

## 拓展：详解Zookeeper

Zookeeper 是一个用于存储少量数据的基于内存的数据库，

主要有两个核心的概念：文件系统数据结构+监听通知机制。

共有四种节点类型

### 1、PERSISTENT持久化目录节点

客户端与zookeeper断开连接后，该节点依旧存在，只要不手动删除该节点，他将永远存在

### 2、PERSISTENT\_SEQUENTIAL持久化顺序编号目录节点

客户端与zookeeper断开连接后，该节点依旧存在，只是Zookeeper给该节点名称进行顺序编号

### 3、EPHEMERAL临时目录节点

客户端与zookeeper断开连接后，该节点被删除

### 4、EPHEMERAL\_SEQUENTIAL临时顺序编号目录节点

客户端与zookeeper断开连接后，该节点被删除，只是Zookeeper给该节点名称进行顺序编号

5. Container 节点（3.5.3 版本新增，如果Container节点下面没有子节点，则Container节点在未来会被Zookeeper自动清除,定时任务默认60s 检查一次）会自动删除

6. TTL 节点( 默认禁用，只能通过系统配置 zookeeper.extendedTypesEnabled=true 开启，不稳定)会自动过期



## 事件监听机制

客户端注册监听它关心的任意节点，或者目录节点及递归子目录节点

1. 如果注册的是对某个节点的监听，则当这个节点被删除，或者被修改时，对应的客户端将被通知
2. 如果注册的是对某个目录的监听，则当这个目录有子节点被创建，或者有子节点被删除，对应的客户端将被通知
3. 如果注册的是对某个目录的递归子节点进行监听，则当这个目录下面的任意子节点有目录结构的变化（有子节点被创建，或被删除）或者根节点有数据变化时，对应的客户端将被通知。

注意：所有的通知都是一次性的，及无论是对节点还是对目录进行的监听，一旦触发，对应的监听即被移除。递归子节点，监听是对所有子节点的，所以，每个子节点下面的事件同样只会被触发一次。

## Zookeeper经典的应用场景

1. 分布式配置中心
2. 分布式注册中心
3. 分布式锁
4. 分布式队列
5. 集群选举
6. 分布式屏障
7. 发布/订阅