Atitit 提升稳定性 数据库死锁

目录

[1.1. 配置数据库死锁检测超时时间从默认50s到10s 1](#_Toc2339)

[1.2. 调整隔离级别到read commit 1](#_Toc32667)

[1.3. mysql数据库连接使用完毕立即释放掉。不在mysql上使用连接池 1](#_Toc24252)

[1.4. 每个连接增加gc检测，超时强制销毁退出。 1](#_Toc26025)

[1.5. 分库，分为100个库，自然提升稳定性， 2](#_Toc24606)

[1.6. 使用小事务机制 只针对必要业务上大事务 2](#_Toc3707)

[2. 常见死锁问题原因 2](#_Toc2018)

[2.1. 行锁变表锁导致死锁问题 2](#_Toc13606)

[2.2. Ab ba死锁问题 2](#_Toc11295)

## 配置数据库死锁检测超时时间从默认50s到10s

## 调整隔离级别到read commit

加快事务进程 与简单化 。。Oracle 和mssql的隔离级别就是这个read commit，业务准确性不会有问题

## mysql数据库连接使用完毕立即释放掉。不在mysql上使用连接池

方法，因为mysql数据库对行锁问题，很容易升级为表锁。所以不建议使用数据库连接池，因为会造成连接迟迟得不到释放，导致事务一直进行中。。mysql数据库连接使用完毕立即释放掉。。

线程池提升的性能可以增加机器负载均衡解决即可。。稳定性优先与性能

## 每个连接增加gc检测，超时强制销毁退出。

。。相当于超时熔断器，一般一个事务执行7秒足够了

## 分库，分为100个库，自然提升稳定性，

锁表死锁概率自然减少到了百分之一

## 使用小事务机制 只针对必要业务上大事务

# 常见死锁问题原因

## 行锁变表锁导致死锁问题

START TRANSACTION;

update user set bals=111 where id=1;

commit;

如果此事务中途等待或长时间不提交，那么会锁表 锁住整个表

造成其他用户也不能修改自己的数据

START TRANSACTION;

update user set bals=22 where id=2;

commit;

## Ab ba死锁问题

增加超时检测 和 及时销毁连接