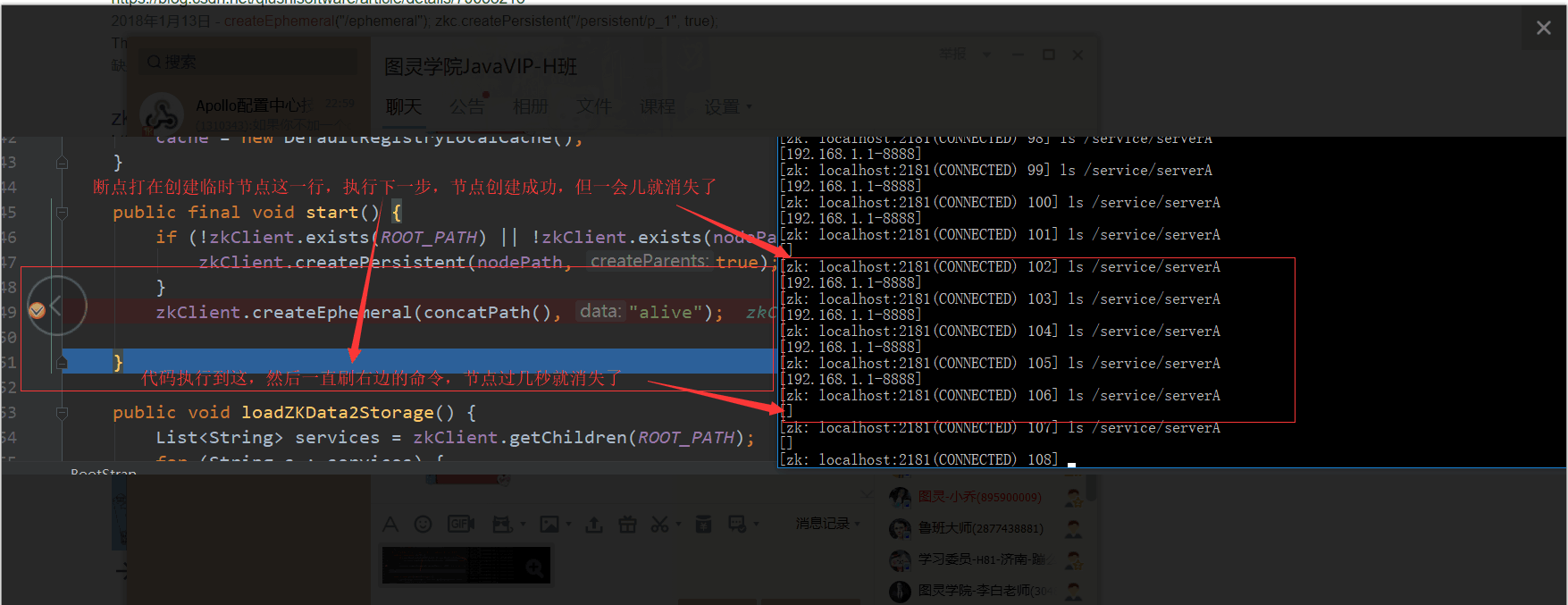
1、但是下面加sleep又正常了，困扰我两天了，帮我问问老师哈（H98）

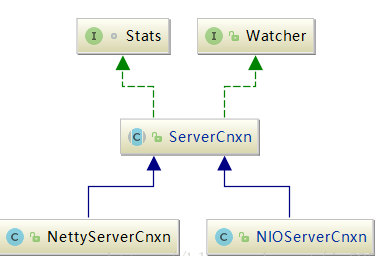
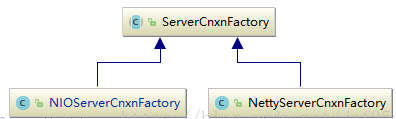


回复：当创建znode的客户端断开连接时，临时节点都会被删除。

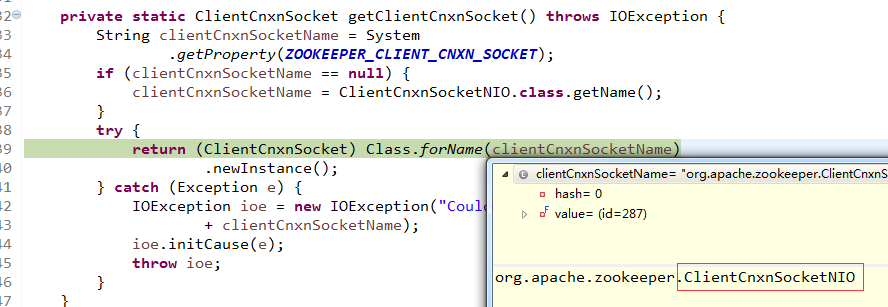
分析：客户端和服务端会建立一个长连接， zk会检查客户端是否存活，如果客户端开断连接（或执行结束），zk会删除临时节点。你这里加上sleep，至少在sleep期间，客户端还没有执行结束，所以在这个时间段临时节点不会被删除，但当sleep结束，且客户端断开连接后，临时节点还是会被删除。

1. （F10）想问个问题，zookeeper的底层原理不是netty吗，但是昨天的课上，服务端底层的启动用的是nio的实现，那是不是zookeeper连接客户端用的是netty，而zookeeper启动时，节点的连接用的时nio呢？

回复：ZooKeeper中使用ServerCnxnFactory管理与客户端的连接，其有两个实现，一个是NIOServerCnxnFactory(默认) ，使用Java原生NIO实现;一个是NettyServerCnxnFactory,使用netty实现;使用ServerCnxn代表一个客户端与服务端的连接



断点结果：



服务端源码分析：https://www.cnblogs.com/siodoon/articles/5438076.html

客户端源码分析：https://blog.csdn.net/cnh294141800/article/details/53039482

1. （H36）

问题1 ：3个节点 1 2 3 ，其实是在3没有启动之前，有两个节点，已经选好了。所以1 2 3 个节点情况下是选2   
回复：

场景一，启动顺序 1 2 3

1、节点1启动，给自己投1票，然后发投票信息，由于其它节点还没有启动，所以它收不到反馈信息，节点1的状态一直是Looking(竞选状态)。

2、节点2启动，给自己投1票，同时与之前启动的节点1交换结果，由于节点2的编号（myid）大于节点1，所以节点2满足胜出的第1个必要条件，且投票数大于半数（2票）（满足胜出的第2个必要条件），所以节点2胜出，节点2的状态为LEADING(领导者状态)，节点1的状态为FOLLOWING(随从状态)。

3、服务器3启动，给自己投1票，同时与之前启动的节点1,2交换信息，由于节点3的编号最大，但之前节点2已经胜出，所以节点3只能是FOLLOWING。

注：交换结果时，myid越大，则获取票数的权重越大

参考：五个节点的选举过程

<https://www.cnblogs.com/ASPNET2008/p/6421571.html>

场景一，启动顺序 3 2 1

参考场景一的过程，当节点3启动时获得1票，节点2启动时又投了1票给节点3（因为节点3的权重大于节点2），此时节点3获得了2票，则满足 投票数大于半数的条件，节点3当选为master。

问题2： zk 的config配置中，是不是要所有节点的选举端口都一样，还是可以不同。只要相应节点的服务器开了那个选集端口就行了

回复：如果是伪集群不能一样(端口冲突)，如果不是伪集群可以一样。

1. （I100）

watcher没反应，为什么和序列化 有关？

答： Zkclient的序列化不能识别控制台操作的序列化方式。

可以这么理解 我在控制台讲的是英文，zkclient只能识别中文，一看不懂就不管了。

2.(L85)

问题：如果一台挂了，只有偶数zk,怎么选出leader，选不出就不服务了吗？

只有2台的时候，挂了一台才不能选，其他的偶数不会的。比如4台 挂了一台 不还有3台嘛，3台不就可以选举嘛。

3.(N15)

1、为什么要选举，随机选一个不行吗

随机选也可以啊，只是老外不喜欢，他们喜欢民主嘛，希望大家投票来选，不喜欢暗箱操作 。等你下次做架构师的时候可以考虑随机选 加油。

1. (I26)

leader 毕竟是单台写，写的量很大的时候。那如何解决 写的压力？

嗯这也是zk的特点也问题所在，所以zk不适合频繁写和海量数据。

1. (L14)

为什么zab能够达到数据一致性

这个我在课上讲了，他是paxos算法的简单版，这个算法就是一个

共识，然后zab的全称是ZooKeeper Atomic Broadcast原子广播大致流程图下图

