# zookeeper数据模型

**临时节点（有序）、 持久化节点（有序）**

zookeeper是一个开源的分布式协调框架; 数据发布订阅、负载均衡、集群、master选举。。。

原子性： 要么同时成功、要么同时失败 （分布式事务）

单一视图： 无论客户端连接到哪个服务器，所看到的模型都是一样

可靠性：一旦服务器端提交了一个事务并且获得了服务器端返回成功的标识，那么这个事务所引起的服务器端的变更会一直保留

实时性： 近实时

zookeeper并不是用来存储数据的，通过监控数据状态的变化，达到基于数据的集群管理。

# 集群配置

1. 修改zoo.cfg

server.id=ip:port:port 第一个Port 数据同步通信、 第二个port ：leader选举（3181）

id=myid (myid 参与leader选举、 在整个集群中表示唯一服务器的标识)

1. dataDir目录下 创建一个myid的文件 ， 内容： server.id对应当前服务器的id号
2. 如果增加observer

需要在第一步中， server.id=ip:port:port:observer ; peerType=observer

# 会话

NOT\_CONNECTED - > CONNECTING ->CONNECTED ->ClOSE

# 数据模型

数据模型是一个树形结构，最小的数据单元是ZNODE

临时节点和持久化节点

临时有序节点

持久化有序节点

# 状态信息

Stat

cZxid = 0xb0000000f

ctime = Sun Aug 13 20:24:03 CST 2017

mZxid = 0xb0000000f

mtime = Sun Aug 13 20:24:03 CST 2017

pZxid = 0xb0000000f

cversion = 0

dataVersion = 0

aclVersion = 0

ephemeralOwner = 0x15dda30f72f0000

dataLength = 2

numChildren = 0

zab协议 ： 如果客户端发了一个事务请求给到leader， 而leader发送给各个follower以后，并且收到了ack，leader已经commit。 在准备ack给各个follower节点comit的时候，leader挂了，怎么处理的。

1. 选举新的leader（zxid的最大值）
2. 同步给其他的folower

# watcher

EventyType

None 客户端与服务器端成功建立会话

NodeCreated 节点创建

NodeDeleted 节点删除

NodeDataChanged 数据变更：数据内容

NodeChildrenChanged 子节点发生变更： 子节点删除、新增的时候，才会触发

watcher的特性  
一次性触发： 事件被处理一次后，会被移除，如果需要永久监听，则需要反复注册

zkClient （ 永久监听的封装）

curator

java api的话， zk.exists , zk.getData 创建一个watcher监听

zookeeper序列化使用的是Jute

# Acl权限的操作

保证存储在zookeeper上的数据安全性问题

schema(ip/**Digest**/world/super)  
授权对象（192.168.1.1/11 , root:root / world:anyone/ super）

# 数据存储

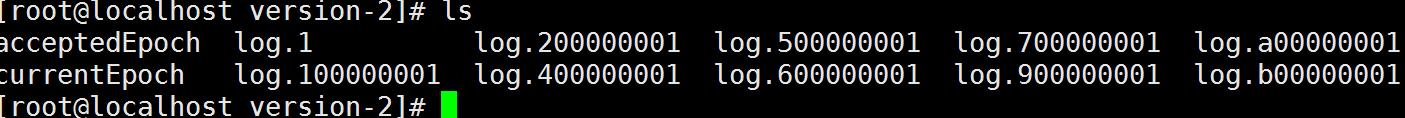
内存数据和磁盘数据

zookeeper会定时把数据存储在磁盘上。

DataDir = 存储的是数据的快照

快照： 存储某一个时刻全量的内存数据内容

DataLogDir 存储事务日志



log.zxid

查看事务日志的命令

java -cp :/mic/data/program/zookeeper-3.4.10/lib/slf4j-api-1.6.1.jar:/mic/data/program/zookeeper-3.4.10/zookeeper-3.4.10.jar org.apache.zookeeper.server.LogFormatter log.200000001

zookeeper 有三种日志

zookeeper.out // 运行日志

快照 存储某一时刻的全量数据

事务日志 事务操作的日志记录

# Dubbo

<https://dubbo.incubator.apache.org/>

dubbo+spring boot +docker

## dubbo能解决什么问题

1. 怎么去维护url

通过注册中心去维护url（zookeeper、redis、memcache…）

1. F5硬件负载均衡器的单点压力比较大

软负载均衡

1. 怎么去整理出服务之间的依赖关系。

自动去整理各个服务之间的依赖

1. 如果服务器的调用量越来越大，服务器的容量问题怎么去评估，扩容的指标

需要一个监控平台，可以监控调用量、响应时间

## Dubbo是什么

dubbo是一个分布式的服务框架，提供高性能的以及透明化的RPC远程服务调用解决方法，以及SOA服务治理方案。

Dubbo的核心部分：

远程通信

集群容错

服务的自动发现

负载均衡

# Dubbo的架构

核心角色

Provider 生产者

Consumer 消费者

Registry 注册器

Monitor 监控

Container 容器

一台高性能的刀片机

32G内存 16核心

虚拟化

4G 2核心 – 8台

PAAS（platform-as-a-service）/ IAAS(infrastucturre-as-a-service)/SAAS(软件即服务)

kvm vm window server