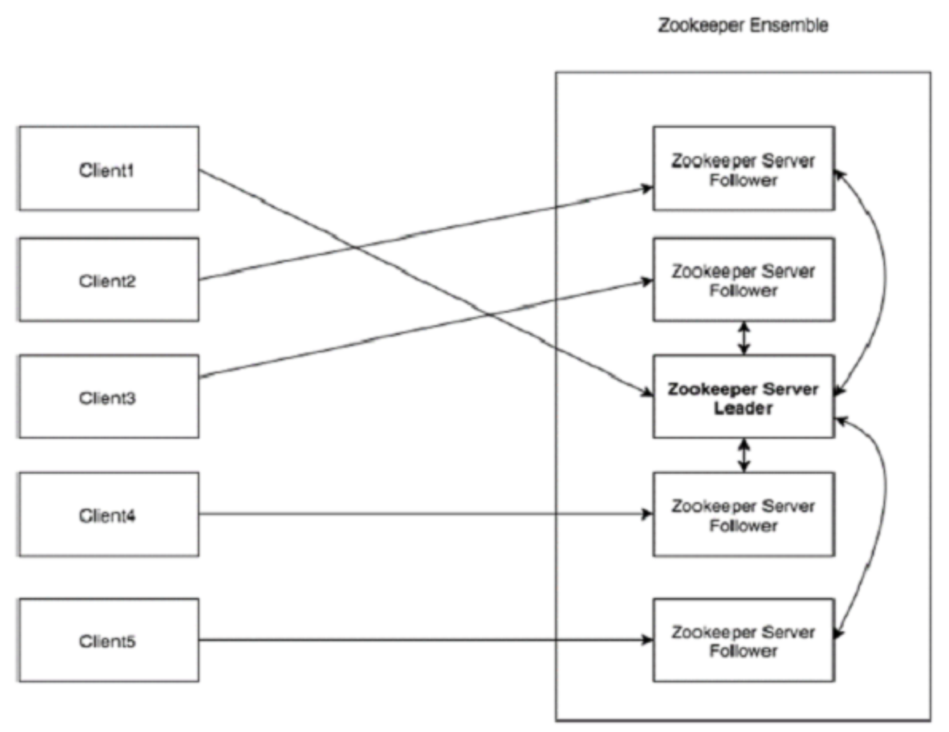
<https://www.w3cschool.cn/zookeeper/zookeeper_fundamentals.html>

## 概述

分布式协调服务，管理大型主机。

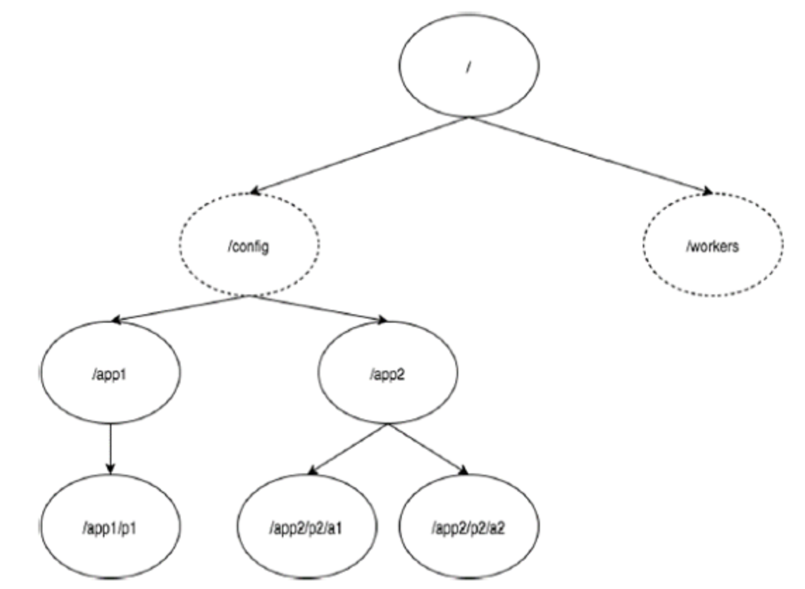
## 基础

### 架构



|  |  |
| --- | --- |
| client | 分布式应用集群中的一个节点，定时发送心跳以告知服务器其存活 |
| server | 服务器，zookeeper总体的一个节点，为客户端提供服务 |
| ensemble | zookeeper服务器组，形成ensemble最少3个节点 |
| leader | 服务启动时选举产生 |
| follower | 跟随leader指令的服务器节点 |

### 层次命名空间



zookeeper节点称为znode，其文件系统树形表示如上图。

/config命名空间用于集中配置管理，/woekers用于命名。

/config下，每个znode最多1M

每个znode存储stat结构，提供znode的元数据：

* 版本号：数据更改时版本号增加
* 访问控制列表：读写等权限
* 数据长度：最多1M

znode类型：

* 持久节点：客户端断开连接不会删除
* 临时节点：断开连接，删除节点，不会有子节点。leader选举中使用
* 顺序节点：可以持久或临时。原始名称+10位序列号。锁定和同步中使用。

### 会话

会话中的请求按照FIFO顺序执行。

### 监视

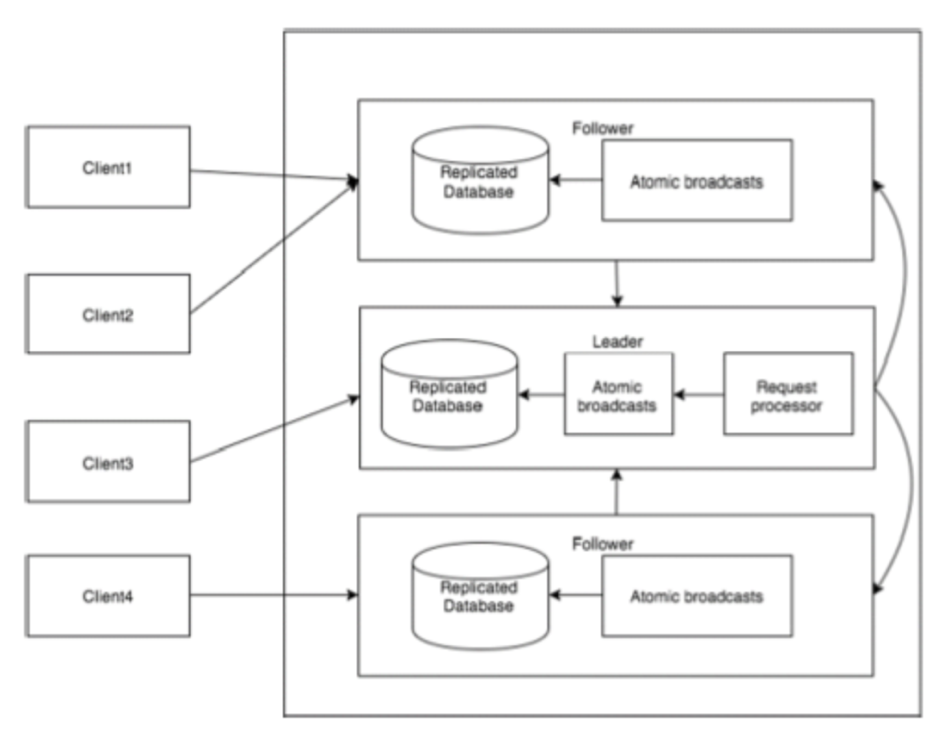
使客户端收到关于zookeeper 集合中更改通知。客户端在读取znode时设置watches。watches会向注册的客户端发送关于znode更改的通知。

znode更改只触发一次watches，如果想再次收到通知，则必须通过另一个读取操作来完成。会话结束，watches删除。

## 工作流

读：从特定znode路径的本地数据库读取数据

写：leader节点将写请求广播到所有节点，如果大多数节点返回成功，则成功，否则失败。大多数节点称为quorum。



## leader选举

* 所有节点创建/app/leader\_election/guid\_的顺序、临时节点。
* 选择顺序号最小的节点成为leader，其他都是follower
* 每个follower节点监视下一个最小数字的znode，例如8监视7，7监视6
* 如果leader关闭，其相应的znode/app/leader\_electionN会被删除
* 下一个follower收到通知，然后检查是否存在其他最小数字的znode
* 其他所有follower选举创建具有最小数字的znode作为leader

## 安装

* java 安装<http://www.oracle.com/technetwork/java/javase/downloads/index.html>
* 移动到/opt/jdk目录下
* 添加路径：./bashrc:export JAVA\_HOME=/opt/jdk/jdk-x.x.x

export PATH=$PATH:$JAVA\_HOME/bin

* source ~/.bashrc
* update-alternatives --install /usr/bin/java java /opt/jdk/jdk1.8.0\_60/bin/java 100
* zk安装：<http://zookeeper.apache.org/releases.html>
* 解压，移动到/opt/zookeeper，配置conf/zoo.cfg中的dataDir
* 启动bin/zkServer.sh start
* 停止bin/zkServer.sh stop
* 进入命令行模式：bin/zkCli.sh