

案例实战：线上数据库莫名其妙的随机性能抖动优化（下）

上一篇文章我们已经给大家详细分析了有时候我们在数据库里执行查询或者更新语句的时候，可能SQL语句的性能会出现不正常的莫名其妙的抖动，平时可能几十毫秒搞定的，现在居然要几秒钟。

其实这种莫名其妙的性能抖动，我们在分析过底层的原理之后，就理解的很清楚了，根本原因还是两个。

第一个，可能buffer pool的缓存页都满了，此时你执行一个SQL查询很多数据，一下子要把很多缓存页flush到磁盘上去，刷磁盘太慢了，就会导致你的查询语句执行的很慢。

因为你必须等很多缓存页都flush到磁盘了，你才能执行查询从磁盘把你需要的数据页加载到buffer pool的缓存页里来。

第二个，可能你执行更新语句的时候，redo log在磁盘上的所有文件都写满了，此时需要回到第一个redo log文件覆盖写，覆盖写的时候可能就涉及到第一个redo log文件里有很多redo log日志对应的更新操作改动了缓存页，那些缓存页还没flush到磁盘，此时就必须把那些缓存页flush到磁盘，才能执行后续的更新语句，那你这么一等待，必然会导致更新执行的很慢了。

所以上述两个场景导致的大量缓存页flush到磁盘，就会导致莫名其妙的SQL语句性能抖动了

那今天我们来说说怎么尽可能优化MySQL的一些参数，减少这种缓存页flush到磁盘带来的性能抖动问题。

其实大家可以想一下，如果要尽量避免缓存页flush到磁盘可能带来的性能抖动问题，那么核心的就两点

第一个是尽量减少缓存页flush到磁盘的频率，第二个是尽量提升缓存页flush到磁盘的速度。

那你想要减少缓存页flush到磁盘的频率，这个是很困难的，因为平时你的缓存页就是正常的在被使用，迟早会被填满，一旦填满，必然你执行下一个SQL会导致一批缓存页flush到磁盘，这个很难控制，除非你给你的数据库采用大内存机器，给buffer pool分配的内存空间大一些，那么他缓存页填满的速率低一些，flush磁盘的频率也会比较低。

所以今天我们主要是给讲解第二个问题的优化，就是如何尽量提升缓存页flush到磁盘的速度

给大家举个例子，假设你现在要执行一个SQL查询语句，此时需要等待flush一批缓存页到磁盘，接着才能加载查询出来的数据到缓存页。

那么如果flush那批缓存页到磁盘需要1s，然后SQL查询语句自己执行的时间是200ms，此时你这条SQL执行完毕的总时间就需要1.2s了。

但是如果你把那批缓存页flush到磁盘的时间优化到100ms，然后加上SQL查询自己执行的200ms，这条SQL的总执行时间就只要300ms了，性能就提升了很多。

所以这里一个关键之一，就是要尽可能减少flush缓存页到磁盘的时间开销到最小。

如果要做到这一点，通常给大家的一个建议就是对于数据库部署的机器，一定要采用SSD固态硬盘，而不要使用机械硬盘，因为SSD固态硬盘最强大的地方，就是他的随机IO性能非常高。

而flush缓存页到磁盘，就是典型的随机IO，需要在磁盘上找到各个缓存页所在的随机位置，把数据写入到磁盘里去。所以如果你采用的是SSD固态硬盘，那么你flush缓存页到磁盘的性能首先就会提高不少。

其次，光是用SSD还不够，因为你还得设置一个很关键的参数，就是数据库的innodb_io_capacity，这个参数是告诉数据库采用多大的IO速率把缓存页flush到磁盘里去的。

举个例子，假设你SSD能承载的每秒随机IO次数是600次，结果呢，你把数据库的innodb_io_capacity就设置为了300，也就是flush缓存页到磁盘的时候，每秒最多执行300次随机IO，那你不是速度很慢么，而且根本没把你的SSD固态硬盘的随机IO性能发挥出来！

所以通常都会建议大家对数据库部署机器的SSD固态硬盘能承载的最大随机IO速率做一个测试，这个可以使用fio工具来测试

fio工具是一种用于测试磁盘最大随机IO速率的linux上的工具，如何使用，大家可以网上搜一下，非常的简单。

查出来SSD固态硬盘的最大随机IO速率之后，就知道他每秒可以执行多少次随机IO，此时你把这个数值设置给数据库的innodb_io_capacity这个参数就可以了，尽可能的让数据库用最大速率去flush缓存页到磁盘。

但是实际flush的时候，其实他会按照innodb_io_capacity乘以一个百分比来进行刷磁盘，这个百分比就是脏页的比例，是innodb_max_dirty_pages_pct参数控制的，默认是75%，这个一般不用动，另外这个比例也有可能会变化，这个比例同时会参考你的redo log日志来计算，但是这个细节大家不用关注了。

其实比例不比例的，我们这里优化不用关注，核心就是把innodb_io_capacity调整为SSD固态硬盘的IOPS也就是随机IO速率就可以了。

另外还有一个参数，是innodb_flush_neighbors，他意思是说，在flush缓存页到磁盘的时候，可能会控制把缓存页临近的其他缓存页也刷到磁盘，但是这样有时候会导致flush的缓存页太多了。

实际上如果你用的是SSD固态硬盘，并没有必要让他同时刷邻近的缓存页，可以把innodb_flush_neighbors参数设置为0，禁止刷临近缓存页，这样就把每次刷新的缓存页数量降低到最少了。

所以呢，针对这次讲的这个案例，就是MySQL性能随机抖动的问题，**最核心**的就是把innodb_io_capacity设置为SSD固态硬盘的IOPS，让他刷缓存页尽量快，同时设置innodb_flush_neighbors为0，让他每次别刷临近缓存页，减少要刷缓存页的数量，这样就可以把刷缓存页的性能提升到最高。

同时也可以尽可能降低每次刷缓存页对执行SQL语句的影响。

End