

解析Oracle多粒度锁的验证步骤 PDF转换可能丢失图片或格式
， 建议阅读原文

https://www.100test.com/kao_ti2020/473/2021_2022__E8_A7_A3_E6_9E_90Orac_c67_473542.htm 创建测试表

aaa@AAA.US.ORACLE.COM>create table test (a number , b number) ; 表已创建。 1 对于未提交的insert操作

aaa@AAA.US.ORACLE.COM>insert into test values (10 , 20) ; 已创建 1 行。 SQL> 0select * from v\$sqllock. ADDR KADDR SID TYPE ID1 D2 LMODE REQUEST CTIME BLOCK 12AB9394 12AB93A8 17 TM 29512 0 3 0 84 0 12AF9218 12AF9324 17 TX

589841 5506 6 0 84 0 可见，对于未提交的insert操作，会产生两个锁，其类型（TYPE）分别为TM和TX，也就是表级意向锁和事务锁。表级意向锁的模式（LMODE）为：3，表示

是row exclusive，即表示此表中的某行获得了行排他锁。事务锁的模式（LMODE）为：6，表示是exclusive，即排他锁，表示此事务获得了排他锁。BLOCK表示此锁是否阻塞了其它的锁，即发生死锁；此处没有。

2 对于提交的insert操作
aaa@AAA.US.ORACLE.COM>commit ; 提交完成。 SQL>

0select * from v\$sqllock. ADDR KADDR SID TYPE ID1 D2 LMODE REQUEST CTIME BLOCK 此处已没有记录，说明在提交后，即完成了锁的释放。

3 对于未提交的0update操作
aaa@AAA.US.ORACLE.COM>0update test set a=11 where a=10 ; 已更新 1 行。 SQL> 0select * from v\$sqllock. ADDR KADDR SID

TYPE ID1 D2 LMODE REQUEST CTIME BLOCK 12AB9394 12AB93A8 17 TM 29512 0 3 0 3 0 12AF9218 12AF9324 17 TX

262153 5590 6 0 3 0 可见0update操作所引起的锁的信息完全等

同于insert操作..... 4 对于提交的update操作

aaa@AAA.US.ORACLE.COM>commit ; 提交完成。SQL>

select * from v\$lock. ADDR KADDR SID TYPE ID1 D2 LMODE
REQUEST CTIME BLOCK 此处已没有记录，说明在提交后，
即完成了锁的释放。 5 对于select操作

aaa@AAA.US.ORACLE.COM>select * from test where a=11. A B

----- 11 20 SQL> select * from v\$lock. ADDR
KADDR SID TYPE ID1 D2 LMODE REQUEST CTIME BLOCK

此处已没有记录，说明select操作不会引起任何锁。这是
与sql server等数据库不同的，这些数据库select操作也会引起
锁，以取得一致读；而oracle是通过回滚机制实现一致读的，
所以不需要引入锁机制，这极大增强了oracle的并发度。

100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com