



大咖丁奇：索引存储顺序和order by不一致，怎么办？

netcat20000

发布于 2022-7-13 16:12

0收藏 2178浏览

我是林晓斌，今天作为【迪B课堂】的客串嘉宾来跟大家分享：**当索引存储顺序和order by不一致，该怎么办？**

林晓斌

林晓斌，网名丁奇，腾讯云数据库负责人，数据库领域资深技术专家。作为活跃的MySQL社区贡献者，丁奇专注于数据存储系统、MySQL源码研究和改进、MySQL性能优化和功能改进，在业务场景分析、系统瓶颈分析、性能优化方面拥有丰富的经验。其创作的《MySQL实战45讲》专栏受众已逾2万人。

根据指定的字段排序来显示结果，是我们写应用时最常见的需求之一了，比如一个交易系统中，按照交易时间倒序显示交易记录。

相信你听说过这样的建议：如果有order by的需求，给需要排序的字段加上索引，就可以避免数据库排序操作。

所谓数据库排序操作，是指数据库在执行过程中，先将满足条件的数据全部读出来，放入内存中，再执行快排，这个内存就是sort_buffer。
如果临时数据量比sort_buffer大，就要把数据放入临时文件，然后做外部排序。
这个排序过程的消耗是比较大的。

所谓避免数据库排序操作，是指执行过程中不需要快排或外部排序。

为什么加上索引就可以避免排序呢？如果索引存储顺序和order by不一致，还需要排序吗？如果是联合索引呢？

今天我们就来说一说，建了索引以后，order by是怎么执行的以及怎么优化。

为了便于说明，我创建一个简单的表，这个表里，除了主键索引id外，还有一个联合索引ab。你可以在文稿中看到这个表的定义。

```
1 CREATE TABLE `t` (  
2   `id` int(11) NOT NULL,  
3   `a` int(11) NOT NULL,  
4   `b` int(11) NOT NULL,  
5   `c` int(11) NOT NULL,  
6   PRIMARY KEY (`id`),  
7   KEY `ab` (`a`,`b`)  
8 ) ENGINE=InnoDB;
```

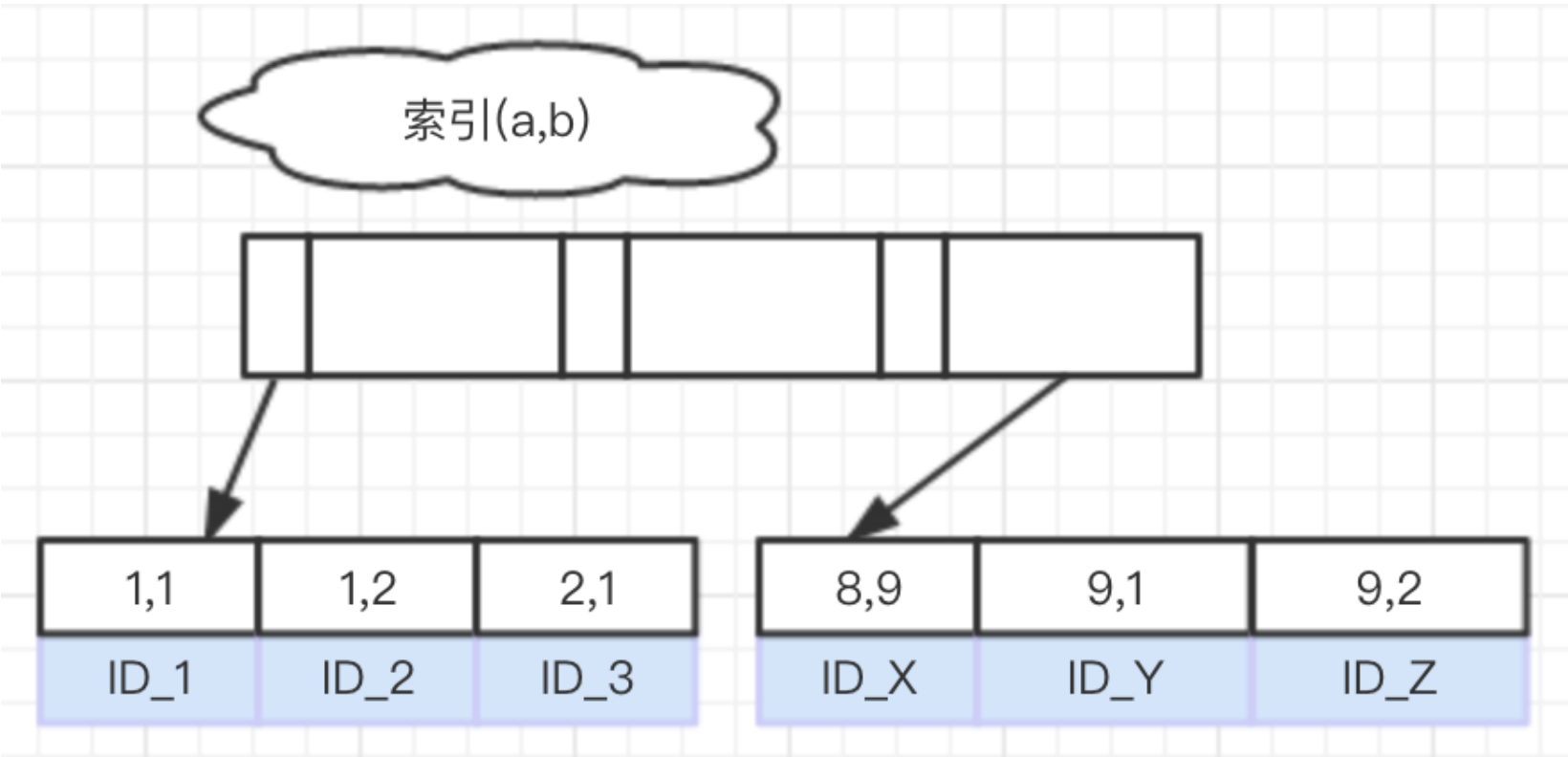
我们来看看不同的业务需求下，SQL语句怎么写，以及在MySQL里是怎么执行的。

单字段排序

一个简单的需求是将这个表的数据，按照a的大小倒序返回。你的SQL语句可以这么写：

```
1 select * from t order by a desc;
```

我们来看看这个联合索引ab的结构，点击可以查看大图。



netcat20000

LV.2

这个用户很懒，还没有个人简介

92

0

150

0

帖子

视频

声望

粉丝

私信

+ 关注

最近发布 觉得TA不错？点个关注精彩不错过

删库跑路只用1秒，数据恢复7天7夜...

2022-07-15 17:58:27发布

鹅厂资深DBA是怎么做数据梳理的？

2022-07-15 17:58:14发布

热门推荐

#星计划# 基于鸿蒙应用开发：每日新闻头条 0回复

#星计划# 丸骑行-OpenHarmony骑行助手 5回复

#星计划#鸿蒙原生应用有问必答 5回复

#星计划# 基于鸿蒙应用开发：任务列表 1回复

#星计划#浅谈ArkUI之Web组件的基础用法 4回复

相关问题

签名不一致报错怎么回事？ 1回答

docker 部署的iserver 操作日志时间与系统不一致 1回答

redis 单线程 是不是就没有并发不一致的问题了 1回答

mysql 主从复制数据不一致，出现I Error_code: 1032 错误什么原因啊？ 1回答

为了满足不同场景下对一致性级别的要求，PolarDB 提供了哪三种一致性级别？ 1回答



上一篇： 前沿观察 | 图数据库项目D...

下一篇： 《权游》人物关系你还捋...

社区精华内容





因此上面这个语句的执行流程就是：

1. 从索引ab上，取最右的一个记录，取出主键值ID_Z；
2. 根据ID_Z到主键索引上取整行记录，作为结果集的第一行；
3. 在索引ab上取上一个记录的左边相邻的记录；
4. 每次取到主键id值，再到主键索引上取到整行记录，添加到结果集的下一行；
5. 重复步骤3、4，直到遍历完整个索引。

可以看到，这个流程中并不涉及到排序操作。我们也可以用explain语句来验证这个结论。

图2是这个语句的explain的结果，可以看到，Extra字段中没有Using filesort字样，说明这个语句执行过程中，不需要用到排序。

```
mysql> explain select * from t order by a desc;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	t	NULL	index	NULL	ab	8	NULL	1000	100.00	Backward index scan

1 row in set, 1 warning (0.00 sec)

图2 order by 不需要排序

组合字段排序

有了上面的分析，我们再来看看下面这个语句：

```
1 select * from t order by a desc,b desc;
```

这个语句的意思是，按照a值倒序，当a的值相同时按照b值倒序。
你一定发现了，这个语句的执行逻辑和执行结果，跟前面的语句是一模一样的，因此也不需要排序。

倒序不需要排序，正序呢？正序的语句是这么写的：

```
1 select * from t order by a,b;
```

显然，这个语句也是不需要排序的，执行流程上，只需要先取ab索引树最左边的节点，然后向右遍历即可。

到这里我们可以小结一下：

1. InnoDB索引树以任意一个叶节点为起始点，可以向左或向右遍历；
2. 如果语句需要的order by顺序刚好可以利用索引树的单向遍历，就可以避免排序操作。

Descending Indexes

接下来我们来看一种不满足”单向遍历“的场景。

```
1 select * from t order by a,b desc;
```

这个语句要求查询结果中的记录排序顺序是：按照a值正序，对于相同的a值，按照b值倒序。
由于不满足单向遍历的要求，因此只能选择使用排序操作。

图3是这个语句explain的结果。

```
mysql> explain select * from t order by a,b desc;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	t	NULL	ALL	NULL	NULL	NULL	NULL	1000	100.00	Using filesort

1 row in set, 1 warning (0.00 sec)

图3 order by 需要排序

extra字段中Using filesort表示使用了排序。

你一定想到了，如果可以让InnoDB在构建索引ab的时候，相同的a里面，b能够从大到小排序，就又可以满足单向遍历的要求了。

在MySQL5.7及之前的版本是不支持这么创建索引的，在8.0版本中支持了这个功能，官方名称是Descending Indexes。
在8.0版本中，我们可以把索引ab的定义做个修改。





```
3      a` int(11) NOT NULL,
4      `b` int(11) NOT NULL,
5      `c` int(11) NOT NULL,
6      PRIMARY KEY (`id`),
7      KEY `ab` (`a`,`b` desc)
8  ) ENGINE=InnoDB;
```

我们将索引ab的定义做了修改，在字段b后面加上desc，表示对于相同的a值， 字段b按照倒序存储。

这个表对应的索引ab的结构图如下，点击可以查看大图。

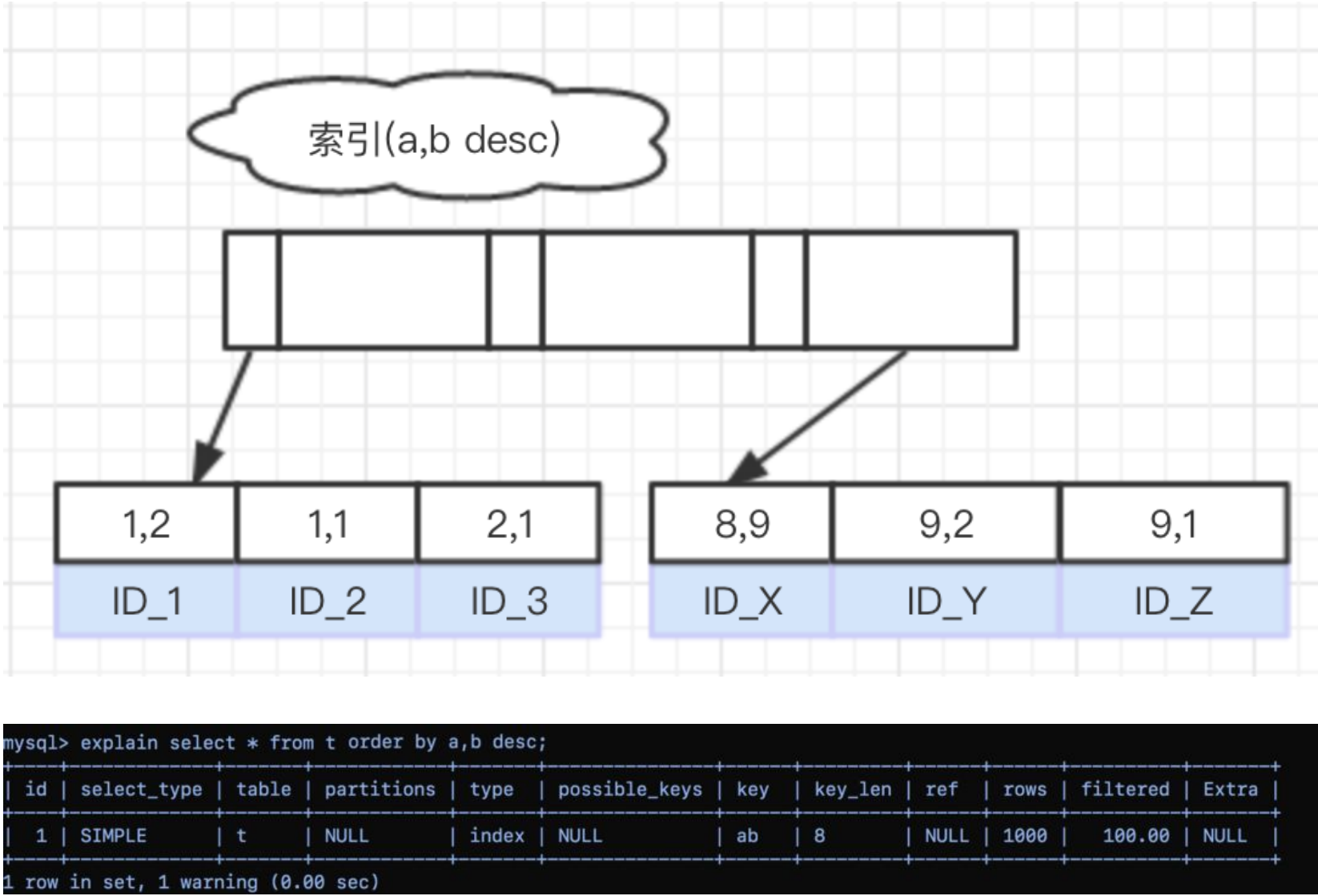


图4 索引(a, b desc) 示意图 和 explain的结果

这样从左到右遍历这个索引的时候，就刚好满足a正序，然后b逆序的要求。

Descending Indexes可以避免这种情况下的排序操作， 语句的执行性能自然就提升了。

应用优化

前面说过，Descending Indexes这个功能是在MySQL 8.0才支持的。那如果你的生产环境上使用的还是低于8.0的版本， 有没有不需要排序的方法呢？

答案是有的，接下来我给大家介绍一种应用端协作的优化方案。

假设我们现在的需求就是在MySQL 5.7版本下， 要求按照”a值正序， 然后b值逆序”的顺序， 返回所有行a和b的值。

首先，为了避免数据库排序， 我们直接执行下面这个语句：

```
mysql> explain select * from t order by a,b desc;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | t | NULL | index | NULL | ab | 8 | NULL | 1000 | 100.00 | NULL |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set, 1 warning (0.00 sec)
```

当然， 这个语句返回的结果集是不满足业务要求的， 但是我们知道， 对于相同的a值， b值是有序递增的， 我们要把这个数据特点利用起来。

执行这个语句后， 应用端的逻辑改造如下：

1. 构造一个空栈(stack)， 栈中的节点可以保存数据行；
2. 读入第一行， 入栈；
3. 读入下一行，
 - a.如果新一行中a值与上一行相同， 将新一行入栈；
 - b.如果新一行中a值与上一行不同， 则将栈中的所有数据行依次出栈并输出， 直到栈清空； 然后新一行入栈。
- 4.重复步骤3直到遍历完整个索引， 将栈中的所有数据行依次出栈并输出， 直到栈清空。


下图5是用图1中的示例数据， 执行上面的流程的效果图。






 蓝初柳 • 3499浏览 • 0回复

鸿蒙系统耗电快**怎么办**？

 Annar • 7687浏览 • 0回复


HDC2021技术分论坛：进程崩溃/应用卡死，故障频频**怎么办**？

 HarmonyOS开发者 • 7082浏览 • 0回复


忘记MySQL密码**怎么办**？一招教你搞定！

 athlon_chen • 3726浏览 • 0回复


Elasticsearch 集群状态变成黄色或者红色，**怎么办**？

 samri • 3590浏览 • 0回复


Elasticsearch 出现“429 rejected”报错，**怎么办**？

 samri • 3904浏览 • 0回复

Elasticsearch 断路器报错了，**怎么办**？

 samri • 3909浏览 • 0回复

Elasitcsearch CPU 使用率突然飙升，**怎么办**？

 samri • 3878浏览 • 0回复

大咖丁奇：加唯一索引怎么会导致丢数据？

 netcat20000 • 2764浏览 • 0回复

PostgreSQL出现死锁**怎么办**？

 LoveBank • 3530浏览 • 0回复

缓存与数据库**不一致**了，咋办？

 mb628db8cb0e5b3 • 4226浏览 • 0回复

PostgreSQL出现死锁**怎么办**？

 laomugua • 4522浏览 • 0回复

PostgreSQL出现死锁**怎么办**？

 laomugua • 3501浏览 • 0回复

发生死锁**怎么办**

 大家好我是佩奇 • 3380浏览 • 0回复

主从结构**不一致**复制问题验证

 Only1You1 • 1914浏览 • 0回复

大表分页查询非常慢，**怎么办**？

 fatherlaw • 3383浏览 • 0回复

Docker容器日志和宿主机的时间**不一致**问题解决 | #打卡不停更#

 Aasee • 1.5w浏览 • 0回复

