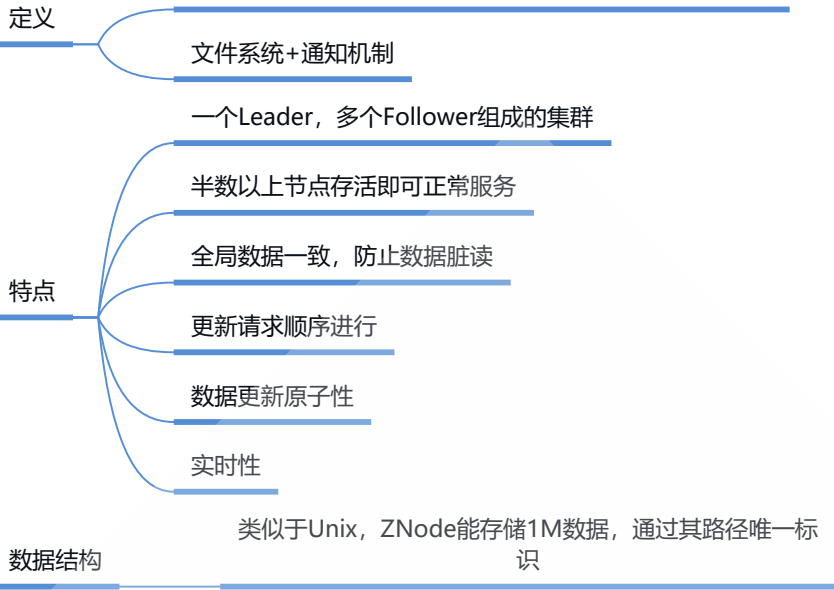


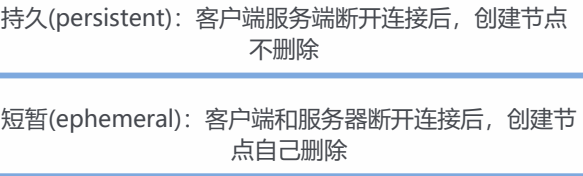
Zookeeper理论文件

Zookeeper内部原理

Zookeeper概述



节点类型



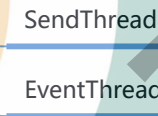
- 1) znode 创建节点的序号 znode&ltbr/>每次修改Zookeeper状态都会收到一个znode形式的消息，也就是Zookeeper事务ID。事务ID是Zookeeper中所有修改操作的序号。每个修改都有唯一的znode，如果znode小于znode，那么znode在znode之前发生。
- 2) ctime - mode 被创建的时间戳(从1970年开始)
- 3) mtime - mode 最后更新的时间戳(从1970年开始)
- 4) mtime - mode 最后更新的子节点znode
- 5) zxid - mode 最后更新的子节点znode
- 6) version - mode 子节点版本号，mode 子节点修改次数
- 7) dataVersion - mode 数据版本号
- 8) aclVersion - mode 访问控制列表的变更号
- 9) ephemeralOwner - 如果是临时节点，这个是mode所有者的session id，如果不是临时节点则是0。
- 10) dataLength - mode 的数据长度
- 11) numChildren - mode 子节点数量

Stat结构体

main() 线程中创建一个Zookeeper客户端



监听器原理



注册监听器列表

注册的监听器

数据或路径变化，发送消息

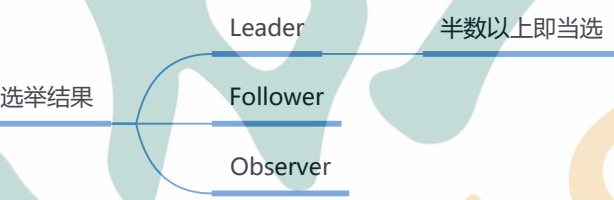
Paxos算法



半数机制

集群中半数以上机器存活，集群可用。所以Zookeeper适合安装奇数台服务器

选举机制

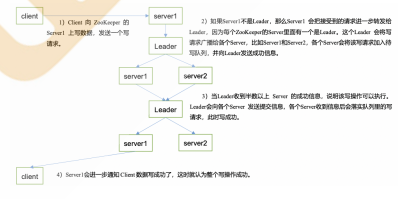


选举机制：ZAB算法（Zookeeper Atomic Boardcast）

写数据流程



流程图



写数据的性能

与集群规模呈负相关

Zookeeper应用模式

- 统一命名服务
- 统一配置管理
- 统一集群管理
- 服务器节点动态上下线
- 软负载均衡