

图文 21 生产经验：如何通过多个Buffer Pool来优化数据库的并发性能？

546 人次阅读 2020-02-14 07:44:37

详情 评论

生产经验：如何通过多个Buffer Pool来优化数据库的并发性能？

如何提问：每篇文章都有评论区，大家可以尽情留言提问，我会逐一答疑

如何加群：购买狸猫技术窝专栏的小伙伴都可以加入狸猫技术交流群，一个非常纯粹的技术交流的地方

具体加群方式，请参见目录菜单下的文档：《MySQL专栏付费用户如何加群》（购买后可见）

1、Buffer Pool在访问的时候需要加锁吗？

前面我们已经把Buffer Pool的整体工作原理和设计原理都已经给大家分析的比较清楚了，基本上目前大家都能够很好的理解，我们对MySQL执行CRUD操作时候的第一步，就是利用Buffer Pool里的缓存来更新或者查询。

那么既然已经把Buffer Pool的原理都讲的差不多了，接着我们就可以来给大家说说Buffer Pool在实际生产环境运行中的一些经验，应该如何对Buffer Pool进行一些配置上的优化，来提升他的访问性能呢？

首先我们来看第一个问题，大家都知道，Buffer Pool其实本质就是一大块内存数据结构，由一大堆的缓存页和描述数据块组成的，然后加上了各种链表（free、flush、lru）来辅助他的运行。

好，那么这个时候假设MySQL同时接收到了多个请求，他自然会用多个线程来处理这多个请求，每个线程会负责处理一个请求，对吧？

然后这多个线程是不是应该会同时去访问Buffer Pool呢？就是同时去操作里面的缓存页，同时操作一个free链表、flush链表、lru链表，是吗？

我们看下图，就是一个多线程并发访问Buffer Pool的示意图。



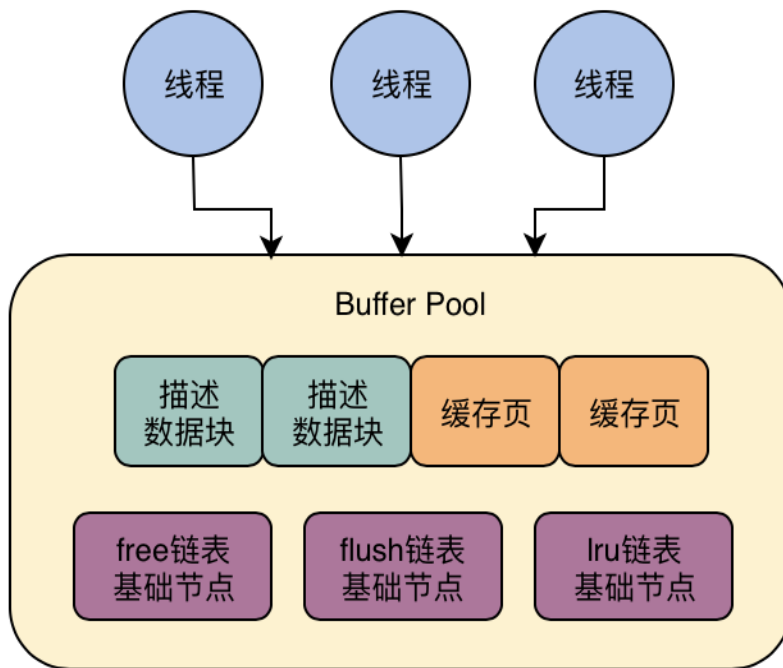
狸猫技术窝

进店逛

相关频道



从零开始
MySQL 进阶教程
已更新3



那么大家思考一下，现在多个线程来并发的访问这个Buffer Pool了，此时他们都是在访问内存里的一些共享的数据结构，比如说缓存页、各种链表之类的，那么此时是不是必然要进行加锁？

对，多线程并发访问一个Buffer Pool，必然是要加锁的，然后让一个线程先完成一系列的操作，比如说加载数据页到缓存页，更新free链表，更新lru链表，然后释放锁，接着下一个线程再执行一系列的操作。

2、多线程并发访问加锁，数据库的性能还能好吗？

既然我们已经解决了第一个问题，就是多线程并发访问一个Buffer Pool的时候必然会加锁，然后很多线程可能要串行着排队，一个一个的依次执行自己要执行的操作，那么此时我问大家第二个问题，此时数据库的性能还能好吗？

应该这么说，即使就一个Buffer Pool，即使多个线程会加锁串行着排队执行，其实性能也差不到哪儿去。

因为大部分情况下，每个线程都是查询或者更新缓存页里的数据，这个操作是发生在内存里的，基本都是微秒级的，很快很快，包括更新free、flush、lru这些链表，他因为都是基于链表进行一些指针操作，性能也是极高的。

所以即使每个线程排队加锁，然后执行一系列操作，数据库的性能倒也是还可以的。

但是再怎么可以，你毕竟也是每个线程加锁然后排队一个一个操作，这也不是特别的好，特别是有的时候你的线程拿到锁之后，他可能要从磁盘里读取数据页加载到缓存页里去，这还发生了一次磁盘IO呢！所以他要是进行磁盘IO的话，也许耗时就会多一些，那么后面排队等他的线程自然就多等一会儿了！

3、MySQL的生产优化经验：多个Buffer Pool优化并发能力

因此这里我们给大家介绍一个MySQL的生产环境优化经验，就是可以给MySQL设置多个Buffer Pool来优化他的并发能力。

一般来说，MySQL默认的规则是，如果你给Buffer Pool分配的内存小于1GB，那么最多就只会给你一个Buffer Pool。

如果你的机器内存很大，那么你必然会给Buffer Pool分配较大的内存，比如给他个8G内存，那么此时你是同时可以设置多个Buffer Pool的，比如说下面的MySQL服务器端的配置。

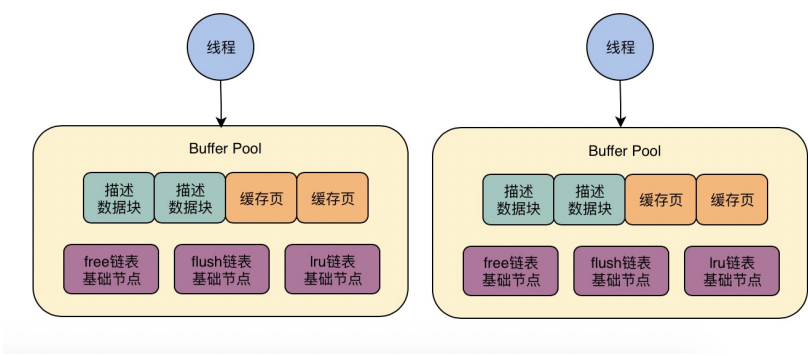
```
[server]
innodb_buffer_pool_size = 8589934592
innodb_buffer_pool_instances = 4
```

我们给buffer pool设置了8GB的总内存，然后设置了他应该有4个Buffer Pool，此时就是说，每个buffer pool的大小就是2GB

这个时候，MySQL在运行的时候就会有4个Buffer Pool了！每个Buffer Pool负责管理一部分的缓存页和描述数据块，有自己独立的free、flush、lru等链表。

这个时候，假设多个线程并发过来访问，那么不就可以把压力分散开来了吗？有的线程访问这个buffer pool，有的线程访问那个buffer pool。

我们看下图：



所以这样的话，一旦你有了多个buffer pool之后，你的多线程并发访问的性能就会得到成倍的提升，因为多个线程可以在不同的buffer pool中加锁和执行自己的操作，大家可以并发来执行了！

所以这个在实际生产环境中，设置多个buffer pool来优化高并发访问性能，是mysql一个很重要的优化技巧。

End

专栏版权归公众号狸猫技术窝所有

未经许可不得传播，如有侵权将追究法律责任

狸猫技术窝精品专栏及课程推荐：

- [《从零开始带你成为消息中间件实战高手》](#)
- [《21天互联网Java进阶面试训练营》（分布式篇）](#)
- [《互联网Java工程师面试突击》（第1季）](#)
- [《互联网Java工程师面试突击》（第3季）](#)
- [《从零开始带你成为JVM实战高手》](#)