当leader崩溃或者leader失去大多数follower，这时zk进入恢复模式，恢复模式需要重新选举出一个新的leader，让所有的server都恢复到一个正确的状态。zk的选举算法有两种：一种是基于basic paxos实现的，另外一种是基于fast paxos。

1.zookeeper选主流程（basic paxos）

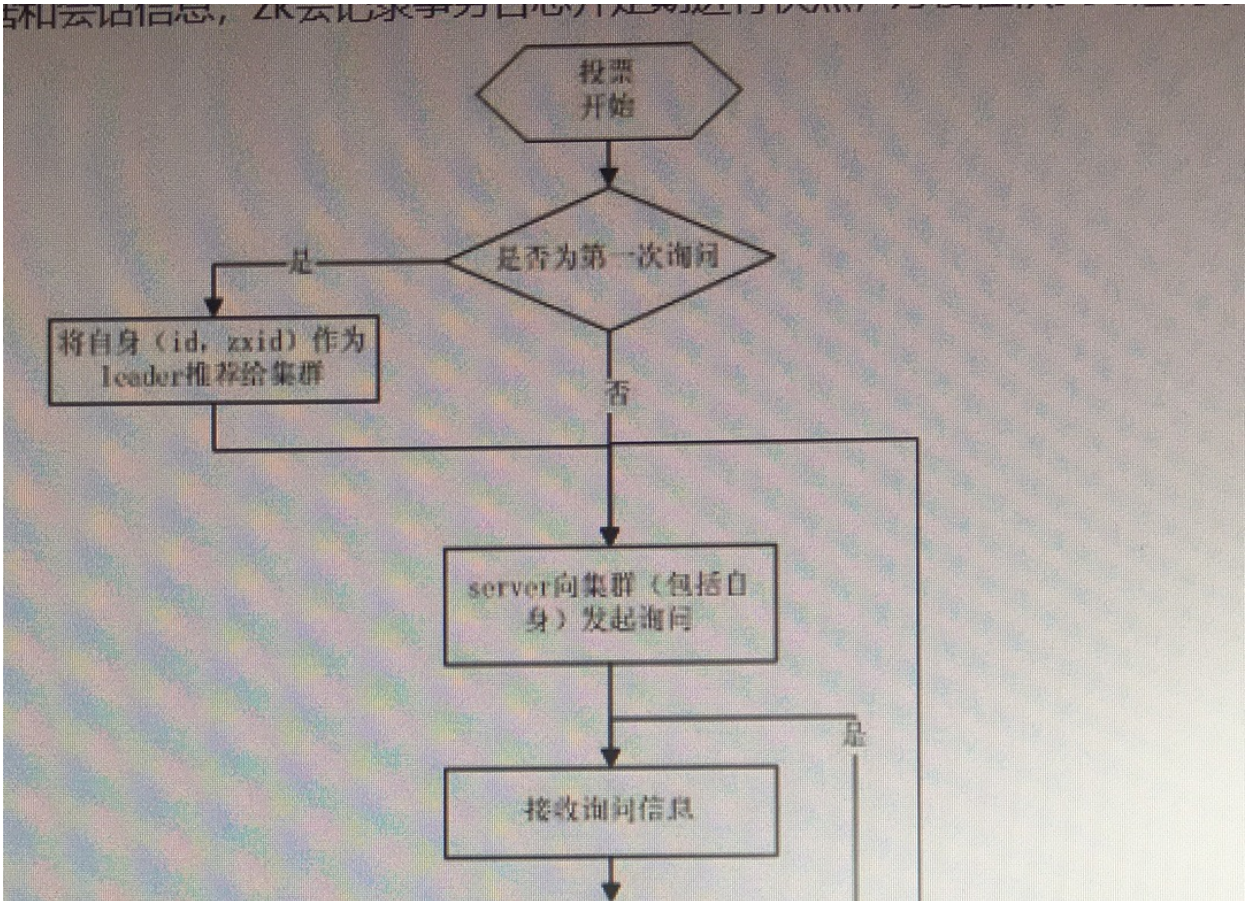
1.1 选举线程由当前server发起选举的线程担任，主要功能是对投票结果进行统计，并选出推荐的server

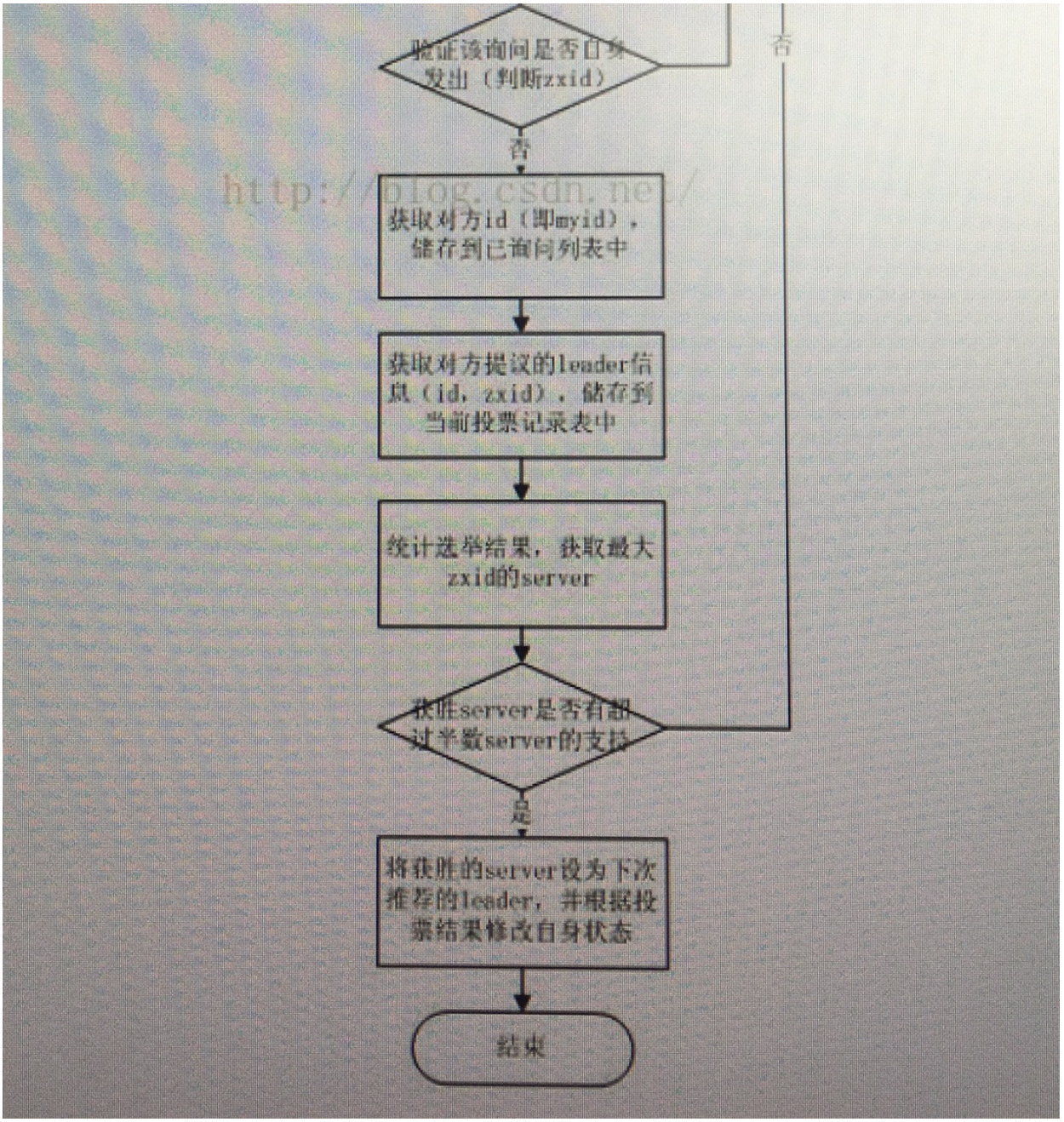
1.2 选举线程首先向所有server发起一次询问（包括自己）

1.3 选举线程收到回复后，验证是否是自己发起的询问（验证zxid是否一致），然后获取对方的id（myid），并存储到当前询问对象列表中，最后获取对方提议的leader相关信息（id，zxid），并将这些信息存储到当次选举的投票纪录表中。

1.4 收到所有server回复以后，就计算出zxid最大的那个server，并将这个server相关信息设置称下一次要投票的server

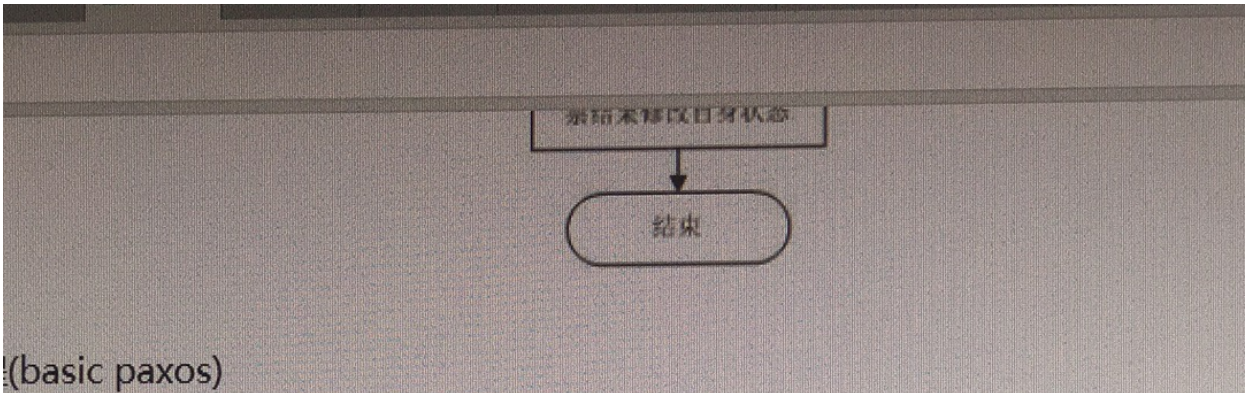
1.5 线程将当前zxid最大的server设置为当前server要推荐的leader，如果此时获胜的server获得n/2 + 1的server票数，设置当前推荐的leader为获胜的server，将根据获胜的server相关信息设置自己的状态，否则，继续这个过程，直到leader被选举出来。通过流程分析我们可以得出：要是leader获得多数server的支持，则server总数必须是奇数2n+1，且存活的server的数目不得少于n+1。每个server启动后都会重复以上流程。在恢复模式下，如果是刚崩溃状态恢复的或者刚启动的server还会从磁盘快照中恢复数据和会话信息，zk会记录事务日志并定期进行快照，方便在恢复时进行状态恢复。



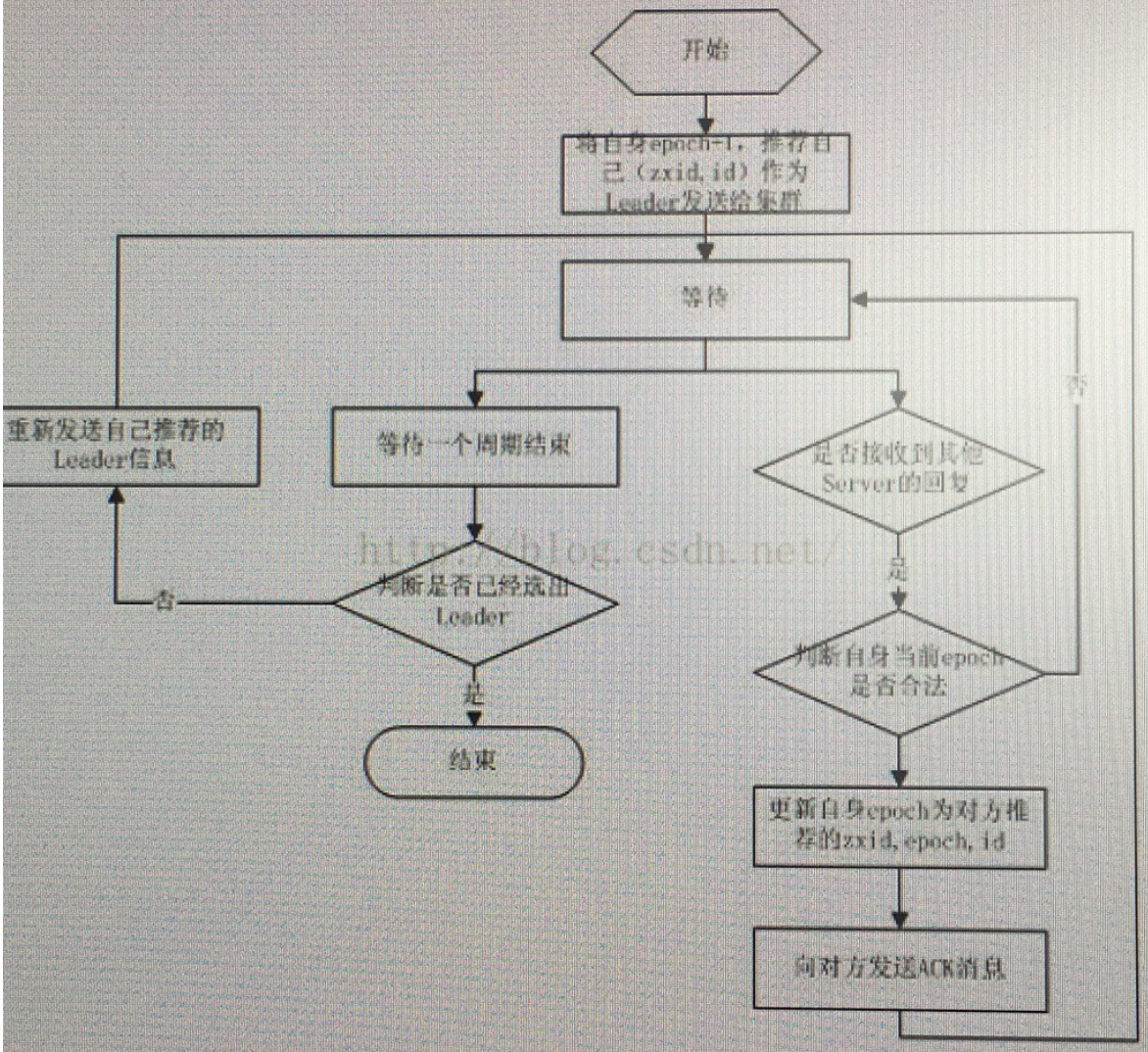


2.zookeeper选主流程

fast paxos流程是在选举过程中，某server首先向所有server提议自己要成为leader，当其它server收到提议以后，解决epoch和zxid的冲突，并接收对方的提议，然后向对方发送接受提议完成的消息，重复这个流程，最后一定能选出一个leader。



过程中，某Server首先向所有Server提议自己要成为leader，当其它Server收到
并接受对方的提议，然后向对方发送接受提议完成的消息，重复这个流程，最后一



r同步流程

入状态同步过程。

接;

将最大的zxid发送给leader;

的zxid确定同步点;