

一、什么是3PC

三阶段提交协议（3PC）主要是为了解决两阶段提交协议的阻塞问题，2pc存在的问题是当协作者崩溃时，参与者不能做出最后的选择。因此参与者可能在协作者恢复之前保持阻塞。三阶段提交（Three-phase commit），是二阶段提交（2PC）的改进版本。

一、什么是3PC

与两阶段提交不同的是，三阶段提交有两个改动点：

- （1）**引入超时机制**：同时在协调者和参与者中都引入超时机制。
- （2）**新增一个阶段**：在第一阶段和第二阶段中插入一个准备阶段，保证了在最后提交阶段之前各参与节点的状态是一致的。

一、什么是3PC

也就是说，除了引入超时机制之外，3PC把2PC的准备阶段再次一分为二，这样三阶段提交就有**CanCommit**、**PreCommit**、**DoCommit**三个阶段。



一、什么是3PC

1.1 CanCommit

之前2PC的一阶段是本地事务执行结束后，最后不Commit，等其它服务都执行结束并返回Yes，由协调者发生commit才真正执行commit。而这里的CanCommit指的是尝试获取数据库锁 如果可以，就返回Yes。

(1) 协调者 向 参与者 发送CanCommit请求。询问是否可以执行事务提交操作。然后开始等待 参与者的响应。

(2) 参与者 接到CanCommit请求之后，正常情况下，如果其自身认为可以顺利执行事务，则返回Yes响应，并进入预备状态。否则反馈No。

特别注意：此时并未执行事务操作，资源未被锁定。

一、什么是3PC

1.2 PreCommit

在阶段一中，如果所有的参与者都返回Yes的话，那么就会进入PreCommit阶段进行事务预提交。这里的PreCommit阶段 跟2PC的第一阶段是差不多的，只不过这里 协调者和参与者都引入了超时机制（2PC中只有协调者可以超时，参与者没有超时机制）。

一、什么是3PC

1.2 PreCommit

情况1: 假如协调者从所有的参与者获得的反馈都是Yes响应，那么就会执行事务的预执行。

(1). 发送预提交请求：协调者向参与者发送PreCommit请求，并进入Prepared阶段。

(2). 事务预提交：参与者接收到PreCommit请求后，会执行事务操作，并将undo和redo信息记录到事务日志中。

(3). 响应反馈：如果参与者成功的执行了事务操作，则返回ACK响应，同时开始等待最终指令。

一、什么是3PC

1.2 PreCommit

情况2：假如有任何一个参与者向协调者发送了No响应，或者等待超时之后，协调者都没有接到参与者的响应，那么就执行事务的中断。

- (1). **发送中断请求**：协调者向所有参与者发送abort请求。
- (2). **中断事务**：参与者收到来自协调者的abort请求之后（或超时之后，仍未收到协调者的请求），执行事务的中断。

一、什么是3PC

1.3 DoCommit

执行提交事务或者回滚操作。

一、什么是3PC

1.3 DoCommit

情况1：执行提交

- (1). **发送提交请求**：协调接收到参与者发送的ACK响应，那么他将从预提交状态进入到提交状态。并向所有参与者发送doCommit请求。
- (2). **事务提交**：参与者接收到doCommit请求之后，执行正式的事务提交。并在完成事务提交之后释放所有事务资源。
- (3). **响应反馈**：事务提交完之后，向协调者发送Ack响应。
- (4). **完成事务**：协调者接收到所有参与者的ack响应之后，完成事务。

一、什么是3PC

1.3 DoCommit

情况2：中断事务

协调者没有接收到参与者发送的ACK响应（可能是接受者发送的不是ACK响应，也可能响应超时），那么就会执行中断事务。

- (1). **发送中断请求**：协调者向所有参与者发送abort请求
- (2). **事务回滚**：参与者接收到abort请求之后，利用其在阶段二记录的undo信息来执行事务的回滚操作，并在完成回滚之后释放所有的事务资源。
- (3). **反馈结果**：参与者完成事务回滚之后，向协调者发送ACK消息
- (4). **中断事务**：协调者接收到参与者反馈的ACK消息之后，执行事务的中断。

二、2PC与3PC的区别

1.3 DoCommit

相对于2PC，3PC主要解决的单点故障问题，并减少阻塞，因为一旦参与者无法及时收到来自协调者的信息之后，他会默认执行commit。而不会一直持有事务资源并处于阻塞状态。但是这种机制也会导致数据一致性问题，因为，由于网络原因，协调者发送的abort响应没有及时被参与者接收到，那么参与者在等待超时之后执行了commit操作。这样就和其他接到abort命令并执行回滚的参与者之间存在数据不一致的情况。