

28 | 一致性与共识（三）：共识与事务之间道不明的关系

2022-04-06 陈现麟

《深入浅出分布式技术原理》

课程介绍 >



讲述：张浩

时长 09:57 大小 9.12M



你好，我是陈现麟。

通过上节课的学习，我们知道了共识问题的使用场景、定义和经典的算法，并且从共识的角度深入探讨了一致性和共识的关系，这让我们对一致性和共识的理解更进了一步。

你应该还记得，在课程第 23 讲“原子性”中提到过，当我们在实现事务的原子性时，采用的是 2PC 或 3PC 这样的共识协议；同时，在课程第 25 讲“持久性”中我们也讲过，通过线性一致性算法来复制数据，可以提高事务的持久性。另外，最显而易见的就是，事务的 ACID 中，C 就是一致性。

那么，你一定在想，在分布式事务中，共识与事务之间是什么关系呢？是不是像共识和线性一致性一样，共识是方法和手段，事务的一致性为目的呢？

在这节课中，我们就一起来讨论一下共识与事务之间的关系。我们先从事务的特性 ACID 的维度，一一来分析事务与共识的关系，然后以它们的关系为基础，探讨事务的本质问题，让你深入理解事务与共识、一致性之间的联系，从根本上理解分布式事务，为以后的工作打下一个坚实的基础。

事务与共识的关系

通过课程第 22 讲“一致性”的学习，我们知道了事务的最终目的是实现一致性，即确保事务正确地将数据从一个一致性的状态，变换到另一个一致性的状态。为了达成这个目标，除了需要应用层的逻辑保证外，在事务层面还需要通过原子性、隔离性和持久性这三个特性一起协作。很有意思的一件事情是，在分布式事务中，事务这三个特性都与共识有一定的关系，下面我们来一一讨论一下。

首先，对于原子性来说，在分布式系统中，需要通过 2PC 或 3PC 之类的原子提交协议来实现。以 2PC 为例，协调者在第一阶段通过接收所有参与者对 Prepare 请求的响应，才能最终确定当前的事务是提交还是中止，而这就是典型的共识场景：所有的参与者都同意，就提交事务；如果有参与者不同意，就中止事务。**所以，我们认为 2PC 或 3PC 之类的原子提交协议是共识协议。**

另外，还要特别注意一点，我们在上节课讨论过，2PC 不是一个完备的共识算法，它满足共识算法的一致同意、诚实性以及合法性，但是在协调者出现故障的时候，并不能满足共识算法的可终止性。

其次，对于隔离性来说，我们一般通过 2PL 或 MVCC 的方式来实现，可是它们能正确实现隔离性的前提条件，建立在底层数据为单副本的基础之上。但是在分布式系统中，为了系统的高可用，底层存储的数据是多副本，**为了对事务操作表现出单副本的状态，数据的复制协议必须是线性一致性的，而线性一致性的数据复制协议，通常都是通过共识算法来实现的。**学到这里，你会发现特别有意思，我们从事务的隔离性深层次去探索，就会触碰到共识这个话题。

最后，对于持久性来说，我们在课程第 25 讲中讨论过，在分布式系统中，为了进一步提高事务的持久性，我们会对数据进行复制，通过冗余来提高持久性。虽然数据复制可以不需要共识，但是就像上一段的讨论那样，为了保障事务的隔离性，数据的复制必须是线性一致性的。所以我们可以得出，**事务为了持久性而引入了数据复制，但是为了保障隔离性，只能选择线性一致性的数据复制算法，而一旦涉及线性一致性，就说明我们又回到共识了。**

通过上面的讨论，你是否会感觉到在分布式系统中，当我们为了实现一个确定性正确的程序，一步一步深挖下去，就一定会碰到共识问题呢？其实这一点很好理解，比如在现实生活中，多人合作完成一件事情，如果人们的意见不能达成一致，是很难将事情正确完成的。想要使他们的意见达成一致，就是共识问题了，**人们通过沟通来达成共识，计算机节点之间通过交换信息来达成共识，本质上都是一样的。**

事务的本质是什么

在本专栏中，我们特别用四节课的时间做了一个“事务”系列课程，主要有两方面原因。一方面为了说明在分布式系统中，事务占有非常重要的位置，另一方面是为了让你学习到与分布式事务相关的技术原理。

但是，这些知识都是从外向内来解释事务是什么，会让我们感觉到分布式事务涉及的技术原理非常繁多，但是正因为有了这些知识的铺垫，现在我们就可以从更深的维度去探讨事务，让分布式事务变得更加简单和清晰了。那么接下来，我们就来探讨一个问题，事务的本质是什么？

首先，我们简单回忆一下事务的隔离级别：读未提交 (Read Uncommitted)、读已提交 (Read Committed)、可重复读 (Repeatable Read)、快照隔离级别 (Snapshot Isolation) 和串行化 (Serializable)，从隔离级别的名称和异常情况中，我们都不难发现，隔离级别都是从**读异常**情况的角度来定义的（其中，脏写和写倾斜也可以看成是，由于脏读和幻读导致的写异常），那么这是为什么呢？

其实这是由于事务面对的数据存储，是单副本数据或线性一致的多副本，单个写操作完成后，读操作都是可以立即读取到的，所以在单个写操作的层面，事务是不会出现异常情况的。但是，**由于事务一般都涉及对多个数据对象的读写操作，为了避免并发事务的相互影响，事务需要将还未提交的写操作结果，与其他并发事务进行隔离处理**，那么如何实现隔离呢？

既然写操作已经实际发生了，那就只能通过读操作进行隔离了，即将一个事务时间内多个离散的写操作，通过对读操作在并发事务之间隔离的方式，使事务的多个操作对外表现为一个原子操作一样。

接着，我们再来梳理一下数据一致性的模型。从课程第 26 讲“数据一致性都有哪些级别”的定义与讨论中，我们不难看出线性一致性、顺序一致性、因果一致性和最终一致性，这四种线性一致性模型讨论的都是，对单个数据对象操作时，单节点或多节点的多个写操作的顺序，以及

复制时延的问题。在数据一致性的模型中，读异常都是由于对单个数据对象的写操作，在多个副本之间的不同原子同步导致的。

到这里，我们会发现事务和数据一致性是非常类似的，它们本质上都是期望它的一个完整操作是原子操作，研究的本质问题都是数据的一致性问题。

只不过事务对一个完整操作的定义是，一个**事务**内，对一个或多个数据对象的一个或多个读写操作，它需要解决的是对多个数据对象操作的一致性问题；而**数据一致性**对一个完整操作的定义是，在多个数据副本上对一个数据对象的写操作，它要解决的是单个数据操作，复制到多个副本上的一致性问题。

“一致性与共识”系列小结

到这里，“一致性与共识”系列课程就结束了，为了让你对这部分知识有一个整体的把握，以及充分的理解，接下来我们分别从一致性、共识以及它们之间的关系出发，做一个小结。

首先，数据的一致性模型定义了，一个数据对象在多个节点上有多个副本时，对外部读写表现出来的现象。数据一致性模型从强至弱分别为：线性一致性、顺序一致性、因果一致性和最终一致性。其中线性一致性是我们目前可以实现的一致性最强的模型，对于线性一致性的数据复制模型，我们可以认为它和操作单副本是一样的结果，基于它搭建的数据系统一般都是 **CP 系统**。

而一致性级别最弱的最终一致性，它只能确保数据最终会一致，并不能明确这个时间有多长。最终一致性牺牲了数据一定程度上的正确性，换取了高性能和高可用，在高并发的互联网场景中经常被使用，基于它搭建的数据系统一般都是 **AP 系统**。

其次，共识是指多个节点（进程）对某一个事情达成一致的结果，一个完备的共识算法需要满足四个要求：一致同意、诚实性、合法性和可终止性。共识算法主要用于解决 **Leader** 选举和分布式锁服务等分布式场景中，最底层、最基础的问题，所以基于 **Leader** 的线性一致性算法，通常都需要依赖共识算法来实现选举。

最后，通过讨论共识与分布式事务之间的关系，我们发现在事务的原子性、隔离性和持久性的实现中，都可以看到共识的身影，并且当我们对事务与数据的一致性进行比较后，**发现事务是多个数据操作的一致性问题，而数据一致性则可以理解为，对多个副本的单个数据对象的事务问题。**

总结

本节课中，我们先讨论了事务与共识的关系，发现它们之间有着非常密切的关系，**世界的尽头在哪里我不知道，但是我可以明确地告诉你，分布式的尽头就是共识。**

然后，我们通过分析分布式事务，并且与数据一致性做对比，发现事务可以理解为对多个数据操作的一致性问题，这样我们对分布式事务的理解就又多了一个维度。其实深入理解事务，是学习好分布式存储的基石，也会为你以后的工作打下一个坚实的基础。

最后，我们对“一致性与共识”系列课程进行了总结和梳理，相信你对于这些知识已经有了非常深入和系统的理解，恭喜你，在学习分布式系统的道路上，跨过了“一致性与共识”这一道坎。


思考题

在课程总结中，有这样一句话，“世界的尽头在哪里我不知道，但是我可以明确地告诉你，分布式的尽头就是共识”，欢迎你来分享一下对这句话的理解。

欢迎你在留言区发表你的看法。如果这节课对你有帮助，也推荐你分享给更多的同事、朋友。

分享给需要的人，Ta订阅超级会员，你最高得 **50** 元

Ta单独购买本课程，你将得 **20** 元

 生成海报并分享

 赞 2  提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 27 | 一致性与共识（二）：它们是鸡生蛋还是蛋生鸡？

下一篇 29 | 分布式计算技术的发展史：从单进程服务到 Service Mesh

精选留言 (3)

 写留言



众口难调，只要是分布式系统就需要共识做协调。不知道这样理解是否合理？



不吃辣

2022-04-06

老师 课程马上就结束了，您讲的太好了，绝对高质量。非常希望你能讲paxos和raft算法，这一块也是多线程绕不开的知识点，补充这部分知识真的很有必要。在注册发现和配置中心中有一个mysql redis etcd zk eureka对比图，如果能理解他们的一致算法，就不用硬背这张图了。



peter

2022-04-06

请教老师一个问题：

Q1: 分布式锁怎么会和共识有关系？

