

详情 评论

从数据的增删改开始讲起，回顾一下Buffer Pool在数据库里的地位

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、一切从数据的增删改开始讲起

好了，到这一讲为止，我们实际上已经初步的讲解了一下MySQL的整体架构设计原理，大家对于MySQL内部包含哪些组件，我们平时更新数据以及查询数据的时候，大致都是怎么做的，都已经有一个比较高层次的了解了。

另外现在我们初步的了解了MySQL的架构原理之后，还给大家介绍了一些我们的数据库相关的生产经验，就是对于任何一个项目，数据库都需要选择好合适的机器，同时做好压测，并且有一个完善的可视化监控系统。

现在可以理解为每个人手头都有了一个可用的数据库，而且对数据库的整体架构原理都有了一定的理解了。那么接下来，我们这个专栏一共有100多讲的内容，我们接着当然要细细的讲解数据库的方方面面了

那我们应该从哪个环节开始入手呢？

当然是从数据库的增删改开始了，因为当你手头有了一个数据库之后，你必然就会去开发一个系统，系统就直接基于数据库做各种增删改查的操作，实现各种各样的业务逻辑

而任何一个系统在使用数据库的时候，一定是从插入数据开始的，也就是首先先会对数据进行增删改的操作。

当你的数据库中有了数据之后，接着才会执行各种各样的查询操作。

所以我们专栏的讲解顺序，就按照你手头有了一个经过压测的、有完善监控的数据库之后，你开发的系统使用数据库的顺序来讲解，先讲解系统对数据库执行各种增删改操作时背后对应的内幕原理，以及事务的原理，包括锁的底层机制，然后讲解你有了数据之后，执行各种复杂的查询操作的时候，涉及到的索引底层原理，查询优化的底层原理。

当然这个中间我们会穿插各种各样的生产实践的案例，就跟我之前讲解的《从0开始带你成为JVM实战高手》专栏一样。

然后讲解完这些之后，我们再来讲解平时我们在开发系统的时候，如何进行数据库的建模，在数据库建模的时候，应该如何注意字段类型、索引类型的一些问题，如何保证数据库避免死锁、高性能的运行。

接着我们再讲解一些高阶的数据库架构设计，比如说主从架构设计以及分库分表架构设计，包括一些生产实践的案例。

所以上面的这些就是我们专栏接下来将要讲解的顺序，这里要给大家提前通知一下，我在实际讲解的过程中，会增加很多内容，比如接下来好几讲都是深度分析Buffer Pool的内容，实际上在原来的大纲中都是没有的。

另外我接下来讲解的过程中，还可能随时会对大纲中原有内容的顺序做出调整，比如说我在讲解完Buffer Pool之后，接着可能直接会深入讲解redo log、undo log、binlog这些机制，同时接着讲解事务机制，锁机制，底层数据存储机制。

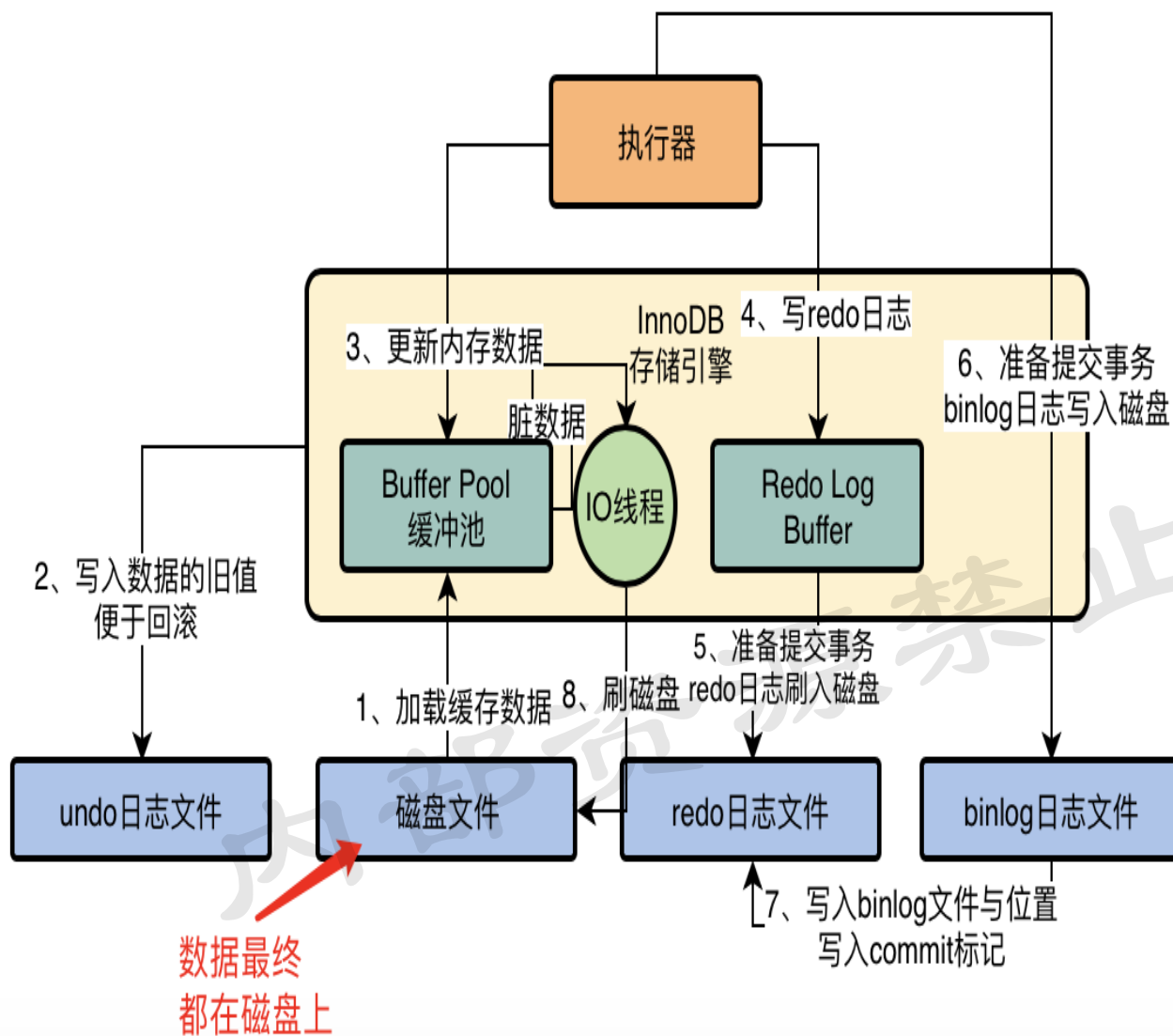
然后这些都讲完之后，才是讲解索引和查询优化的内容，所以希望大家能明白我们随时会对大纲内容做出额外的扩充，以及我们随时会调整大纲内容的顺序。

好，那么从这篇文章开始，让我们一起来探索数据库的各种底层机制和生产实践案例吧！

2、回顾一下Buffer Pool是个什么东西？

现在我们先来回顾一下数据库中的Buffer Pool是个什么东西？其实他是一个非常关键的组件，因为我们通过之前的讲解都知道一点，那就是数据库中的数据实际上最终都是要存放在磁盘文件上的，如下图所示。

内部资源禁止外传

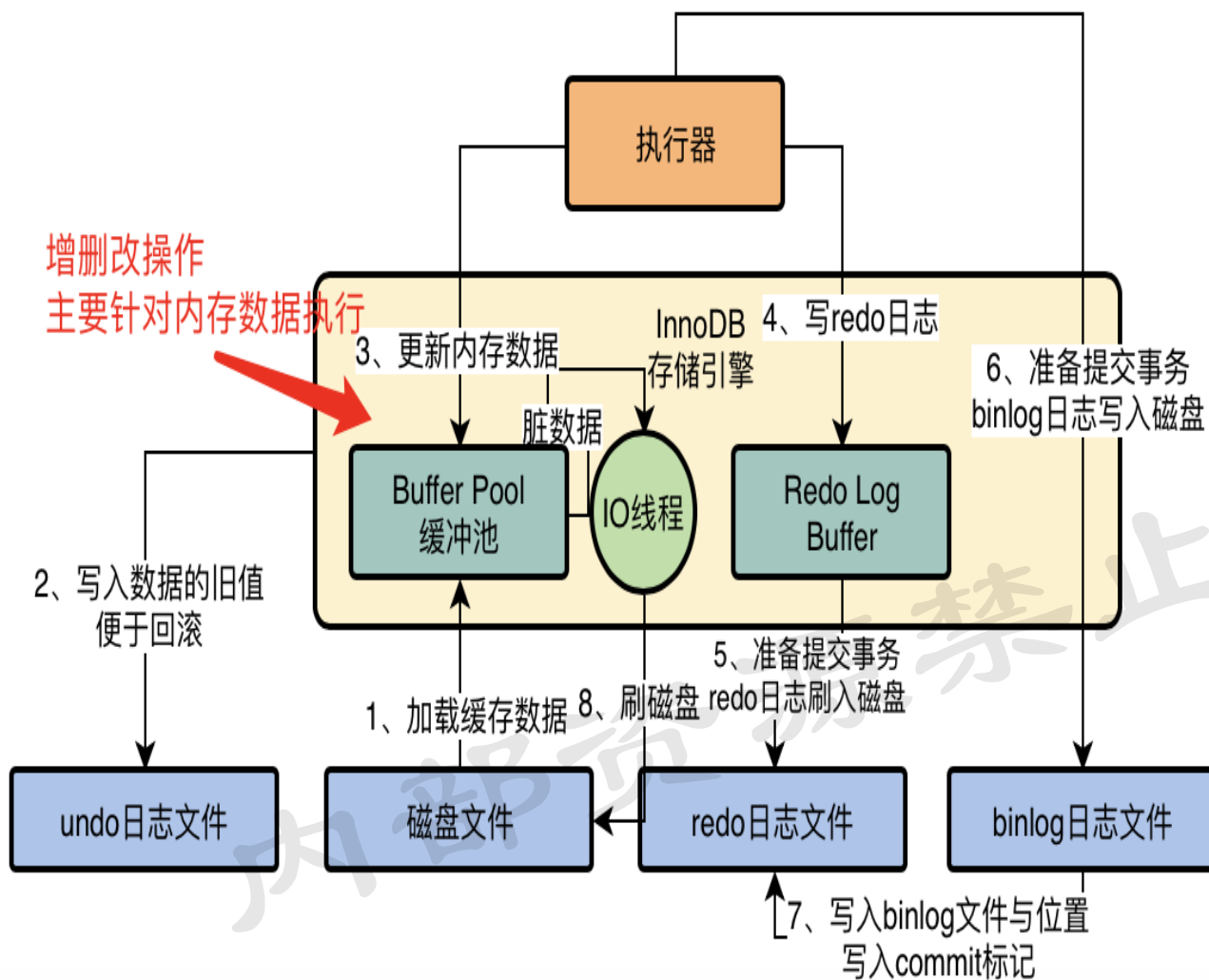


但是我们在对数据库执行增删改操作的时候，不可能直接更新磁盘上的数据的，因为如果你对磁盘进行随机读写操作，那速度是相当慢的，随便一个大磁盘文件的随机读写操作，可能都要几百毫秒。如果要是那么搞的话，可能你的数据库每秒也就只能

处理几百个请求了！

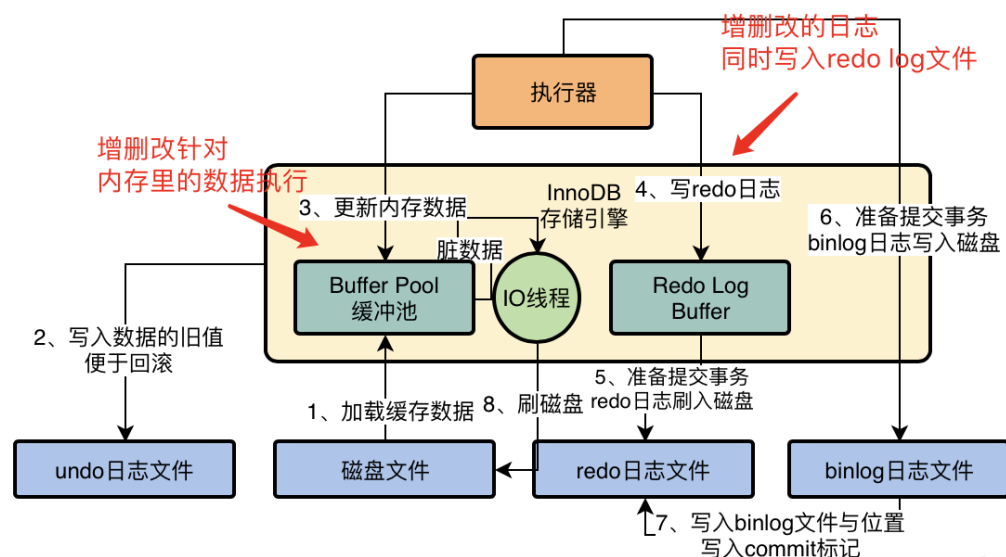
之前我们也都讲解过了，你在对数据库执行增删改操作的时候，实际上主要都是针对内存里的Buffer Pool中的数据进行的，也就是你实际上主要是对数据库的内存里的数据结构进行了增删改，如下图所示。

内部资源禁止外传



当然，我们之前都说过，其实每个人都担心一个事，就是你在数据库的内存里执行了一堆增删改的操作，内存数据是更新了，但是这个时候如果数据库突然崩溃了，那么内存里更新好的数据不是都没了吗？

所以其实之前我们开头就用了很多篇幅讲这个问题，MySQL就怕这个问题，所以引入了一个redo log机制，你在对内存里的数据进行增删改的时候，他同时会把增删改对应的日志写入redo log中，如下图。



万一你的数据库突然崩溃了，没关系，只要从redo log日志文件里读取出来你之前做过哪些增删改操作，瞬间就可以重新把这些增删改操作在你的内存里执行一遍，这就可以恢复出来你之前做过哪些增删改操作了。

当然对于数据更新的过程，他是有一套严密的步骤的，还涉及到undo log、binlog、提交事务、buffer pool脏数据刷回磁盘，等等。我们之前都讲过了，这里不再重复，仅仅是带着大家重新回顾一下数据库中的Buffer Pool这个东西。

3、Buffer Pool的一句话总结

所以这里我们简单对Buffer Pool这个东西做一下总结，他其实是数据库中我们第一个必须要搞清楚的核心组件，因为增删改操作首先就是针对这个内存中的Buffer Pool里的数据执行的，同时配合了后续的redo log、刷磁盘等机制和操作。

所以Buffer Pool就是数据库的一个内存组件，里面缓存了磁盘上的真实数据，然后我们的Java系统对数据库执行的增删改操作，其实主要就是对这个内存数据结构中的缓存数据执行的。

这一篇文章我们先对Buffer Pool这个东西的定位做一个简单的回顾，下一篇文章我们分析一下Buffer Pool这个内存数据结构里到底包含了一些什么东西。

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》](#)

Copyright © 2015-2020 深圳小鹅网络技术有限公司 All Rights Reserved. [粤ICP备15020529号](#)

 小鹅通提供技术支持