

Lawson's Blog

好记性不如烂笔头！

博客园 首页 新随笔 联系 订阅 管理

随笔 - 71 文章 - 0 评论 - 97

Mysql查询语句使用select.. for update导致的数据库死锁分析

近期有一个业务需求，多台机器需要同时从Mysql一个表里查询数据并做后续业务逻辑，为了防止多台机器同时拿到一样的数据，每台机器需要在获取时锁住获取数据的数据段，保证多台机器不拿到相同的数据。

我们Mysql的存储引擎是innodb，支持行锁。解决同时拿数据的方法有很多，为了更加简单，不增加其他表和服务的情况下，我们考虑采用select... for update的方式，这样X锁锁住查询的数据段，表里其他数据没有锁，其他业务逻辑还是可以操作。

这样一台服务器比如select .. for update limit 0,30时，其他服务器执行同样sql语句会自动等待释放锁，等待前一服务器锁释放后，该台服务器就能查询下一个30条数据。如果要求更智能，oracle支持for update skip locked跳过锁区域，这样能不等待马上查询没有被锁住的下一个30条记录。

下面说下mysql for update导致的死锁。

经过分析，mysql的innodb存储引擎实为锁虽然是锁行，但它内部是锁索引的，根据where条件和select的值是否有主键或非主键索引来判断怎么锁，比如只有主键，则锁主键索引，如果只有非主键，则锁非主键索引，如果主键非主键都有，则内部会按照顺序锁。但同样的select .. for update语句怎么就死锁了呢？同样的sql语句查询条件和结果顺序都一致，按理不会导致一个锁了主键索引，等待锁非主键索引，另外一个锁了非主键索引，等待主键索引导致的死锁。

最后经过分析，我们项目里发现是for update的sql语句，和另外一个update非select数据的sql语句导致的死锁。

比如有60条数据，select .. for update查询第31-60条数据，update在更新1-10条数据，按照innodb存储引擎的行锁原理，应该不会导致不同行的锁导致的互相等待。开始以为是行锁在数据量较大情况下，会锁数据块。导致一个段的数据被锁住，但经过大量数据测试，发现感觉把整个表都锁住了，但实际不是。

下面举几个例子说明：

数据从id =400000的数据开始，IsSuccess和GetTime字段都为0，现在如果400000数据的IsSuccess为1了。执行下面两条sql。

```
1  -- 1:
2  set autocommit=0;
3  begin;
4  select * from table1 where getTime < 1 and IsSuccess=0 order by id asc limit 0,30 for update;
5  commit;
6  -- 2:
7  update table1 a set IsSuccess=0 where id =400000;
```

第一条sql语句先不commit，则第二条sql语句将只能等待，因此第二条sql语句把IsSuccess修改为0，IsSuccess非主键索引锁了值为0的索引数据，第二条sql语句将无法把数据更新到被锁的行里。

再执行下面的sql语句

```
1  -- 1:
2  set autocommit=0;
3  begin;
4  select * from table1 where getTime < 1 and IsSuccess=0 order by id asc limit 0,30 for update;
5  commit;
6  -- 2:
7  update table1 a set IsSuccess=2 where id =400000;
```

这样第二条sql语句将可以执行。因为IsSuccess=2的索引段没有被锁。

上面的例子知道了锁索引段后还比较容易看懂，下面就奇葩一点：

先把id =400000数据的GetTime修改为1，IsSuccess=0，然后一次执行sql：

公告

昵称：Lawson
园龄：9年2个月
粉丝：54
关注：32
+加关注

2017年12月						
<	日	一	二	三	四	五
	26	27	28	29	30	1
	3	4	5	6	7	8
	10	11	12	13	14	15
	17	18	19	20	21	22
	24	25	26	27	28	29
	31	1	2	3	4	5

搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

最新随笔

1. 远程debug调试java代码
2. JAVA语言搭建白盒静态代...
3. Linux上搭建Hadoop2.6....
4. B/S系统常见缺陷整理和...
5. 防止sql注入和sqlmap介绍
6. GoCom自行部署版0day...
7. Mysql查询语句使用selec...
8. i春秋url地址编码问题
9. 利用旅游有缘人社交+B2...
10. 数据推送装置及方法专利...

我的标签

JAVA (5) .NET (3) cvs (2)
性能计数 (2) 异步 (2)
mysql (2) NLog (1)
Nutch (1) OAuth (1)
office (1) 更多

随笔档案(71)

2016年12月 (1)
2016年3月 (1)

```
1  -- 1:
2  set autocommit=0;
3  begin;
4  update ctripticketchangeresultdata a set issuccess=1 where id =400000;
5  commit;
6  -- 2:
7  select * from table1 where getTime < 1 and IsSuccess=0 order by id asc limit 0,30 for update;
```

第1个sql先不commit，按照道理只会锁40000这行记录，第二个sql执行，按照道理只能查询从400001记录的30条记录，但第二个sql语句会阻塞等待。

原因是第一个sql语句还没有commit也没有rollback，因此它先锁主键索引，再锁IsSuccess的非主键索引，第二个sql语句由于where里要判断IsSuccess字段的值，由于400000这条数据以前的IsSuccess是0，现在更新为1还不确定，可能会回滚，因此sql2需要等待确定400000这条数据的IsSuccess是否被修改。sql2的sql语句因为判断了GetTime<1，实际400000这条记录已经不满足了，但按照锁索引的原理，所以sql2语句会被阻塞。

因此如果根据**业务场景**，可以把sql2语句的IsSuccess条件取消掉，并且这里GetTime查询条件由GetTime<1修改为GetTime=0，这样即可不阻塞直接查询出来。

GetTime用范围查询导致的锁影响经过分析，还不是间隙锁的问题，感觉应该用范围作为条件，所有从第0行开始的所有查找范围都会被锁住。比如这里更新400000会被阻塞，但更新400031不会被阻塞。

我们项目出现死锁，就是这个原理，一条sql语句先锁主键索引，再锁非主键索引；另外一条sql语句先锁非主键索引，再锁主键索引。虽然两个sql语句期望锁的数据行不一样，但两个sql语句查询或更新的条件或结果字段如果有相同列，则可能会导致互相等待对方锁，2个sql语句即引起了死锁。

个人总结一下innodb存储引擎下的锁的分析，可能会有问题：

- 1、更新或查询for update的时候，会在where条件中开始为每个字段判断是否有锁，如果有锁就会等待，因为如果有锁，那这个字段的值不确定，只能等待锁commit或rollback后数据确定后再查询。
- 2、另外还和order by有关系，因为可能前面数据有锁，但从后面查询一个范围就可以查询。
- 3、另外limit也有关系，比如limit 20,30从第20条记录取30行数据，但第一行数据如果被锁，因为不确定回滚还是提交，也会锁等待。

因此从筛选查询条件经过的地方都会判断锁，如果有锁，因为数据不确定，都会等待锁释放。本文是个人测试结果，没有深入分析内部原理，可能有不准确的地方。留作自己以后参考。

标签: mysql

好文要顶 关注我 收藏该文

Lawson

关注 - 32

粉丝 - 54

加关注

11

- « 上一篇: i春秋url地址编码问题
- » 下一篇: GoCom自行部署版Oday漏洞getshell和查看所有人聊天记录等问题记录

posted @ 2015-11-30 22:58 Lawson 阅读(22120) 评论(2) 编辑 收藏

评论列表

#1楼

2016-09-30 21:05 罗霄(Sto)

写得非常棒！

支持(0) 反对(0)

#2楼

2017-02-09 16:32 走走or停停

“第1个sql先不commit，按照道理只会锁40000这行记录，第二个sql执行，按照道理只能查询从400001记录的30条记录，但第二个sql语句会阻塞等待。”

第二个sql执行 正常就是查询从40000开始的30条记录，因为第一个sql没有提交，除非你的数据库隔离级别是读未提交 不过正常mysql默认的隔离级别是repeatable read，是不会读到其他事物未提交的数据的

支持(1) 反对(0)

刷新评论 刷新页面 返回顶部

- 2016年1月 (2)
- 2015年12月 (2)
- 2015年11月 (2)
- 2015年10月 (2)
- 2015年9月 (2)
- 2015年6月 (1)
- 2014年6月 (2)
- 2013年10月 (1)
- 2013年3月 (2)
- 2012年12月 (5)
- 2012年11月 (1)
- 2012年10月 (1)
- 2012年9月 (2)
- 2012年8月 (1)
- 2012年7月 (8)
- 2012年6月 (1)
- 2012年5月 (1)
- 2012年4月 (1)
- 2012年2月 (1)
- 2012年1月 (1)
- 2011年12月 (2)
- 2011年11月 (1)
- 2011年10月 (1)
- 2011年9月 (1)
- 2011年8月 (1)
- 2011年7月 (3)
- 2011年5月 (1)
- 2011年4月 (2)
- 2011年2月 (2)
- 2011年1月 (2)
- 2010年12月 (5)
- 2010年11月 (3)
- 2008年12月 (1)
- 2008年11月 (4)
- 2008年10月 (1)

相册⁽¹⁾

我(1)

最新评论

- 1. Re:代码性能分析-Dottra...
给我个注册机谢谢。
327965506@qq.com
--眼泪的距离
- 2. Re:搭建内网搜索平台
如果能有相关的源代码就更好了
~~~  
--菜鸟\_黑马
- 3. Re:搭建内网搜索平台  
写得非常不错，最近正打算也做一个类似的项目~~~受教了~~~  
--菜鸟\_黑马
- 4. Re:JAVA和.NET开发过程...  
不错，一直对Java不了解，很想了解了解。  
--荆棘人
- 5. Re:Mysql查询语句使用s...  
“第1个sql先不commit，按照道理只会锁40000这行记录，第二个sql执行，按照道理只能查询从400001记录的30条记录，但第二个sql语句会阻塞等待。”第二个sql执行 正常就是查询从40.....  
--走走or停停

阅读排行榜

- 1. Mysql查询语句使用selec...
- 2. 配置精简版Oracle客户端(...
- 3. Tomcat配置虚拟目录、...
- 4. 浅谈ClassLoader(14278)
- 5. mysql在linux下修改存储...

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

最新IT新闻:

- 网约车司机疲劳驾驶成马路杀手 Uber将规定时间上限
  - 设计图曝光：三星双屏折叠手机原来是这样的
  - 比特币一年涨20倍？别只见吃肉 忘了投机客跳楼
  - 朱啸虎谈最讨厌的创业者：认为自己什么都是对的
  - 智能音箱大战全面开火，那如何成为一名全栈语音识别工程师？
- » 更多新闻...

最新知识库文章:

- 以操作系统的角度述说线程与进程
  - 软件测试转型之路
  - 门内门外看招聘
  - 大道至简，职场上做人做事做管理
  - 关于编程，你的练习是不是有效的？
- » 更多知识库文章...

评论排行榜

1. 重构桌面飘着圣诞老人，...
2. 服务器被挂Iframe木马的...
3. 存储过程内建临时表和临...
4. 代码性能分析-Dottrace...
5. JAVA和.NET使用DES对...

推荐排行榜

1. 对IIS7经典模式和集成模...
2. JAVA和.NET使用DES对...
3. MSSQL表分区的创建(4)
4. Tomcat配置虚拟目录、...
5. 浅谈ClassLoader(2)