

# ZOOKEEPER CLIENT架构 分析——ZK无法重连问题排查

基础研发——张翔

# 问题三段式

What

介绍问题背景，关键点，梳理排查思路

Why

分析zk client和Curator架构，掌握问题原因

How

问题解决方案，以及可行性分析

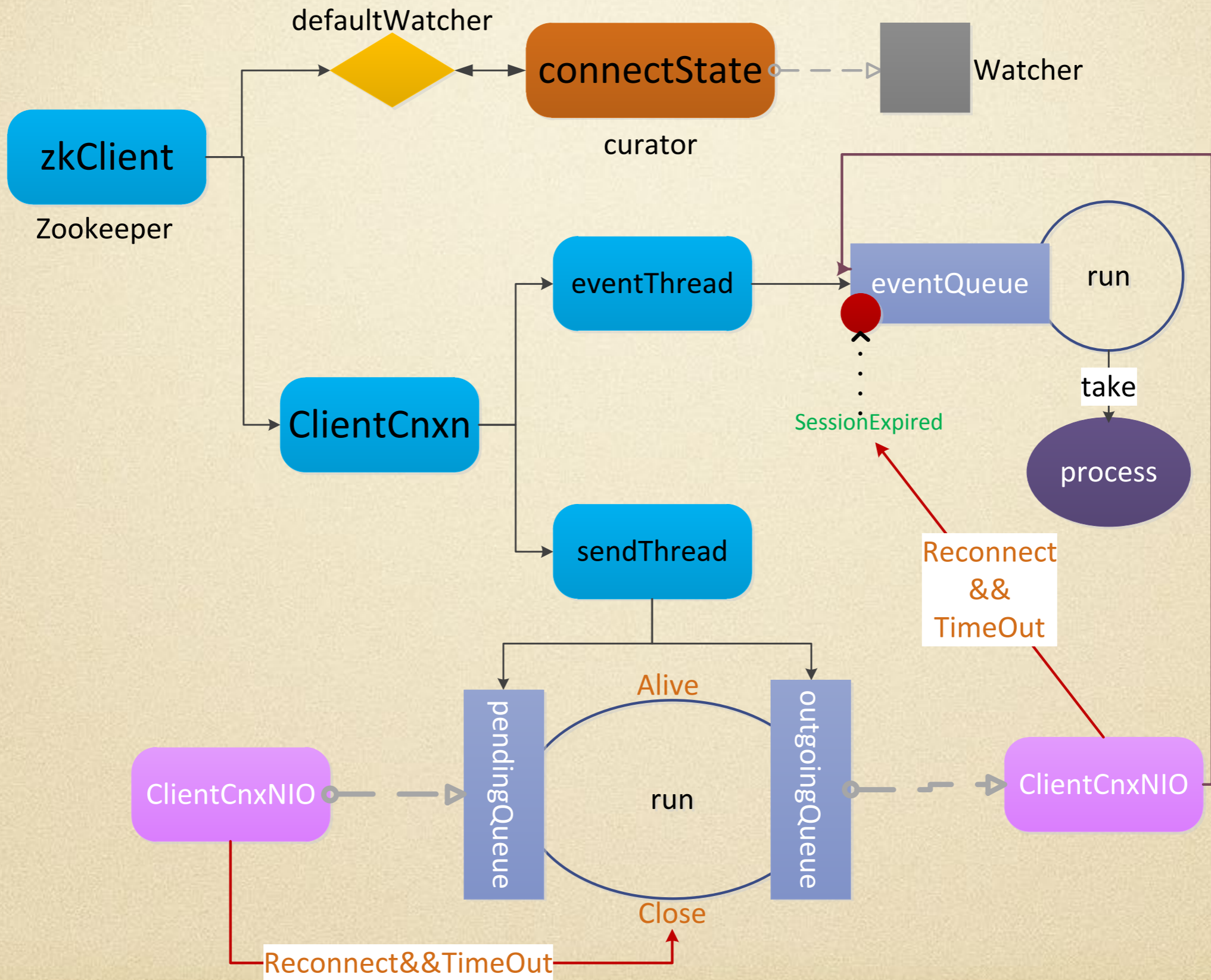
# What: 背景

- mnservice在网络中断一段后，zk链接无法自愈
- 数据量规模小时可以实现zk链接自愈重连
- zk client本身在具有应用层IO重连机制
- mnservice使用了Curator作为工作库进行zk操作

# What: 排查思路分析

- 数据量规模上来后，wather增多；问题是否与reconnect后session恢复相关？
- zookeeper client架构是什么样的，session恢复时进行哪些操作？
- Curator扮演什么角色，参与session恢复吗？

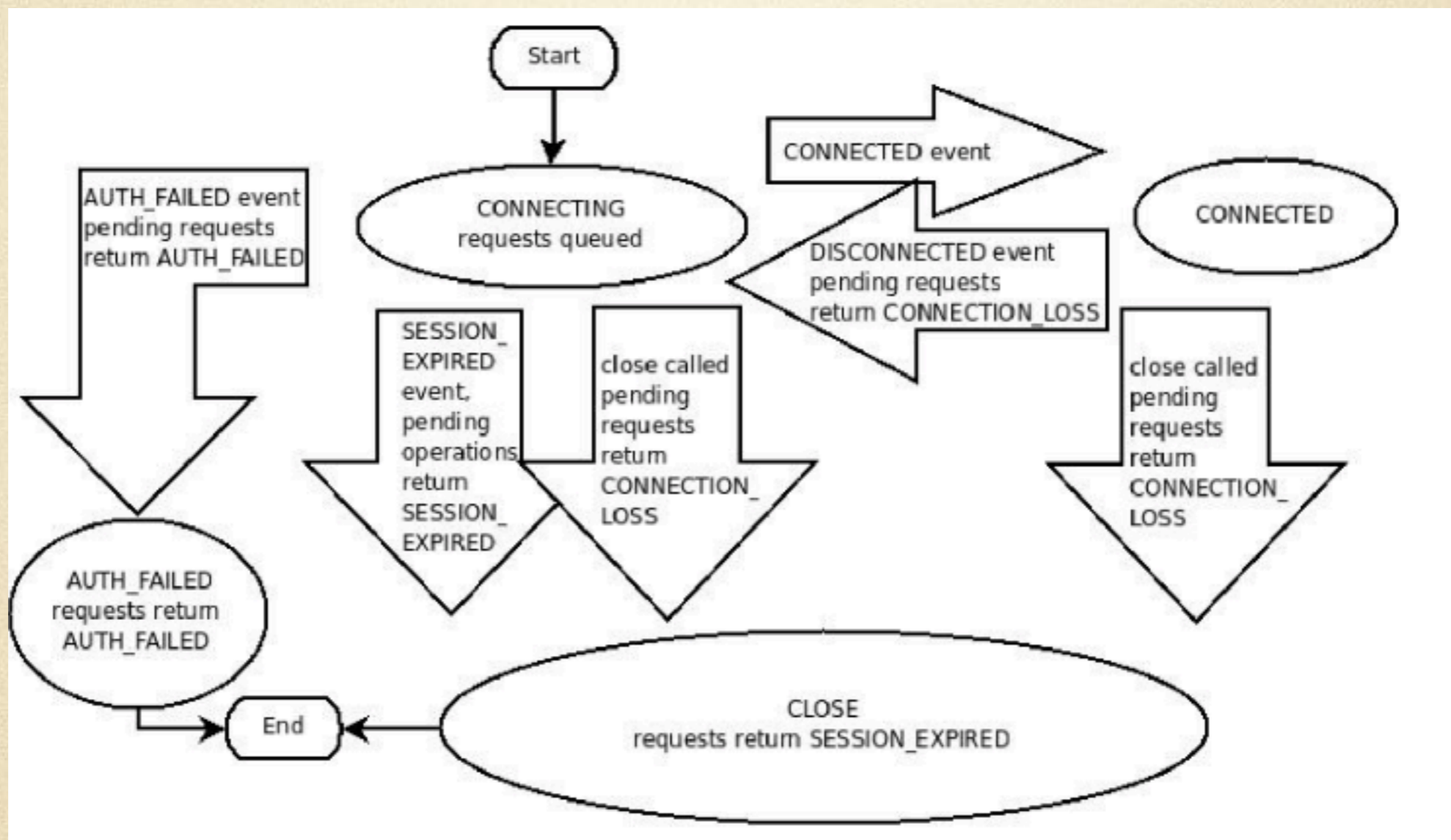
# why: 架构剖析



# What: zk session恢复

- client将Watcher, Auth, sessionId包装成Packet放入sendQueue发送给server
- 网络恢复, NIO重连后触发primeConnect函数
- primeConnect打包多个Packet, 尺寸128KB左右 <1MB, 合法性保证
- Session恢复逻辑, 无问题

# What: sessionExpired



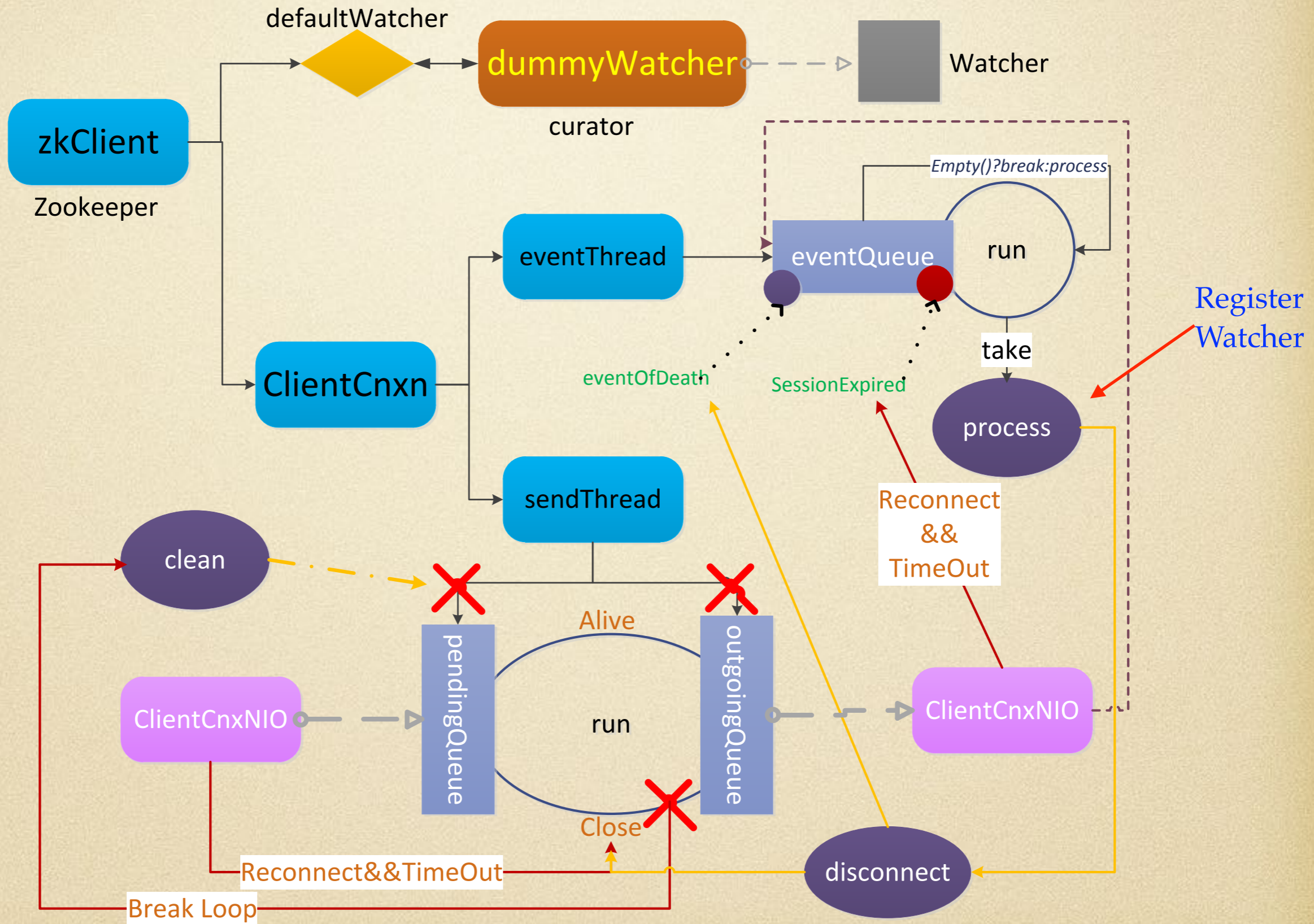
- **SESSION\_EXPIRED:** Zookeeper Client本身无法自愈恢复 session，这是根本原因吗？
- 那为什么Watcher少时，能够自动恢复链接呢？

# Curator的session重建

- session重建：网络恢复，抛出SessionExpired事件，触发默认回调，重新创建zookeeper client
- 疑问点1： watcher增多后这个机制为何无效？
- 疑问点2： zookeeper底层NIO一直在重建网络连接，为什么网络恢复后链接没有重建成功？



# why: session 销毁



# What: Watcher事件类型

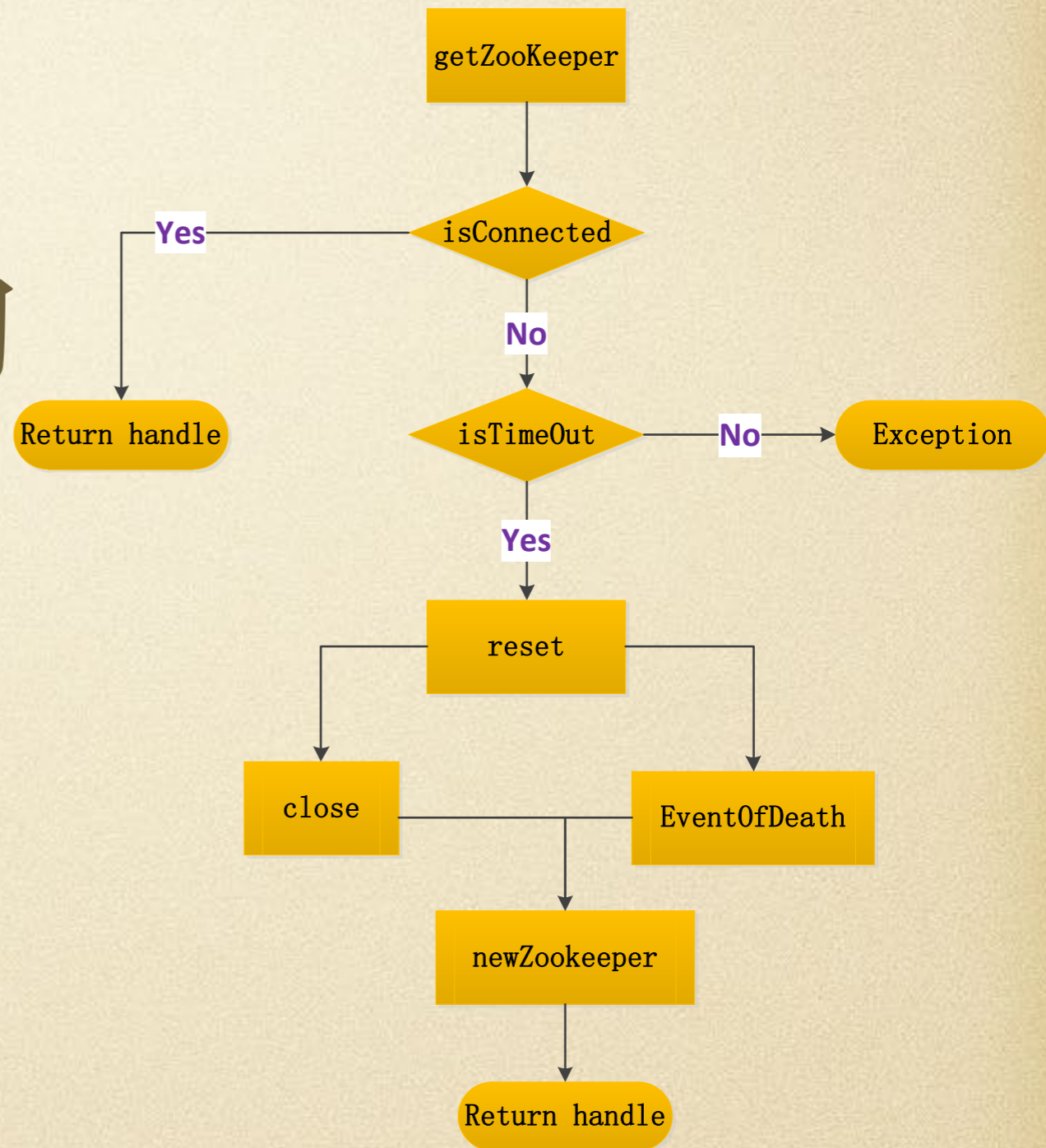
- EventType.Node: 链接相关的状态Disconnected、SyncConnected、AuthFailed、Expired, 所有watcher都会被触发。
- EventType.NodeCreate: 节点创建
- EventType.NodeDeleted: 节点删除
- EventType.NodeDataChanged: 节点数据变动
- EventType.NodeChildrenChanged: 子节点节点变动

# What: 初步结论

- 疑点1解答: watcher register依赖网络通讯或者超时, 同步watcher机制下消费事件队列缓慢, 阻塞默认watcher回调, 无法发起session重建操作
- 疑点2解答: sessionExpired导致主循环退出, 进行clean操作, 关闭socket
- 新问题: 异步watcher下socket能够成功重建, 但存在watcher丢失现象

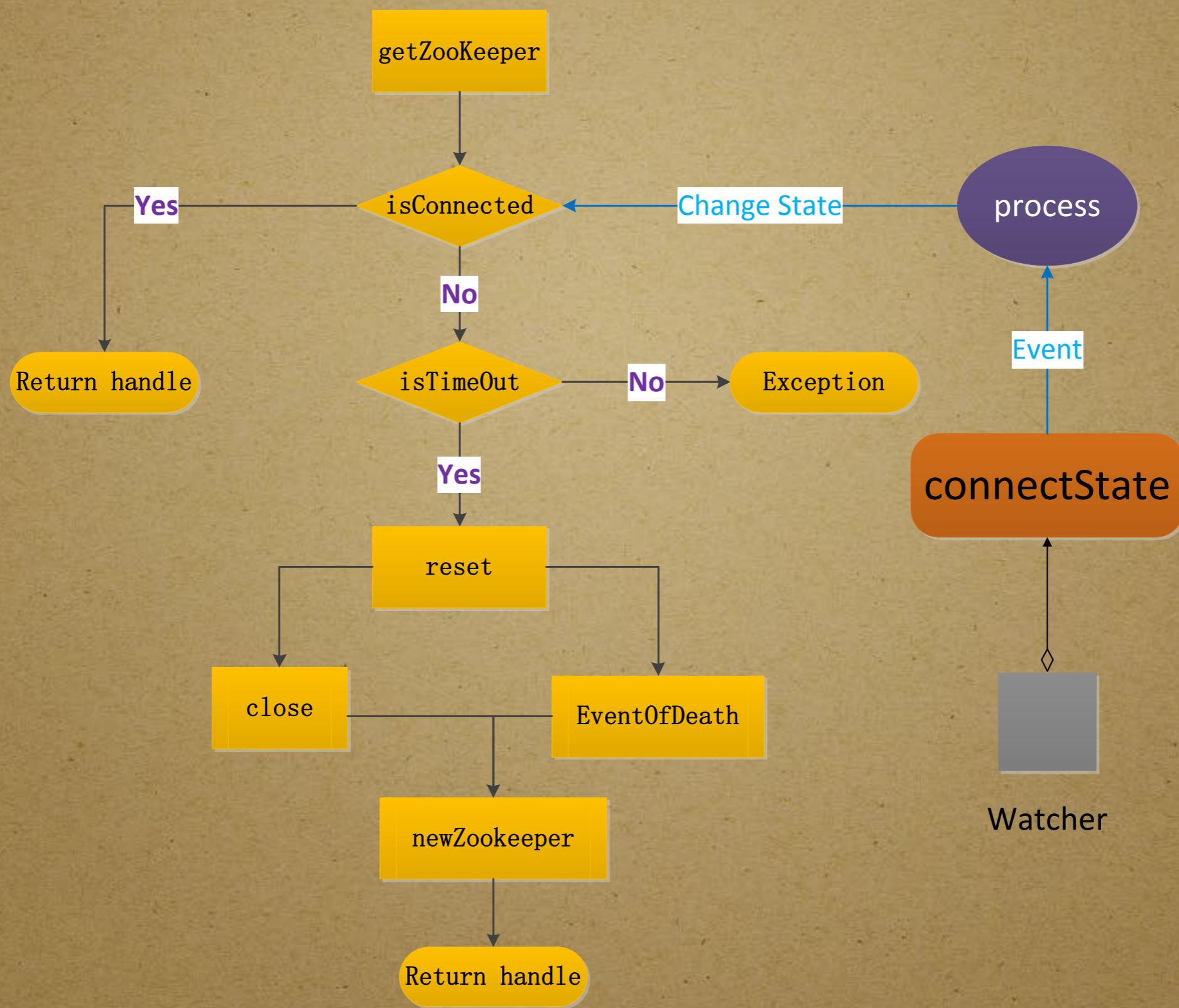
# Curator 主动 销毁session

- mnsync异步线程强同步，拉取zk数据
- 检测connect状态是否断开
- 检测超时，主动发起重建session



# What: 中期结论

- wather丢失: Curator进行zk操作前, 状态检测发现不可用, 主动重建session; 这个session不存  
watchers
- 网络恢复后: NIO直接重连socket, 丢失  
watchers
- 最终疑问🤔: 既然Curator可以主动重建session,  
为什么同步Watcher机制时, 未进行该操作?



isConnected的状态修改依赖默认Watcher的事件触发

# How: 解决思路

- watcher 注册才有异步接口方式，避免阻塞默认 watcher 事件回调
- mnsd 分为两类 zk client，一类负责 getData，另一类负责 addWatcher，避免 watcher 丢失

# Reference

- <http://www.hawkers.cc/2018/01/zookeeper-watcher-reconnect.html>
- <https://github.com/apache/zookeeper>
- <https://zookeeper.apache.org/doc/current/zookeeperProgrammers.html>