ZAB

一、基本概念

ZAB 协议全称就是 ZooKeeper Atomic Broadcast protocol，是 ZooKeeper 用来实现一致性的算法。

zookeeper 根据 ZAB 协议建立了主备模型完成 zookeeper 集群中数据的同步。这里所说的主备系统架构模型是指，在 zookeeper 集群中，只有一台 leader 负责处理外部客户端的事物请求(或写操作)，然后 leader 服务器将客户端的写操作数据同步到所有的 follower 节点中。 ZAB 的协议核心是在整个 zookeeper 集群中只有一个节点即 Leader 将客户端的写操作转化为事物(或提议 proposal)。Leader 节点再数据写完之后，将向所有的 follower 节点发送数据广播请求(或数据复制)，等待所有的 follower 节点反馈。在 ZAB 协议中，只要超过半数 follower 节点反馈 OK，Leader 节点就会向所有的 follower 服务器发送 commit 消息。即将 leader 节点上的数据同步到 follower 节点之上。

ZAB 协议中主要有两种模式，第一是消息广播模式；第二是崩溃恢复模式

二、消息广播模式

1、 在 zookeeper 集群中数据副本的传递策略就是采用消息广播模式。

2、 ZAB 协议中 Leader 等待 follower 的 ACK 反馈是指"只要半数以上的 follower 成功反馈即可，不需要收到全部 follower 反馈"

3、 **zookeeper 中消息广播的具体步骤如下：**

1) 客户端发起一个写操作请求

2) Leader 服务器将客户端的 request 请求转化为事物 proposql 提案，同时为每个 proposal 分配一个全局唯一的 ID，即 ZXID。

ZXID：共有 64 位，其中高 32 位为 epoch 即 leader 的轮次，低 32 位为一个 counter，每当产生一个新的 leader，都将 epoch 自增，并将 counter 清零，而在同一个伦次中，每当产生一个新的 proposal，都将 counter 自增。

3) leader 服务器与每个 follower 之间都有一个队列，leader 将消息发送到该队列

4) follower 机器从队列中取出消息处理完(写入本地事物日志中)毕后，向 leader 服务器发送 ACK 确认。

5) leader 服务器收到半数以上的 follower 的 ACK 后，即认为可以发送 commit

6) leader 向所有的 follower 服务器发送 commit 消息。

4、 **zookeeper 采用 ZAB 协议的核心就是只要有一台服务器提交了 proposal，就要确保所有的服务器最终都能正确提交 proposal。**

5、 **leader 服务器与每个 follower 之间都有一个单独的队列进行收发消息，使用队列消息可以做到异步解耦。leader 和 follower 之间只要往队列中发送了消息即可。**

三、崩溃恢复

1、 zookeeper 集群中为保证任何所有进程能够有序的顺序执行，只能是 leader 服务器接受写请求，即使是 follower 服务器接受到客户端的请求，也会转发到 leader 服务器进行处理。

2、 如果 leader 服务器发生崩溃，则 zab 协议要求 zookeeper 集群进行崩溃恢复和 leader 服务器选举。

3、 ZAB 协议崩溃恢复要求满足如下 2 个要求：

- **确保已经被 leader 提交的 proposal 必须最终被所有的 follower 服务器提交**。
- **确保丢弃已经被 leader 出的但是没有被提交的 proposal**。

4、 根据上述要求，新选举出来的 leader 不能包含未提交的 proposal，即新选举的 leader 必须都是已经提交了的 proposal 的 follower 服务器节点。同时，新选举的 leader 节点中含有最高的 ZXID。这样做的好处就是可以避免了 leader 服务器检查 proposal 的提交和丢弃工作。

5、 leader 服务器发生崩溃时分为如下场景：

- leader 在提出 proposal 时未提交之前崩溃，则经过崩溃恢复之后，新选举的 leader 一定不能是刚才的 leader。因为这个 leader 存在未提交的 proposal。
- leader 在发送 commit 消息之后，崩溃。即消息已经发送到队列中。经过崩溃恢复之后，参与选举的 follower 服务器(刚才崩溃的 leader 有可能已经恢复运行，也属于 follower 节点范畴)中有的节点已经是消费了队列中所有的 commit 消息。即该 follower 节点将会被选举为最新的 leader。剩下动作就是数据同步过程。

四、数据同步

在 zookeeper 集群中新的 leader 选举成功之后，leader 会将自身的提交的最大 proposal 的事物 ZXID 发送给其他的 follower 节点。follower 节点会根据 leader 的消息进行回退或者是数据同步操作。最终目的要保证集群中所有节点的数据副本保持一致。