



ZHDBA.COM  
中华数据库行业协会

# TDSQL PROXY: 金融级数据库中间件

郑寒 @腾讯 计费平台部

# 目录

## CONTENTS

01

背景

02

分区方案

03

兼容性

04

分布式事务

05

丰富的功能

# 01

## 背景

单机数据库满足不了海量的业务数据

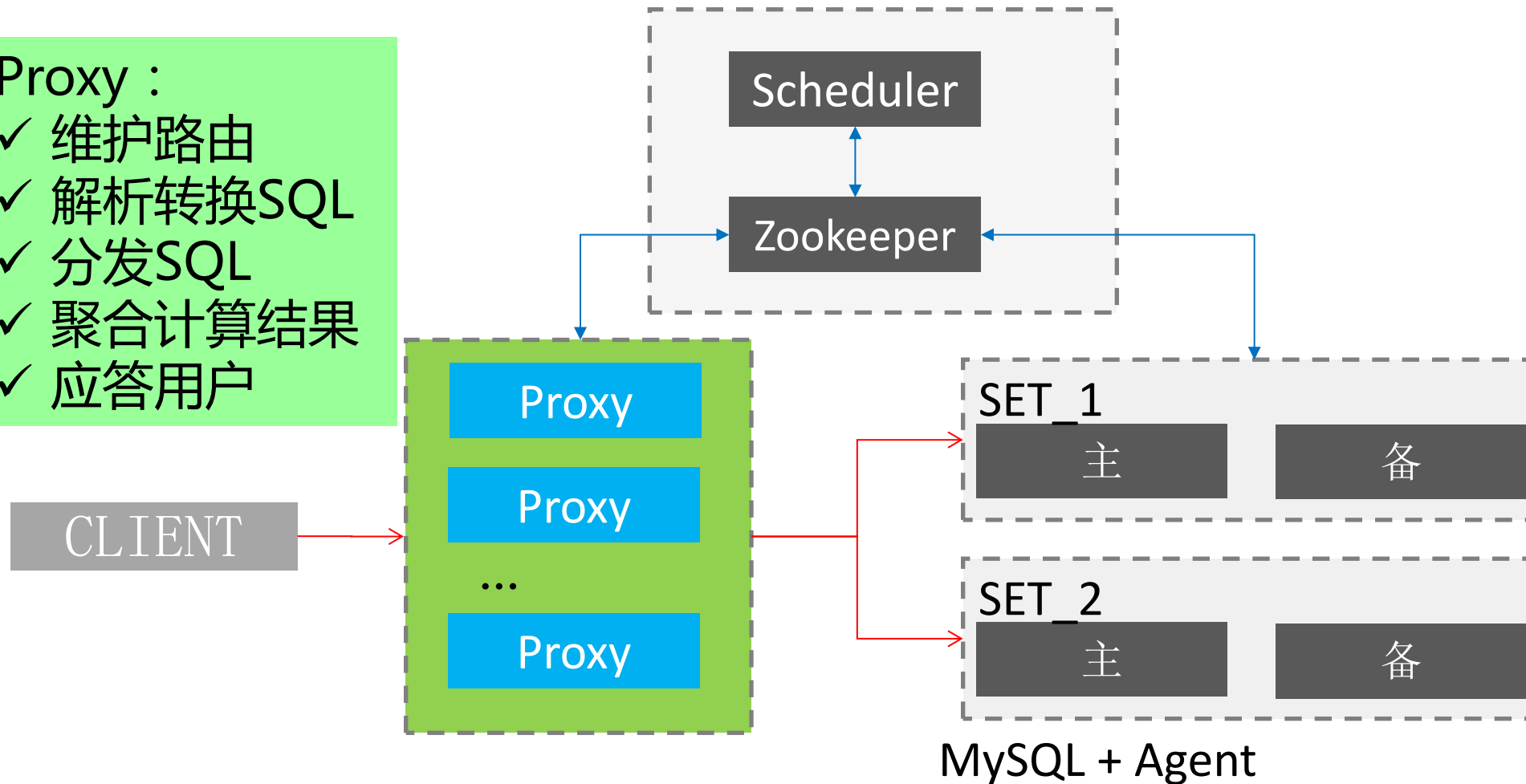
解决方案:

- ✓业务分库分表
- ✓NOSQL
- ✓NEWSQL
- ✓中间件

# TDSQL总体架构

Proxy :

- ✓ 维护路由
- ✓ 解析转换SQL
- ✓ 分发SQL
- ✓ 聚合计算结果
- ✓ 应答用户



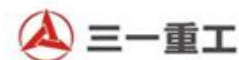
# 业务场景

TDSQL是针对金融联机交易场景的高一致性数据库集群

## • 专有云



## • 腾讯云



# 02

## 分区方案

# 分区方案

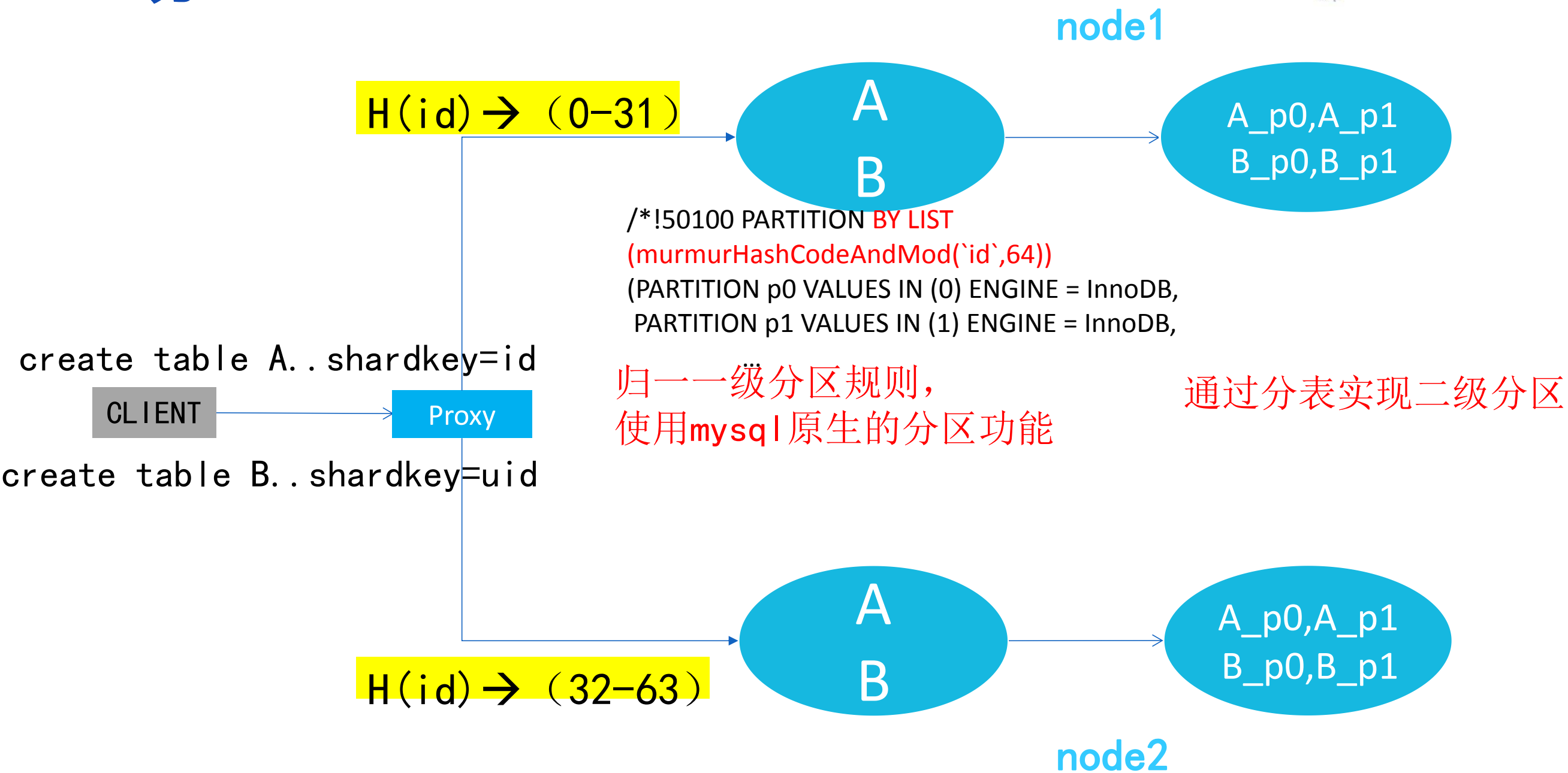
- 均衡数据分布和访问
- 快速扩容
- 相关表的JOIN
- 快速删除流水数据



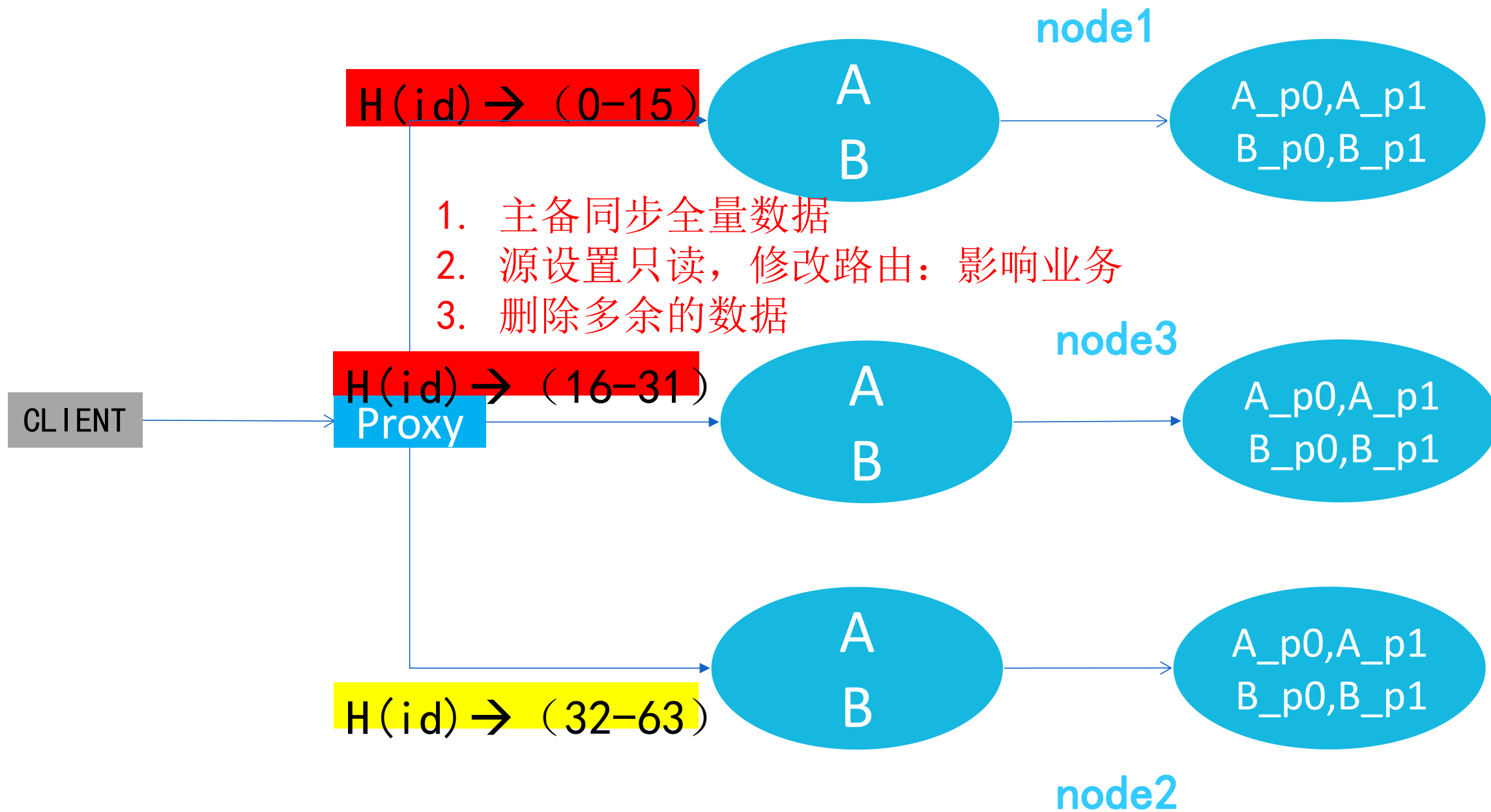
- ✓ 自定义HASH函数+LIST
- ✓ Groupshard+Global表
- ✓ 二级分区



# 分区

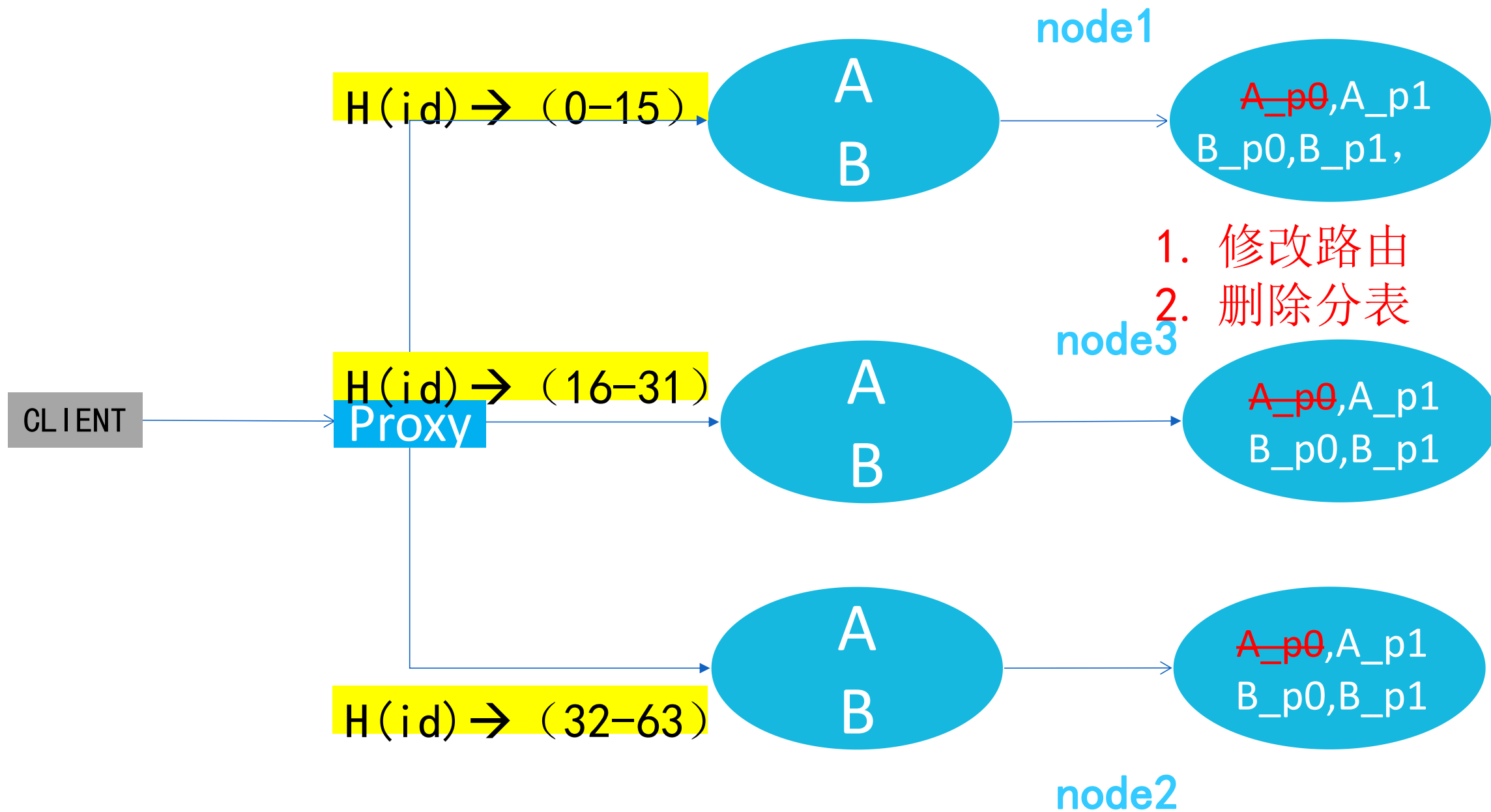


# 扩容



1. 主备同步全量数据
2. 源设置只读，修改路由：影响业务
3. 删除多余的数据

# 增删数据

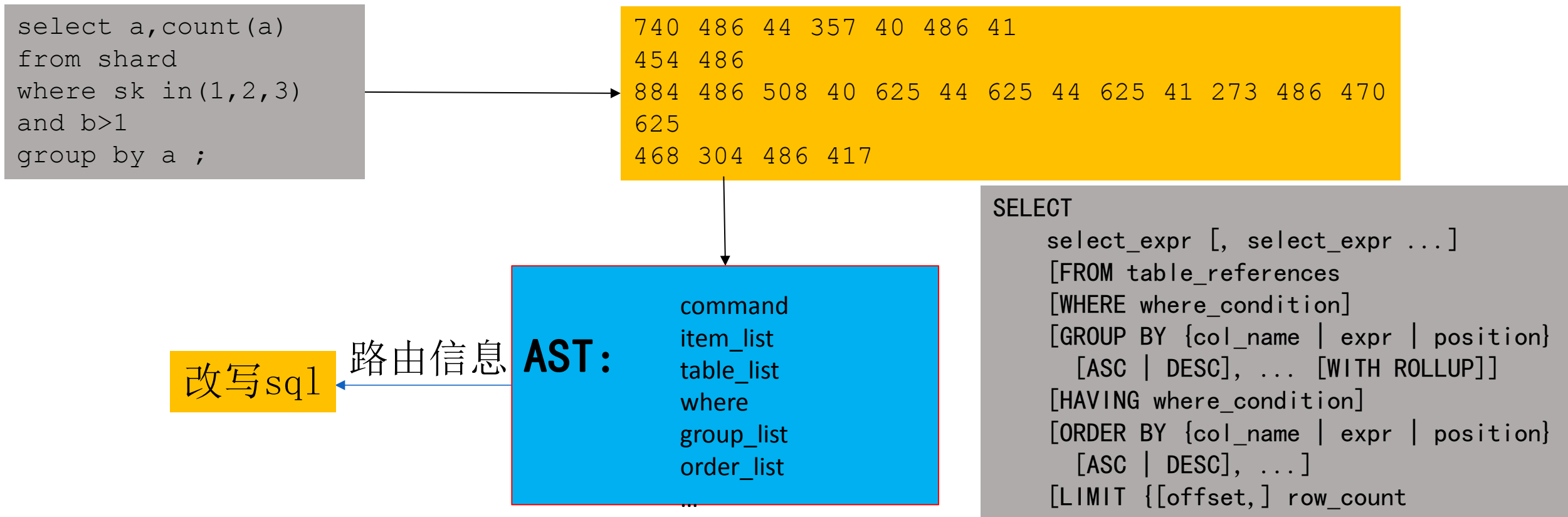


# 03

## 兼容性

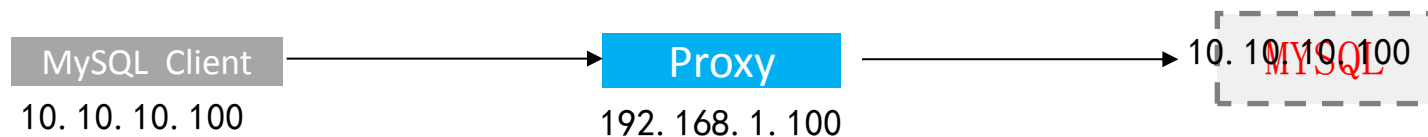
# 语法

- 基于mysql语法文件
  - ✓ 在proxy层拒绝有问题的sql
  - ✓ 增加定制化的规制
  - ✓ 基于语法树改写sql



# 协议

- 连接协议兼容MySQL
  - ✓ 客户端不需修改连接流程
- 支持预编译，复制协议
  - ✓ Proxy实现文本协议和二进制协议的转换
- 权限
  - ✓ 支持IP白名单
  - ✓ 通过IP透传提供和mysql一样的权限控制



```
4      capability flags, CLIENT_PROTOCOL_41 always set
4      max-packet size
1      character set
1      string[23]  reserved (all 101)
string[NUL]  username
if capabilities & CLIENT_PLUGIN_AUTH_LENENC_CLIENT_DATA {
length-int  length of auth-response
string[n]   auth-response
} else if capabilities & CLIENT_SECURE_CONNECTION {
1          length of auth-response
```

# 功能

- 聚合函数: sum, max, min, count, avg, distinct, group by, order by, limit
- 全局唯一序列
  - ✓ 基于zookeeper实现分布式唯一
- 子表
  - ✓ 改写结果集, 用户透明
- JOIN
  - ✓ shardkey相同: 各个set内单独聚合, proxy层汇总
  - ✓ global表

```
select A. a, avg(A. b)
from A join B on A. sk=B. sk
where A. hired=20160203
group by A. a
```

sql改写

```
select A. a, avg(A. b), sum(A. b), count(A. b)
from ((select * from test. A_TDSQL_SUBp1) A
join (select * from test. B_TDSQL_SUBp1) B
on((A. sk = B. sk)))
group by A. a order by A. a
```

结果集聚合

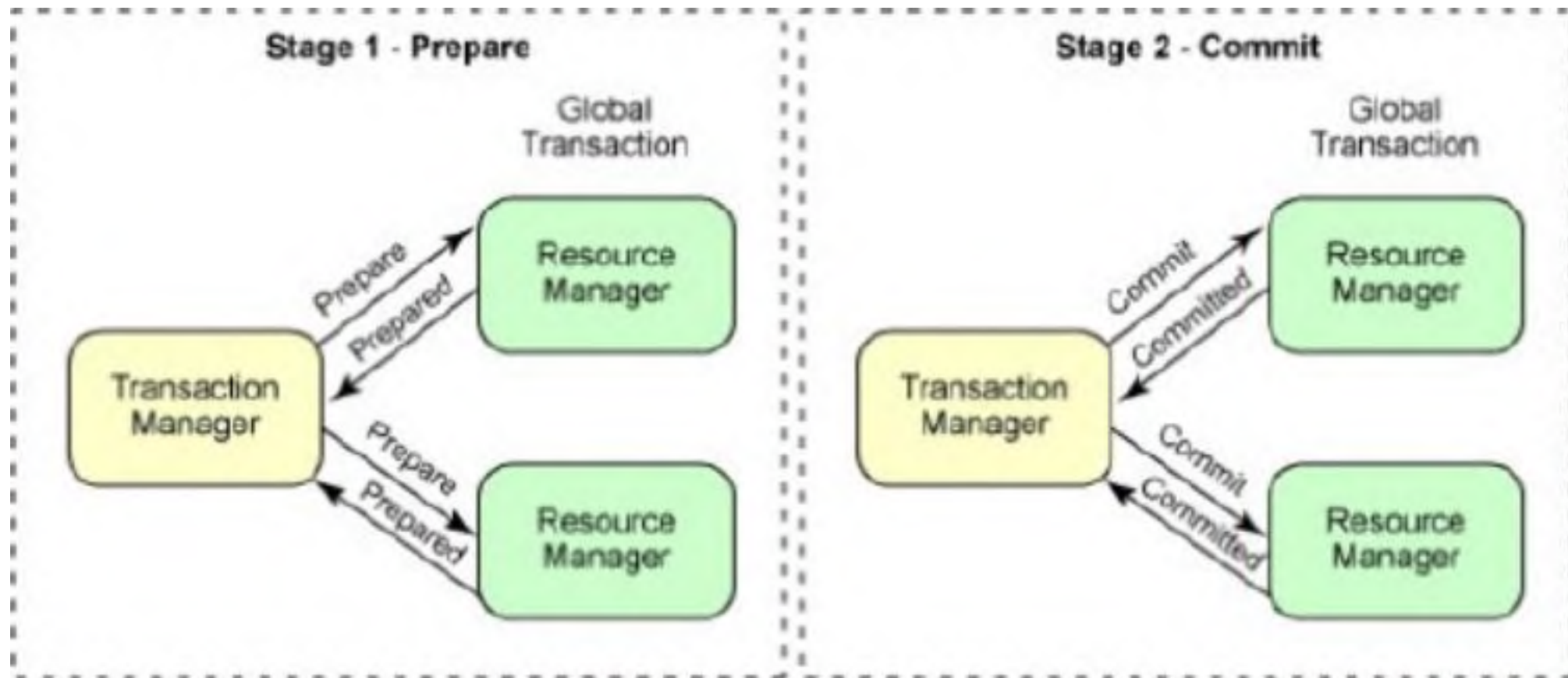
# 04

## 分布式事务



# 原理：两阶段提交

- Proxy作为TM，后端数据库作为RM
- 用数据库存储日志用于故障恢复



# 例子：

client	set_1	set_2
begin;		
insert into t1 values(1,2000);	XA START 'xa-gtid-1';	
	insert into t1 values(1,2000);	
insert into t1 values(3,4000);		XA START 'xa-gtid-1'
		insert into t1 values(3,4000);
commit;	xa end 'xa-gtid-1'; xa prepare 'xa-gtid-1';	
		xa end 'xa-gtid-1'; xa prepare 'xa-gtid-1';
	<b>insert into xa.commit_log values('xa-gtid-1');</b>	
	xa commit 'xa-gtid-1';	
		xa commit 'xa-gtid-1';

# 特性

- 实现了分布式事务两阶段提交的全部的功能和错误处理
  - 使用层面对用户透明
  - 单set的操作自动退化为单机事务
  - 存储日志流程异步化
- 
- 无法保证MVCC select快照一致性
  - 最终一致性

# 05

## 丰富的功能

# 丰富的功能

- 灵活的读写分离
  - ✓ 多种模式：hint, sql解析, 只读账号
- 动态升级
  - ✓ 核心功能so话, 升级不影响已有连接
- 流控
  - ✓ 根据后端状态实时控制sql速度
- 控制指令
  - ✓ 状态查询, 配置修改
  - ✓ sql透传
- 断连优化
  - ✓ 后端断连尽量不影响用户
  - ✓ 识别是否在事务中



ZHDBA.COM  
中华数据库行业协会



# Q&A

