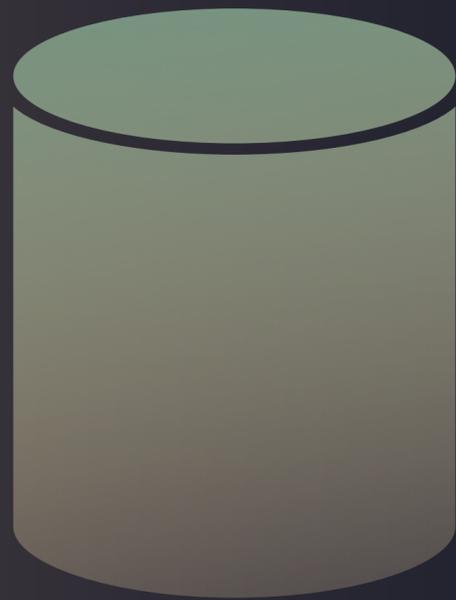


- OOM策略
- 自研内存控制算法

- 定制机型
- 合理的buffer/pool比例

- 赠送内存
- 多轮降价

UDB的“取”



内存最大降幅 **37%**

海外最大降幅 **41%**

只读实例磁盘最大降幅 **47%**

三节点MongoDB降幅 **45%**

仲裁节点 **免费**

UDB的“舍”

MySQL

```
mysql> show grants for root@'%';
+-----+
| Grants for root@% |
+-----+
| GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION |
+-----+
```

```
mysql> SELECT version();
+-----+
| version() |
+-----+
| 5.7.16-uc1oudrel1-log |
+-----+
1 row in set (0.00 sec)

mysql> SET GLOBAL sql_mode='ONLY_FULL_GROUP_BY,STRICT_
Query OK, 0 rows affected (0.00 sec)

mysql>
```

MongoDB

```
{
  "role" : "superuser",
  "db" : "admin",
  "isBuiltin" : false,
  "roles" : [
    {
      "role" : "root",
      "db" : "admin"
    },
    {
      "role" : "backup",
      "db" : "admin"
    },
    {
      "role" : "restore",
      "db" : "admin"
    }
  ]
}
```

成长型业务的数据库诉求

业务特点分析

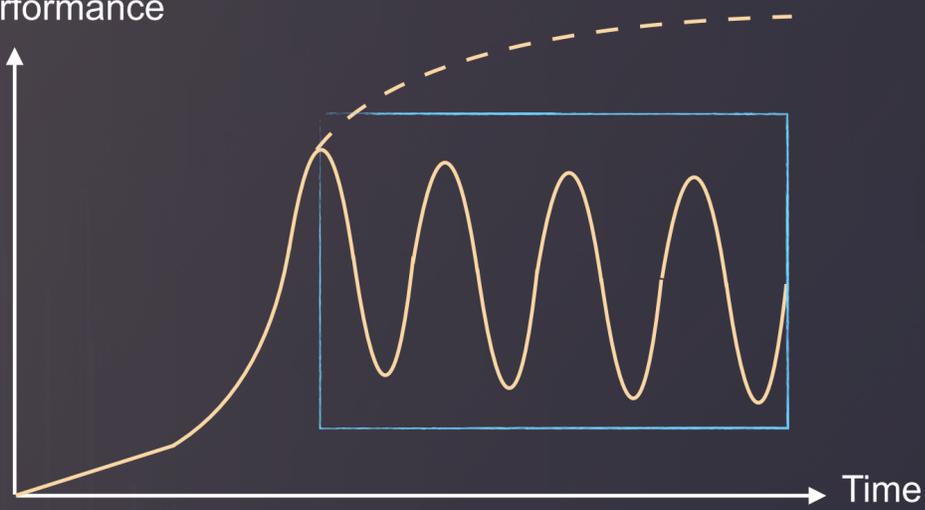
痛点与共性需求

UDB解决之道

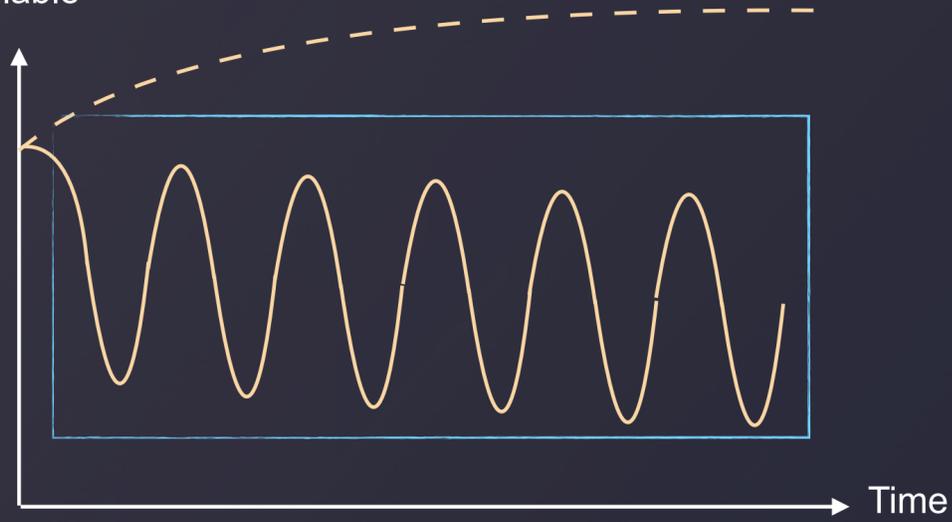
2019 TIO

成长型业务特点分析

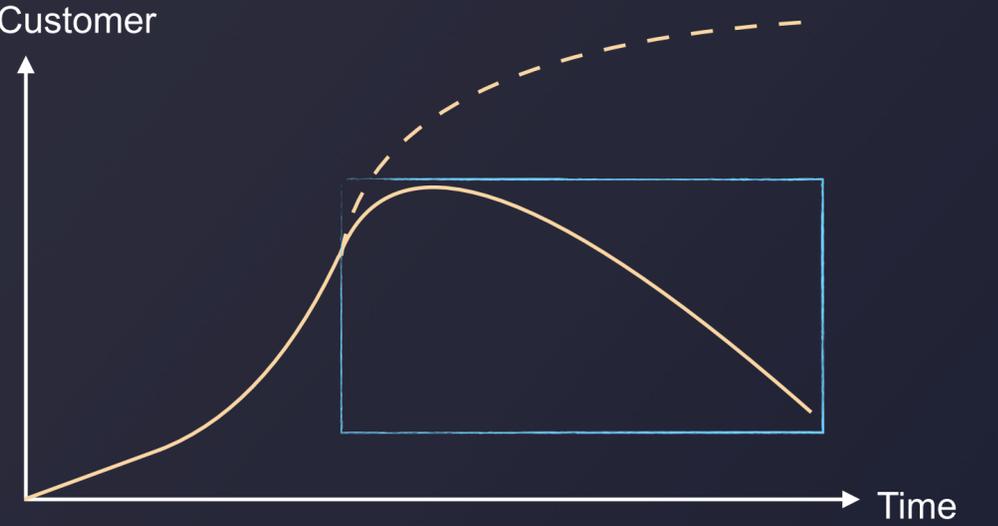
Throughput
/Performance



Available



Value
/Customer



痛点与共性需求

 业务优先
但稳定性变未知

 业务弹性
但数据库不弹性

 规模上量
但运维效率跟不上

先抗住

再优化

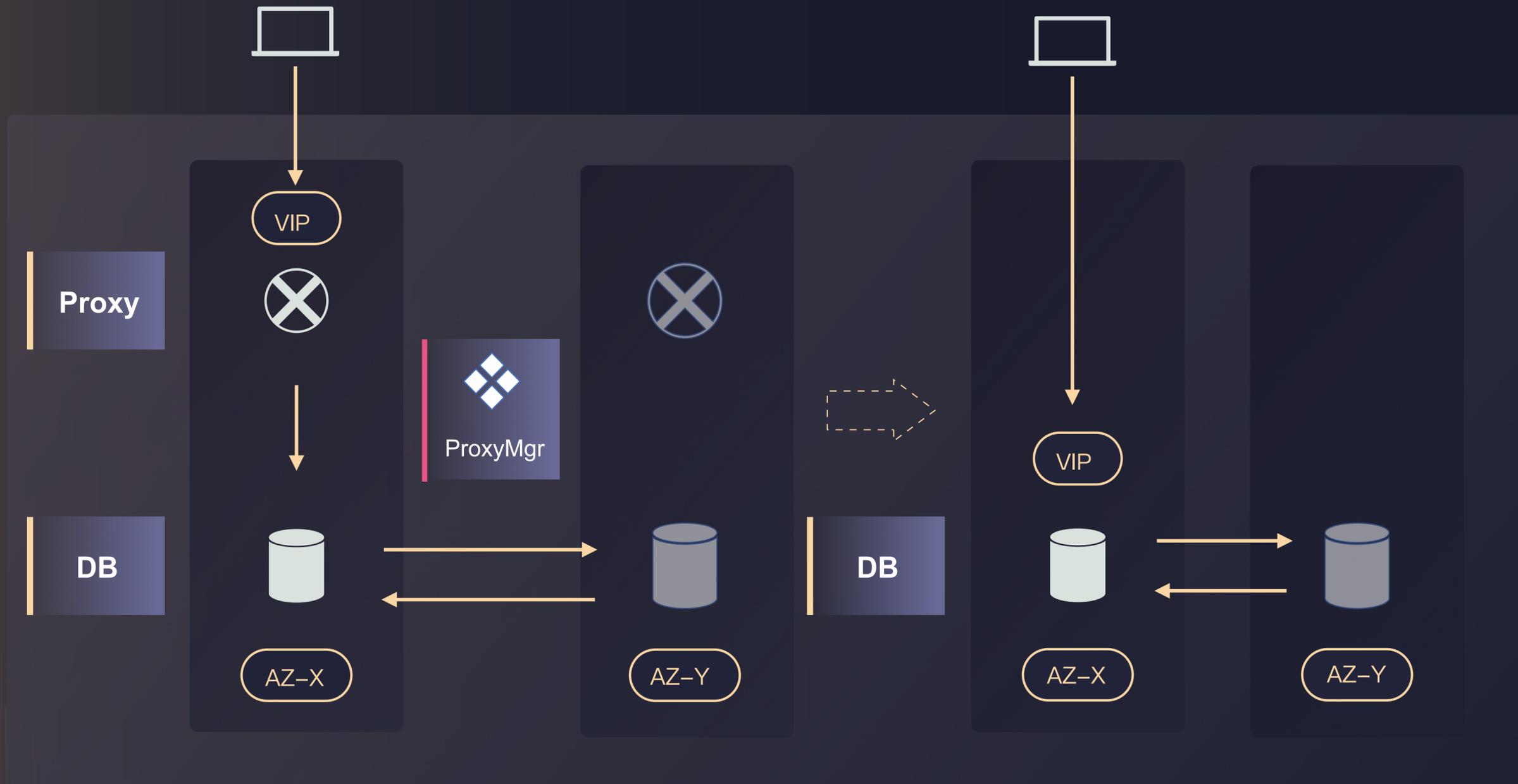
先弹性

再拆分

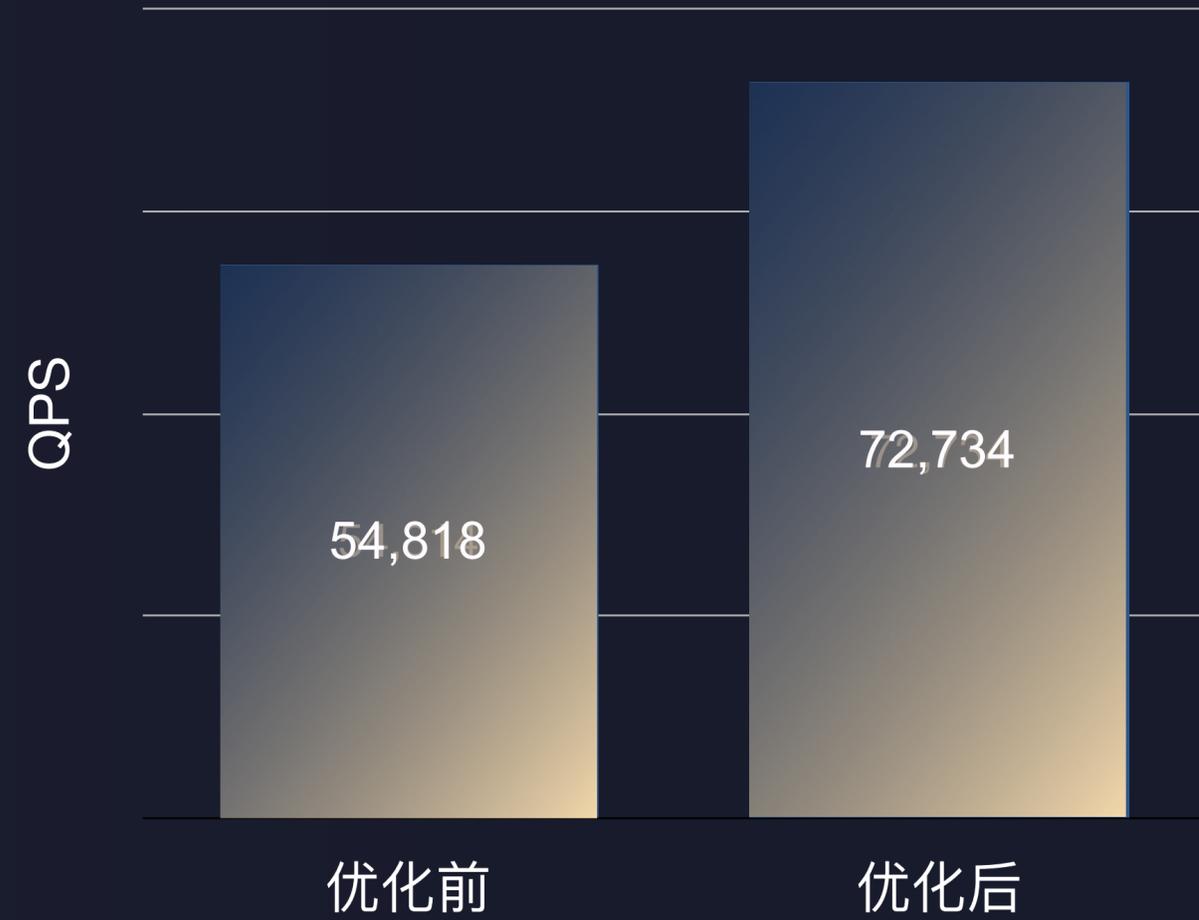
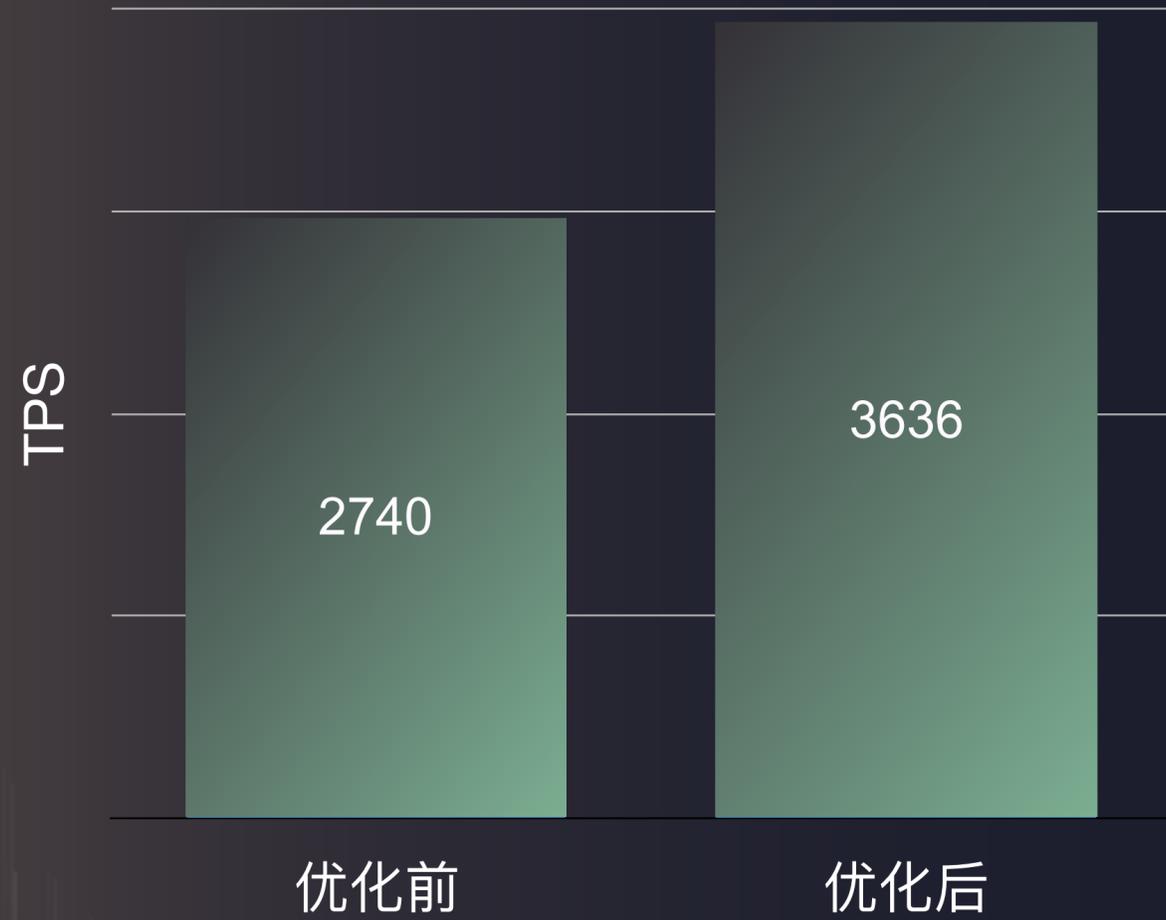
先管理

再精细

UDB解决之道 – 可用性

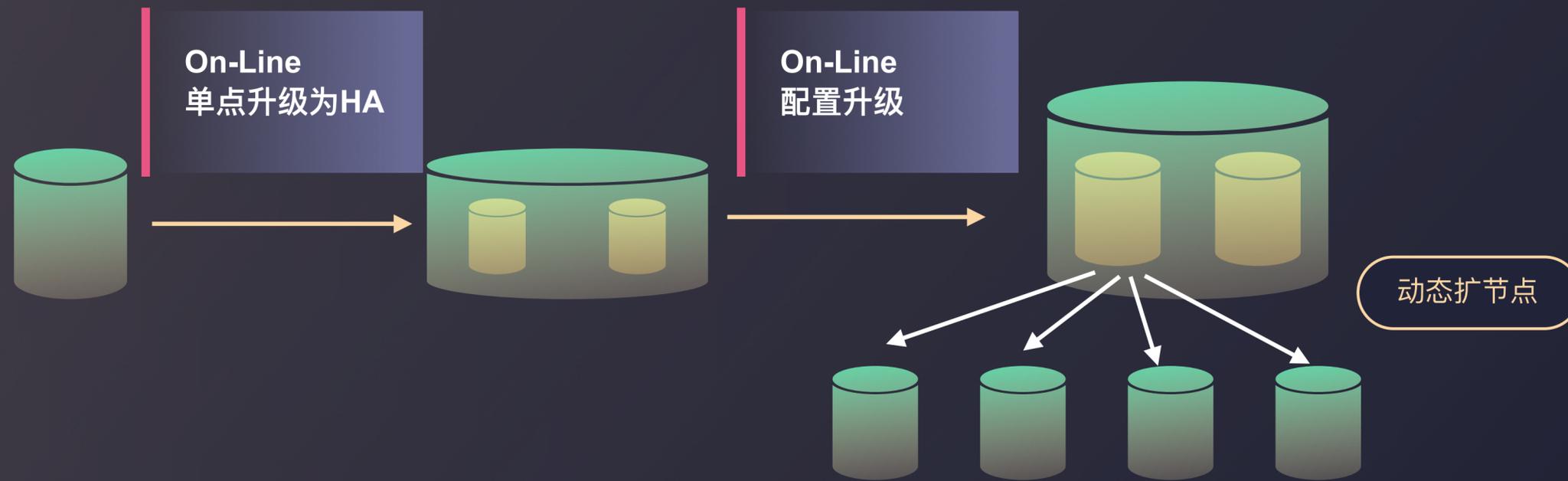


UDB解决之道 – 高性能

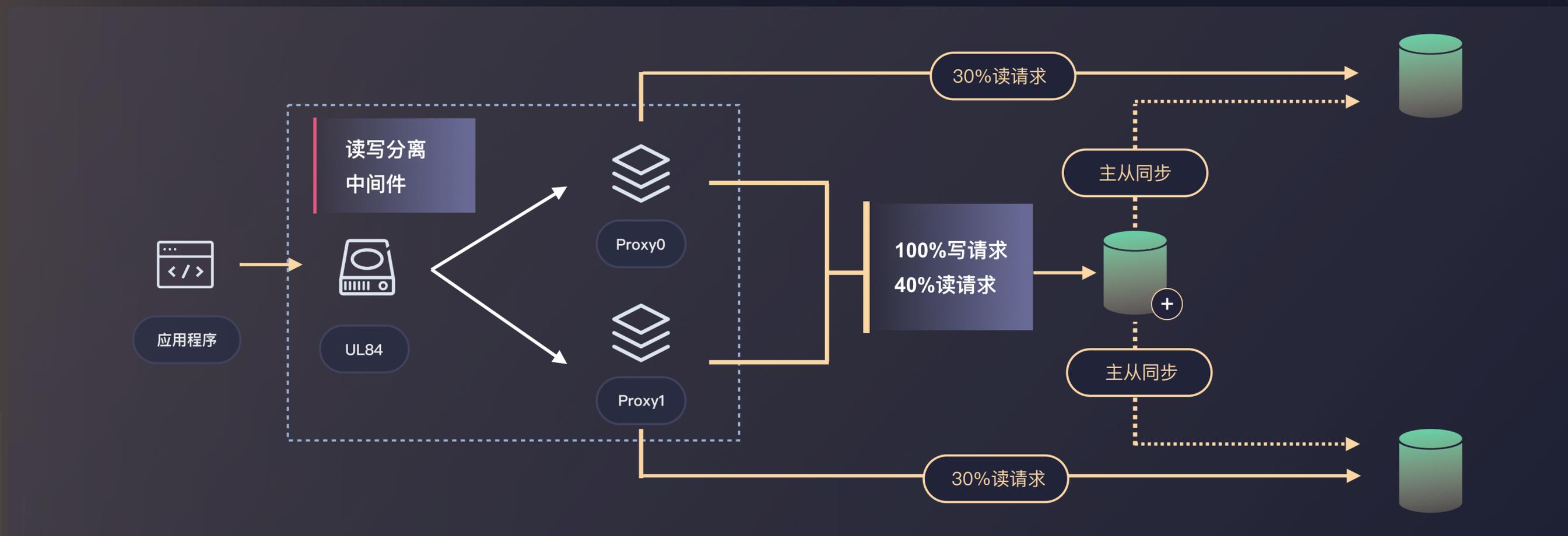


```
/usr/bin/sysbench /usr/share/sysbench/oltp_read_write.lua --db-driver=mysql --mysql-host=$host --mysql-user=ucloudbackup --mysql-password=$password --mysql-db=sysbench --tables=64 --table-size=10000000 --report-interval=10 --threads=100 --time=36000
```

UDB解决之道 – 多维度可扩展

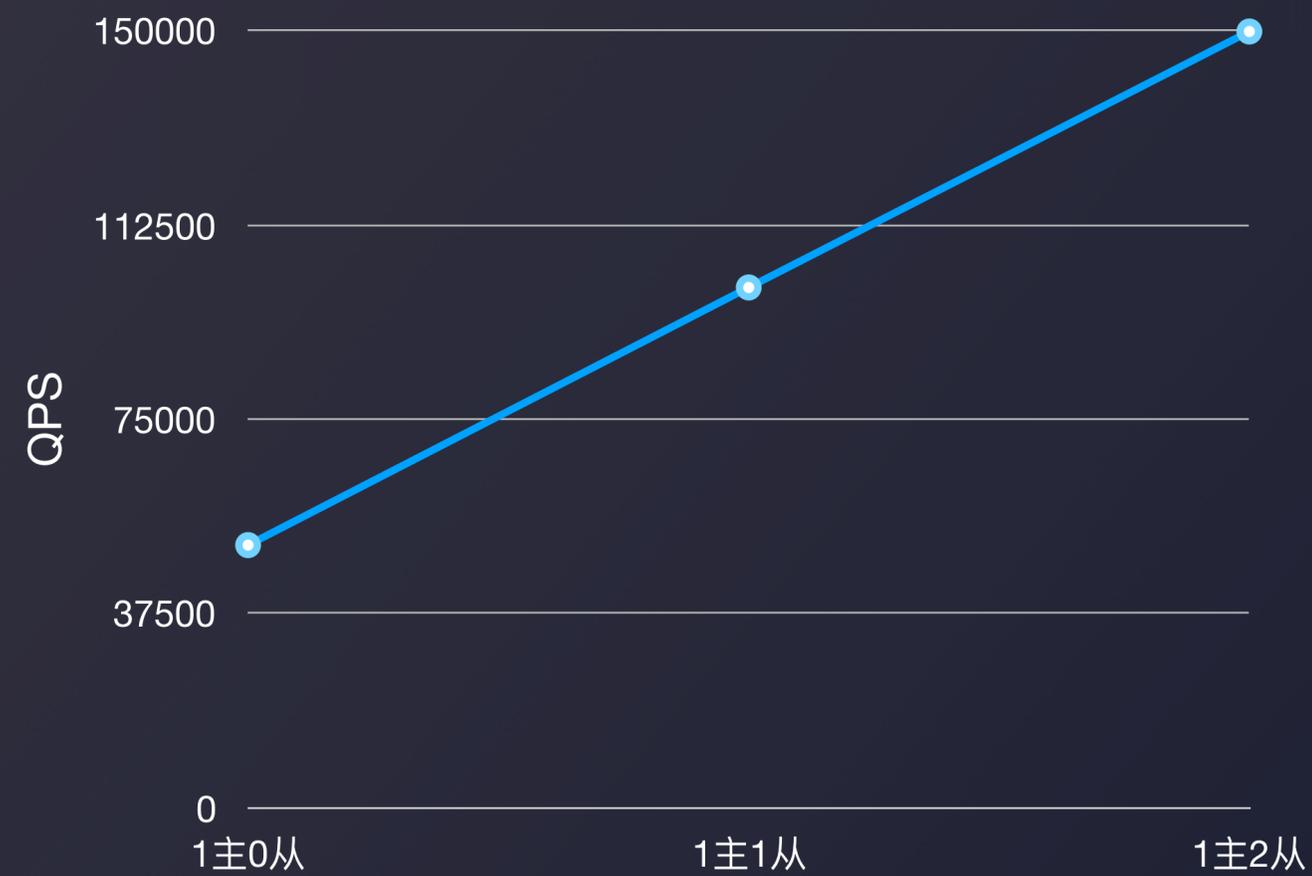
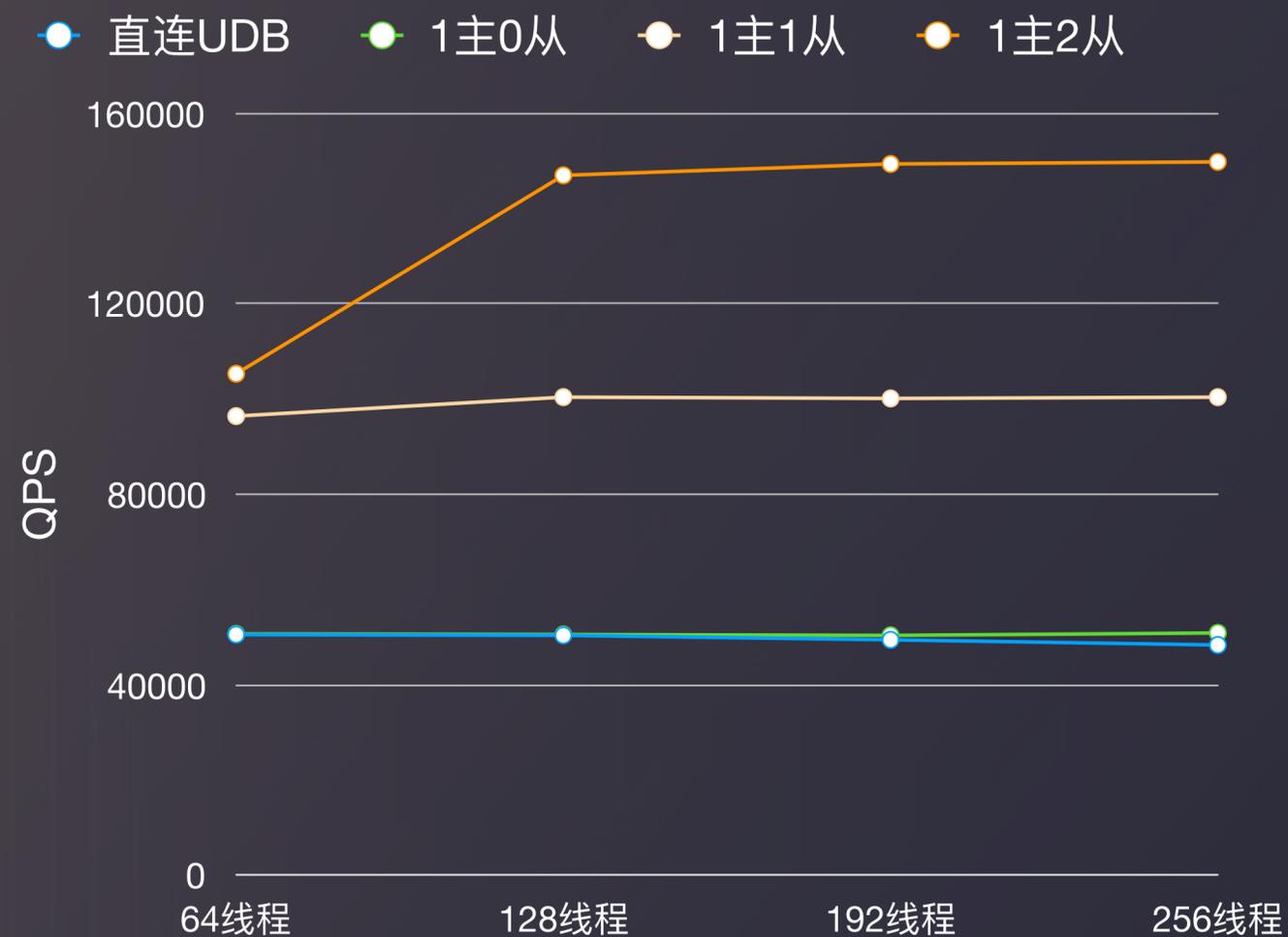


UDB解决之道 – 多维度可扩展

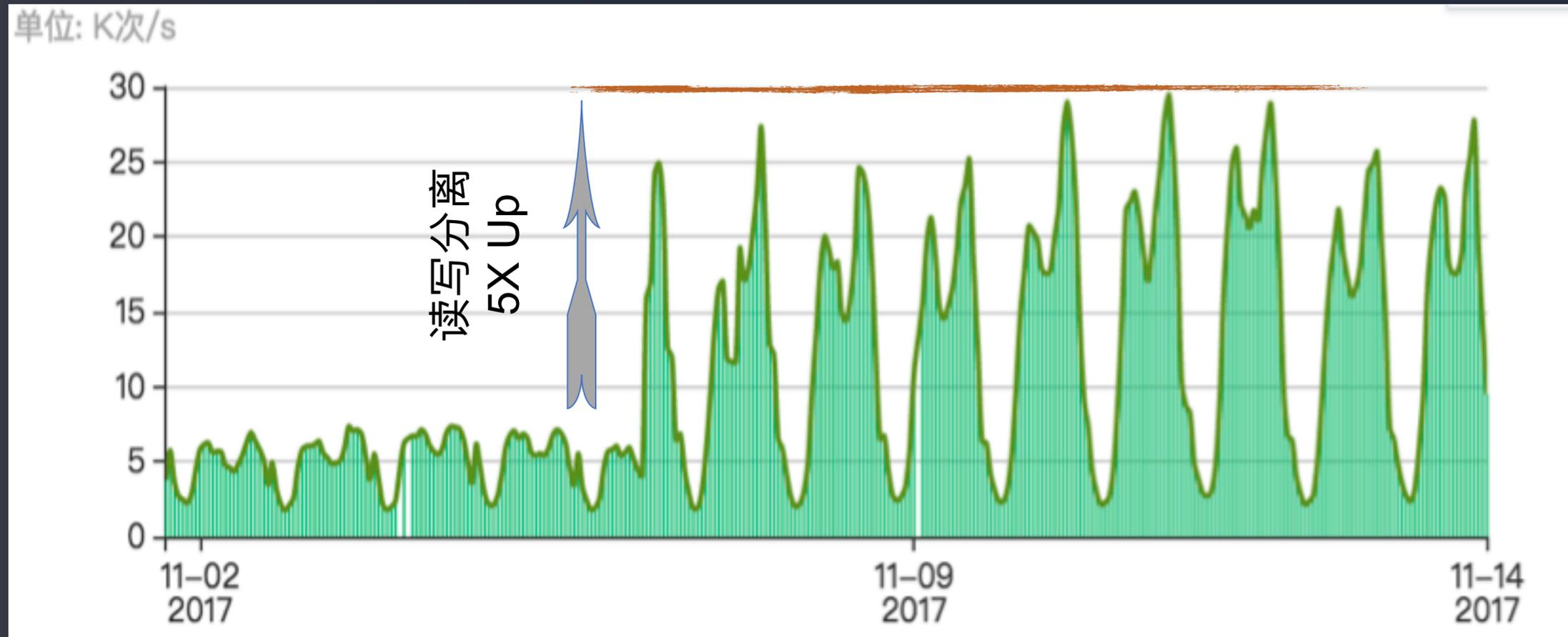


- 自研中间件
- 自定义路由策略
- 完全兼容MySQL协议
- 线性水平扩展
- 100%兼容客户端
- 无性能损耗
- 数据链路无单点故障
- 免费在线开启

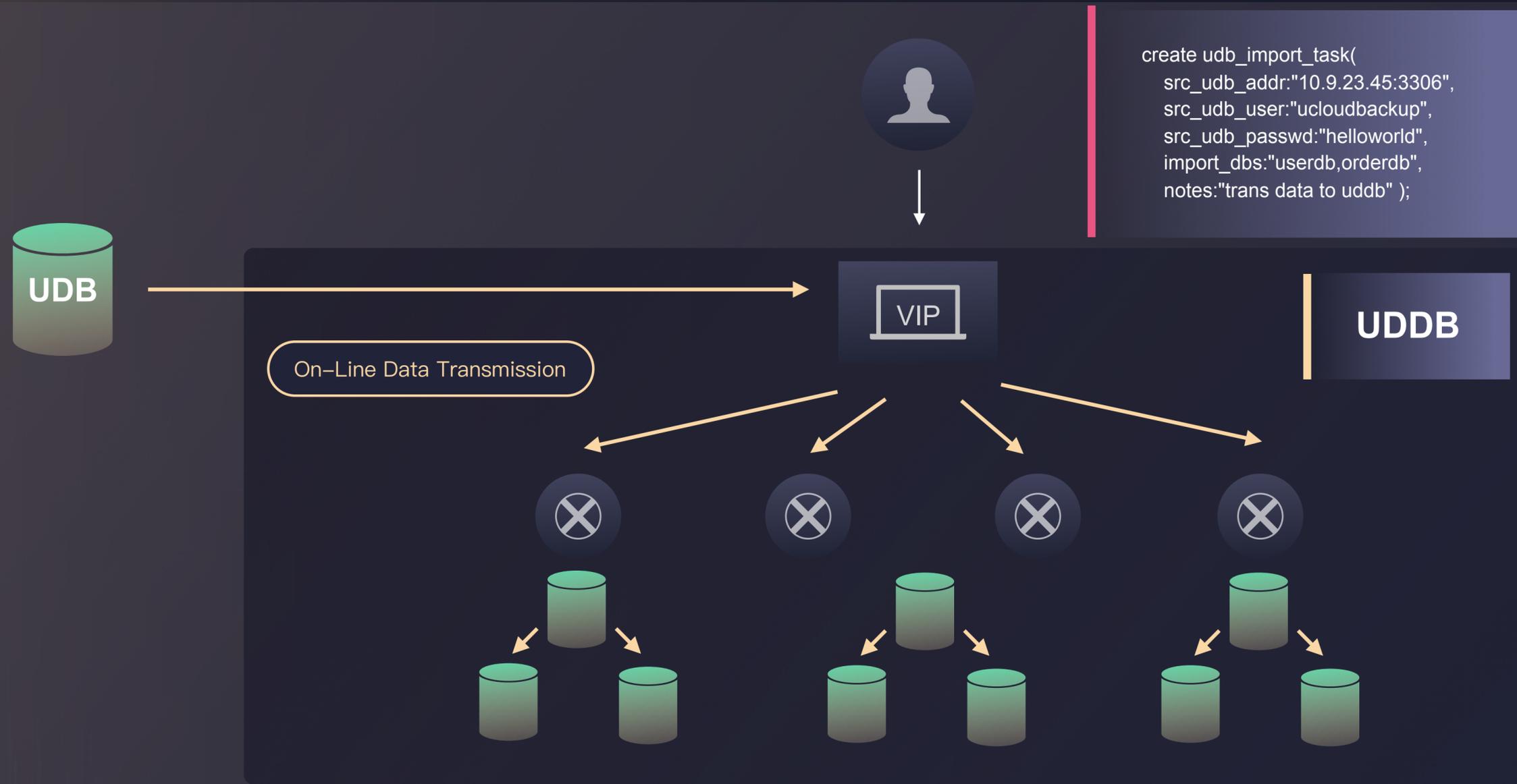
UDB解决之道 - 多维度可扩展



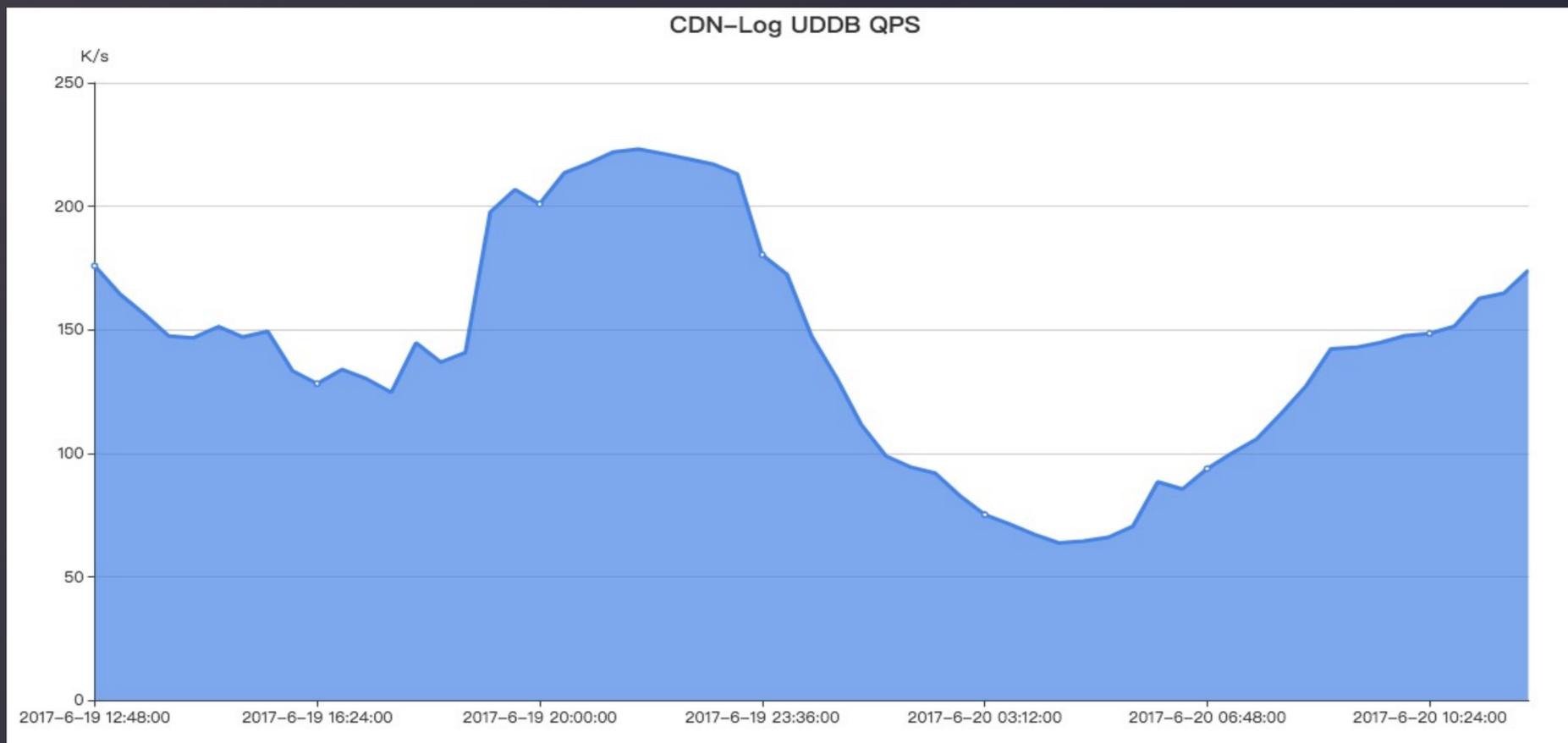
UDB解决之道 – 多维度可扩展



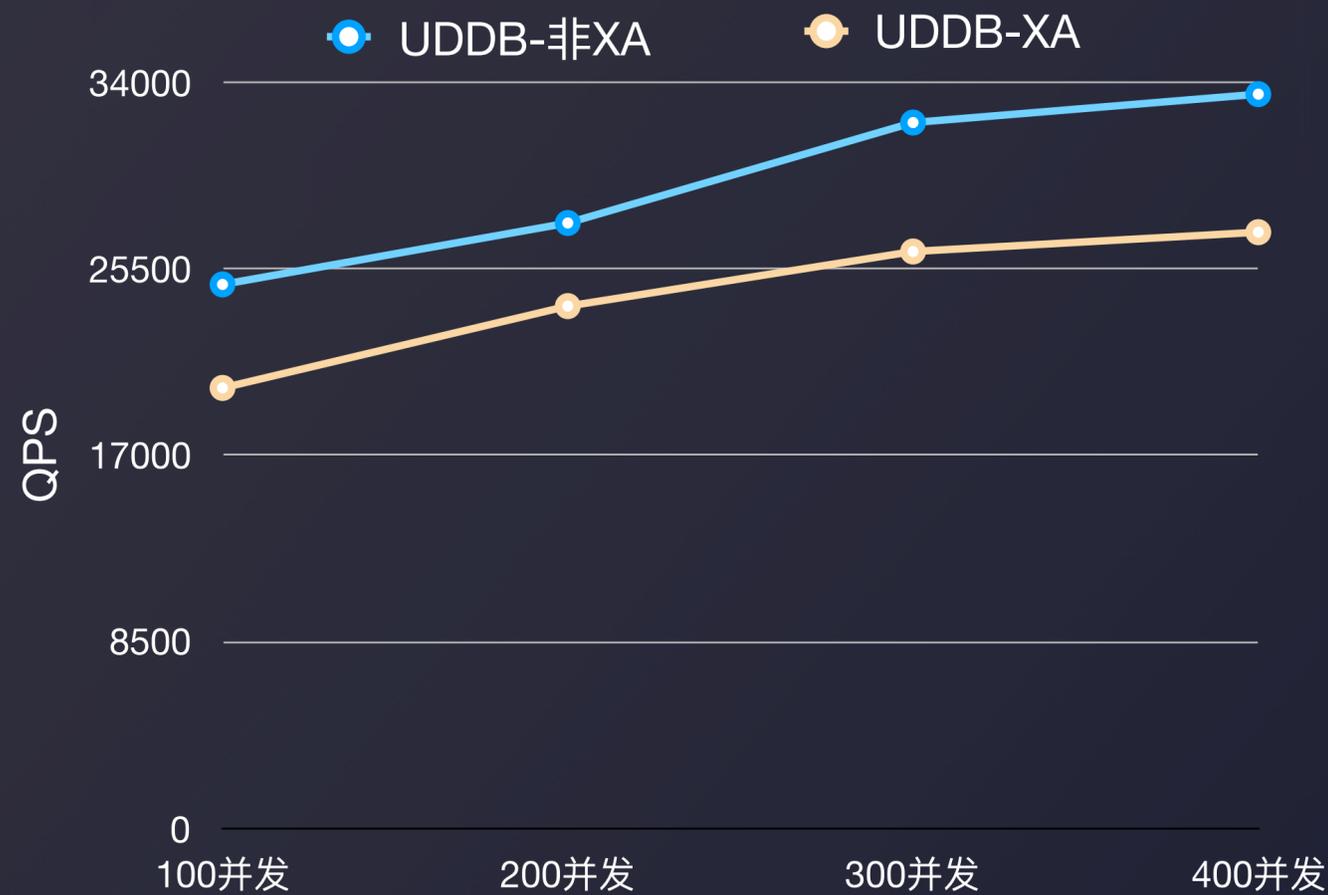
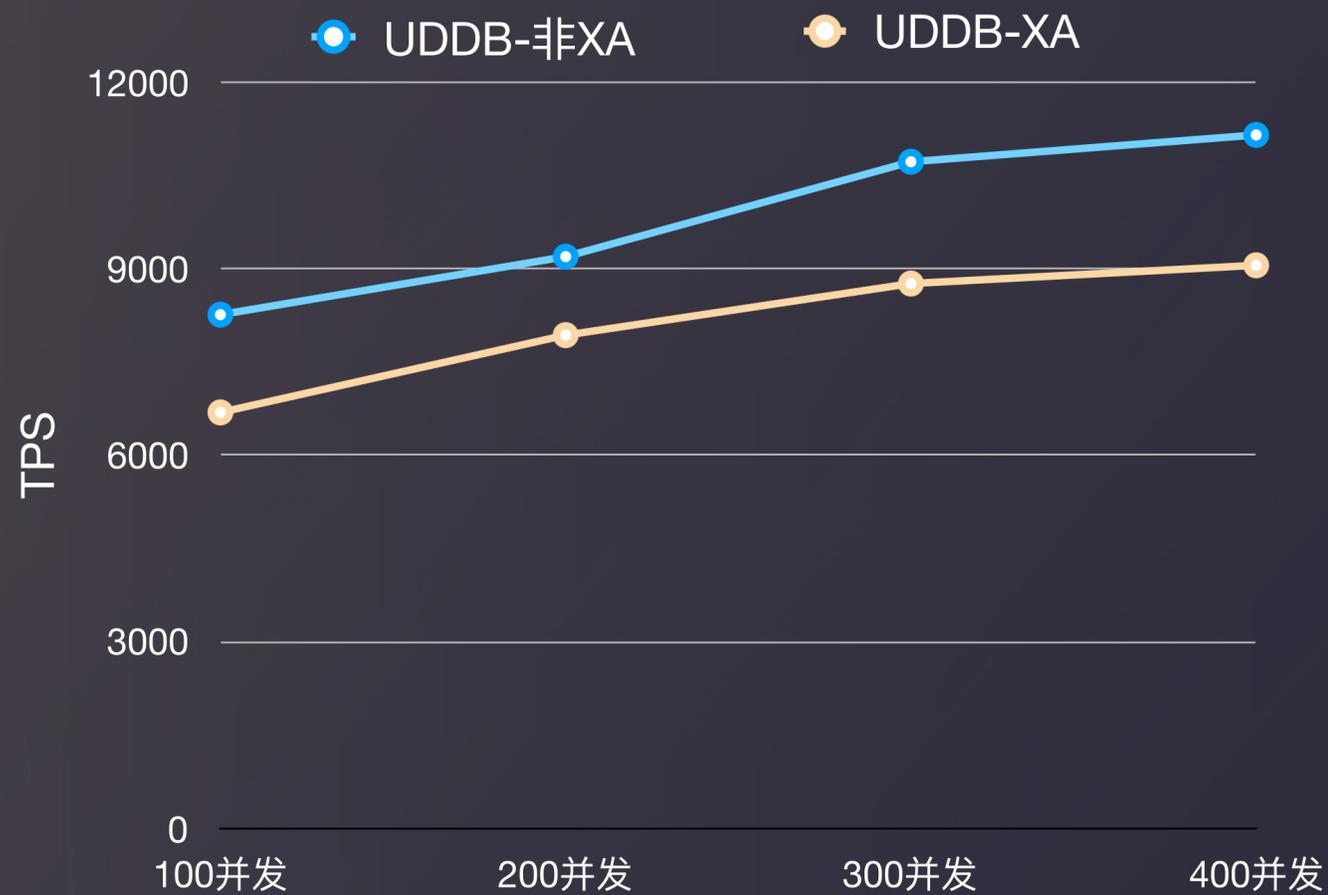
UDB解决之道 – 多维度可扩展



UDB解决之道 – 多维度可扩展



UDB解决之道 – UDDB分布式事务



UDB解决之道 — UDDDB分布式事务

事务级别	说明
MULTI_WRITE (1)	不支持事务，任何下发到MySQL节点的语句都是自动提交。 主要用于数据导入阶段
BROADCAST_TABLE_XA_AT_AUTO_COMMIT (2)	广播表非事务写转XA事务
ALL_TABLE_XA_AT_AUTO_COMMIT (3)	所有表非事务多写转XA事务
ALL_TABLE_XA_AT_NOAUTO_COMMIT (4)	begin转XA事务处理
SELECT_FOR_UPDATE_AT_NOAUTO_COMMIT (5)	下发到MySQL节点的select语句，均带上for update，从而实现强一致性

UDB解决之道 – UDDDB分布式Join

```
select
  t_product.name,
  count(*),
  sum(price)
from
  t_user,
  t_product,
  t_order
where
  t_user.uid=t_order.uid
  and t_product.pid=t_order.pid
group by
  t_product.name
order by
  t_product.name desc;
```

```
mysql> select * from t_order,t_product,t_user where t_user.uid=t_order.uid and t_product.
pid=t_order.pid and t_product.name like '%-%';
+-----+-----+-----+-----+-----+-----+-----+-----+
| uid | pid | create_time | pid | name          | price | uid | name |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1002 | 12 | 2147483647 | 12 | mysql-book    | 58    | 1002 | dean |
| 1002 | 13 | 2147483647 | 13 | train-ticket  | 199   | 1002 | dean |
| 1003 | 12 | 2147483647 | 12 | mysql-book    | 58    | 1003 | yonghui |
+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.07 sec)
```

```
mysql> select t_product.name, count(*), sum(price) from t_order,t_product,t_user where t_
user.uid=t_order.uid and t_product.pid=t_order.pid group by t_product.name order by t_pro
duct.name desc;
+-----+-----+-----+
| name          | count( *) | sum( t_product.price) |
+-----+-----+-----+
| train-ticket  | 1         | 199                    |
| mysql-book    | 2         | 116                    |
| mbp           | 1         | 7999                   |
| iphone7      | 1         | 5588                   |
+-----+-----+-----+
4 rows in set (0.07 sec)
```

UDB赋能案例解析

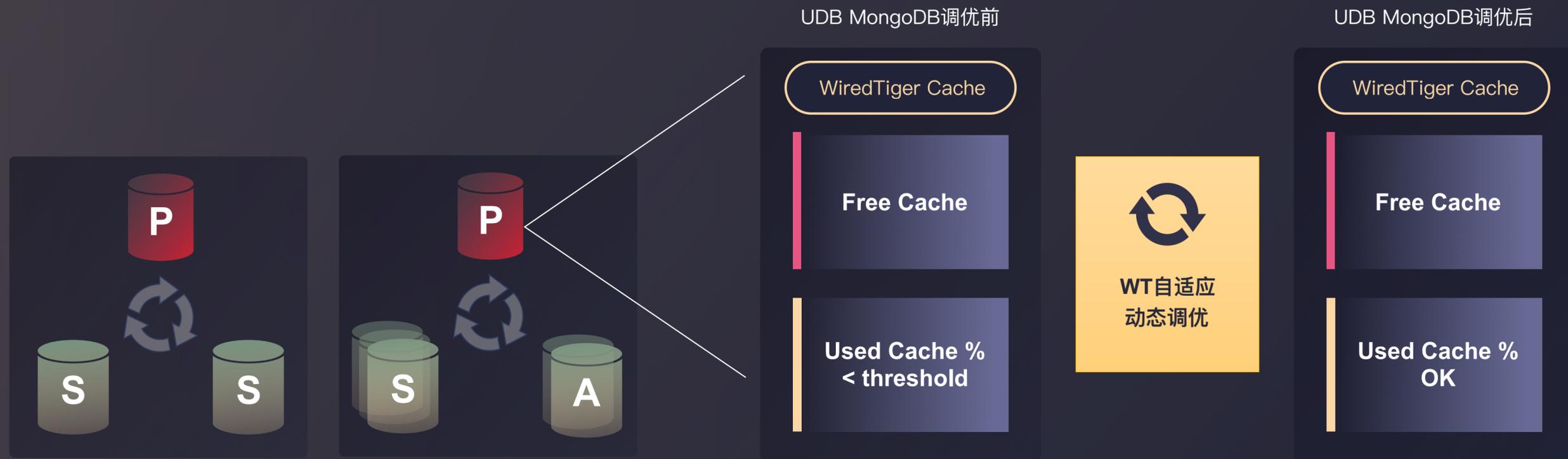
 All-in-One: 自适应性能调优

 618/双11: 差异化保障与持续可用

 抢红包: 多维度弹性可扩展

2019 TIO

All-in-One: 自适应性能调优



● 静态变动态

● 按量自适应调整

● 维持合理使用率

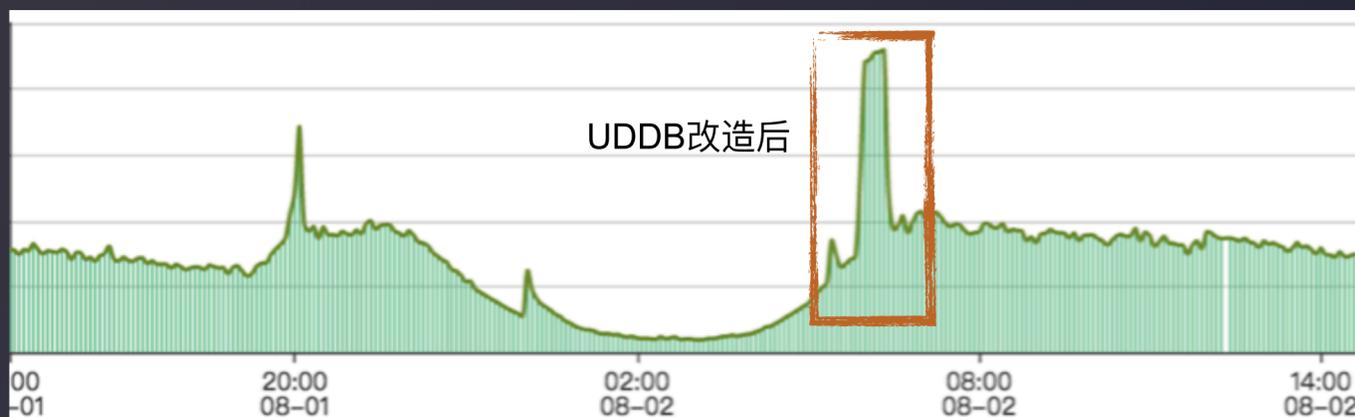
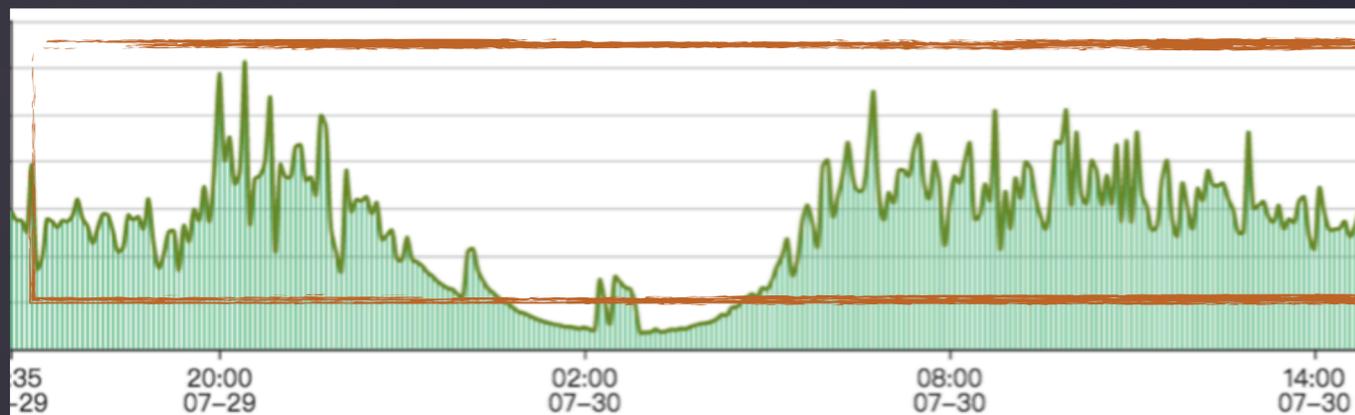
● 抗业务高峰



抢红包：多维度弹性可扩展

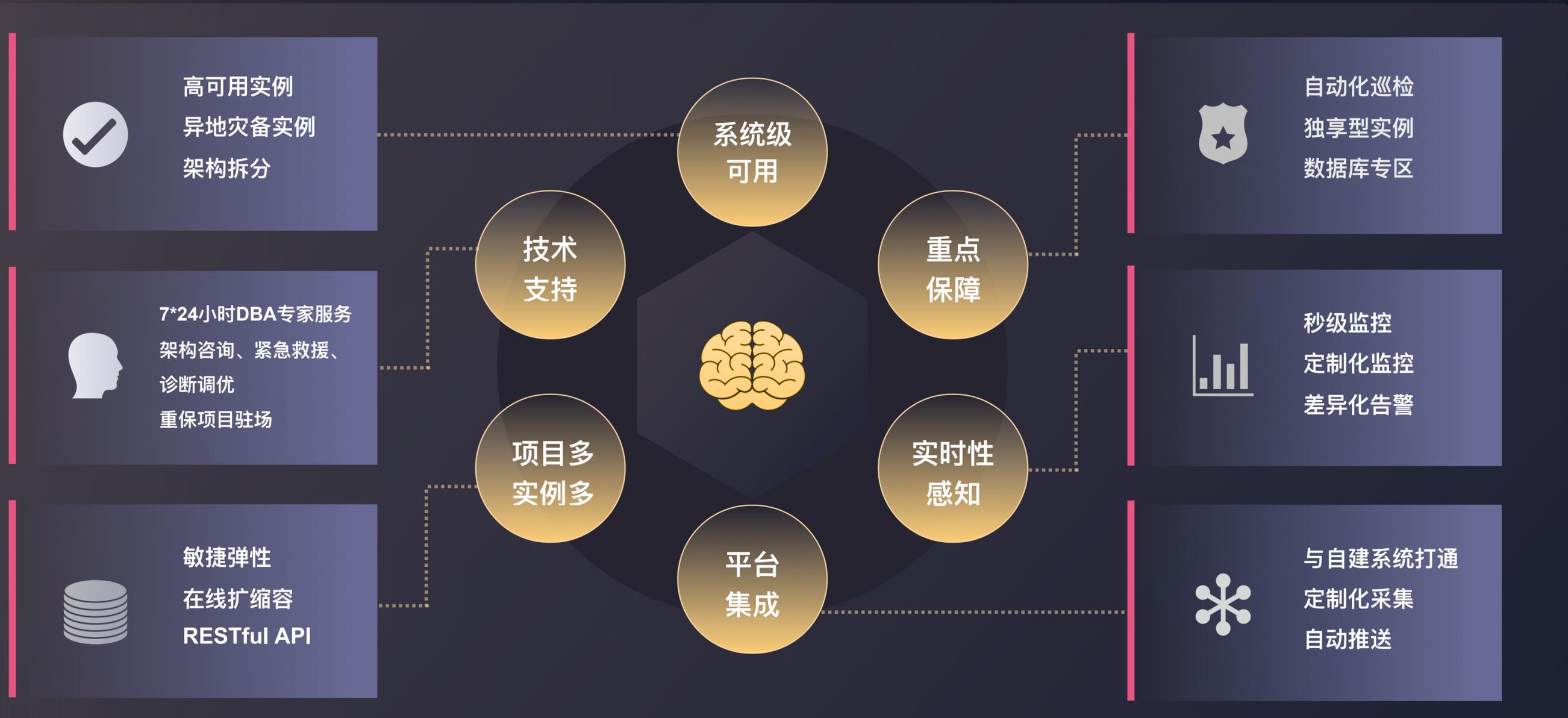
- 高并发
- 波动大、有毛刺
- 业务上线慢

- 高性能抗业务高峰
- 去毛刺
- 业务敏捷上线



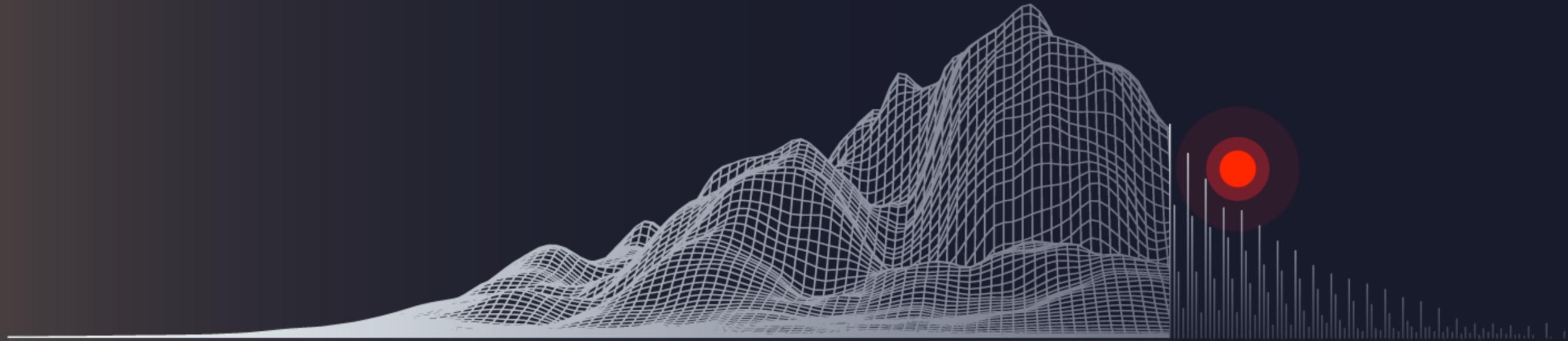


618/双11: 差异化保障与持续可用



618/双11: 差异化保障与持续可用





THANKS