# ZooKeeper: Wait-free coordination for Internet-scale systems

## 中文翻译

## 英文原文

<https://pdos.csail.mit.edu/6.824/papers/zookeeper.pdf>

## 有用资料

<https://blog.csdn.net/wangyangzhizhou/article/details/52698555>

<https://blog.csdn.net/u012935901/article/details/49714893>

<https://www.jianshu.com/p/fb527a64deee>

<https://github.com/brpc/braft/blob/master/docs/cn/zab_protocol.md>

<http://blog.jobbole.com/104985/>

<https://www.cnblogs.com/bangerlee/p/5991417.html>

# Zab协议

## 简介

a leader-based atomic broadcast protocol

## Watch

## 两种基本保证

### 顺序写（Linearizable writes）

### 先进先去（FIFO client order）

## 两种模式

### 恢复模式（选主）

### 广播模式（同步）

## 三种状态

### LOOKING

当前Server不知道leader是谁，正在搜寻

### LEADING

当前Server即为选举出来的leader

### FOLLOWING

leader已经选举出来，当前Server与之同步

# 选主

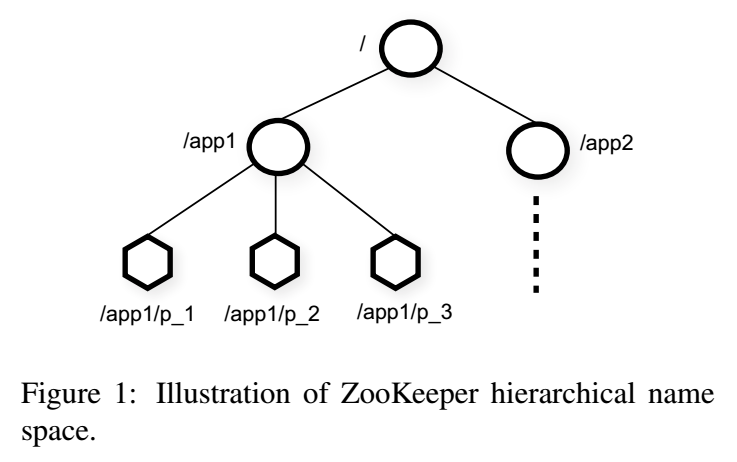
## 参考资料

<https://blog.csdn.net/cnh294141800/article/details/52959028>

<http://www.cnblogs.com/leesf456/p/6107600.html>

# Znode

## 图



## 两种znode

### Regular znode

### Ephemeral znode

没有子节点。

## Sequential flag

# 客户端接口（Client API）

## create(path, data, flags)

## delete(path, version)

## exists(path, watch)

## getData(path, watch)

在节点存在的时候才会设置watch。

## setData(path, data, version)

## getChildren(path, watch)

## sync(path)

# 用途

## 配置管理（Configuration Management）

创建一个znode用于存储配置信息，然后进程启动的时候使用这个znode的路径作为参数，然后进程可以读取这个路径的信息，同时设置watch为true，这样一旦有配置更新，就会被通知。

## 会合（Rendezvous）

创建一个znode，然后把znode的路径作为参数传给master和worker，然后master把自己的地址和端口填充到这个znode，worker可以读取这个znode。

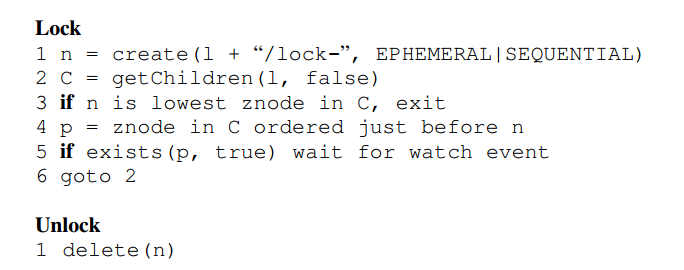
## 组成员关系（Group Membership）

创建一个znode用于放置group信息，每个进程启动之后，在group下创建一个临时znode，设置SEQUENTIAL标志为true，这样group下就由所有的组员。

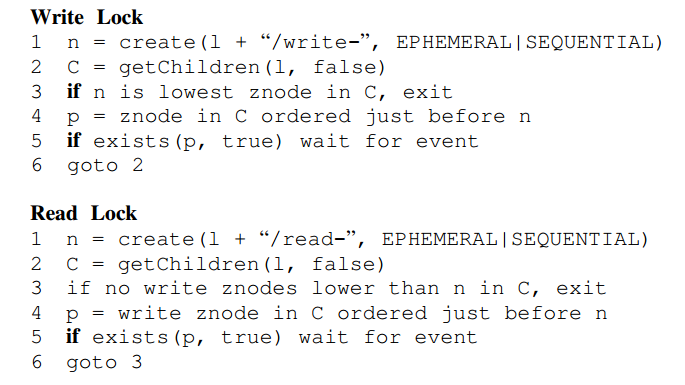
## 简单的锁（Simple Locks ）

客户端在指定位置创建一个znode，设置EPHEMERAL为true，如果创建成功则持有锁。

### Simple Locks without Herd Effect



### Read/Write Locks

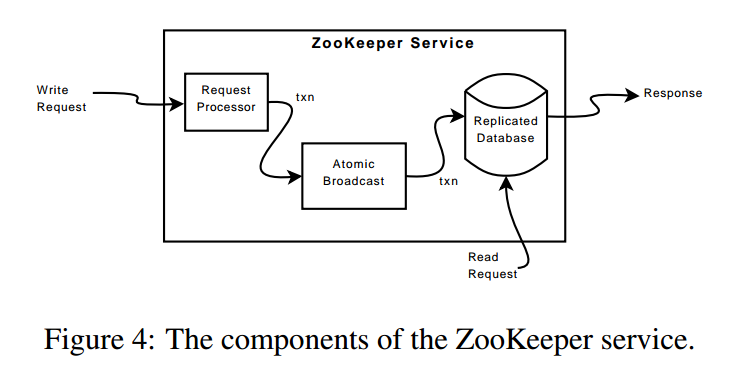


## Double Barrier

创建一个znode作为barrier，绑定一个阀值，每个进程在barrier下创建znode，？？？

# Zookeeper实现

## 简介



读通过本地实现，写通过Leader实现。