



## 说说你对分布式事务两阶段提交协议的理解

- 分布式事务是指会涉及到操作多个数据库的事务,在分布式系统中,各个节点之间在物理上相互独立,通过网络进行沟通和协调。

- XA 就是 X/Open DTP 定义的交易中间件与数据库之间的接口规范(即接口函数),交易中间件用它来通知数据库事务的开始、结束以及提交、回滚等。XA 接口函数由数据库厂商提供。

二阶段提交(Two-phase Commit)是指,在计算机网络以及数据库领域内,为了使基于分布式系统架构下的所有节点在进行事务提交时保持一致性而设计的一种算法(Algorithm)。通常,二阶段提交也被称为是一种协议(Protocol)。在分布式系统中,每个节点虽然可以知晓自己的操作时成功或者失败,却无法知道其他节点的操作的成功或失败。当一个事务跨越多个节点时,为了保持事务的 ACID 特性,需要引入一个作为协调者的组件来统一掌控所有节点(称作参与者)的操作结果并最终指示这些节点是否要把操作结果进行真正的提交(比如将更新后的数据写入磁盘等等)。因此,二阶段提交的算法思路可以概括为:参与者将操作成败通知协调者,再由协调者根据所有参与者的反馈情报决定各参与者是否要提交操作还是中止操作。

- 准备阶段

- 事务协调者(事务管理器)给每个参与者(资源管理器)发送 Prepare 消息,每个参与者要么直接返回失败(如权限验证失败),要么在本地执行事务,写本地的 redo 和 undo 日志,但不提交,到达一种“万事俱备,只欠东风”的状态。

- 提交阶段

- 如果协调者收到了参与者的失败消息或者超时,直接给每个参与者发送回滚(Rollback)消息;否则,发送提交(Commit)消息;参与者根据协调者的指令执行提交或者回滚操作,释放所有事务处理过程中使用的锁资源。(注意:必须在最后阶段释放锁资源)

- 缺点

- 同步阻塞问题

1、执行过程中,所有参与节点都是事务阻塞型的。

- 单点故障

2、由于协调者的重要性,一旦协调者发生故障。参与者会一直阻塞下去。

- 数据不一致(脑裂问题)

3、在二阶段提交的阶段二中,当协调者向参与者发送 commit 请求之后,发生了局部网络异常或者在发送 commit 请求过程中协调者发生了故障,导致只有一部分参与者接受到了 commit 请求。于是整个分布式系统便出现了数据不一致性的现象(脑裂现象)。

4、阶段无法解决的问题(数据状态不确定)

如果协调者再发出 commit 消息之后宕机,而唯一接收到这条消息的参与者同时也宕机了。那么即使协调者通过选举协议产生了新的协调者,这条事务的状态也是不确定的,没人知道事务是否被已经提交。