上一篇教程我们讲解了分布式的基础，同时ZooKeeper的单机安装和集群搭建（配置observer节点），以及集群中每个节点的角色，接下来讲解的是

一、zoo.cfg配置文件

（一）常规的配置

tickTime=2000 zookeeper中最小的时间单位长度 （ms）

initLimit=10 follower节点启动后与leader节点完成数据同步的时间

syncLimit=5 leader节点和follower节点进行心跳检测的最大延时时间

dataDir=/tmp/zookeeper 表示zookeeper服务器存储快照文件的目录

dataLogDir 表示配置 zookeeper事务日志的存储路径，默认指定在dataDir目录下

clientPort 表示客户端和服务端建立连接的端口号： 2181

下面是一个zoo.cfg的配置文件

```
# The number of milliseconds of each tick
tickTime=2000
# The number of ticks that the initial
# synchronization phase can take
initLimit=10
# The number of ticks that can pass between
# sending a request and getting an acknowledgement
syncLimit=5
# the directory where the snapshot is stored.
# do not use /tmp for storage, /tmp here is just
# example sakes.
dataDir=/tmp/zookeeper
# the port at which the clients will connect
clientPort=2181
# the maximum number of client connections.
# increase this if you need to handle more clients
#maxClientCnxns=60
#
# Be sure to read the maintenance section of the
# administrator guide before turning on autopurge.
#
# http://zookeeper.apache.org/doc/current/zookeeperAdmin.html#sc_maintenance
#
# The number of snapshots to retain in dataDir
#autopurge.snapRetainCount=3
# Purge task interval in hours
# Set to "0" to disable auto purge feature
#autopurge.purgeInterval=1
#集群配置
server.1=192.168.190.101:2888:3881
server.2=192.168.190.102:2888:3881
server.3=192.168.190.105:2888:3881
server.4=192.168.190.104:2888:3881:observer

~
```

（二）高级配置

二、ZooKeeper中的一些概念

1、数据模型

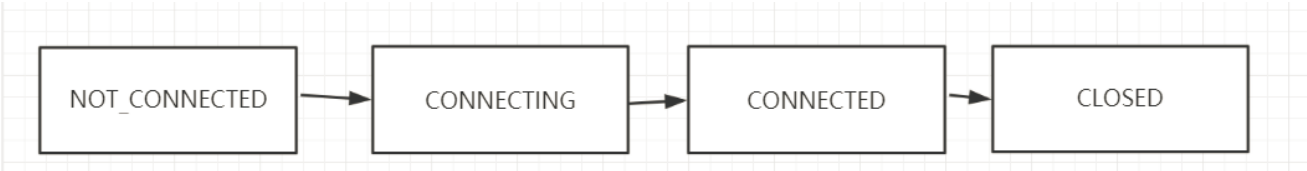
zookeeper的数据模型和文件系统类似，每一个节点称为：znode. 是zookeeper中的最小数据单元。每一个znode上都可以

保存数据和挂载子节点。 从而构成一个层次化的属性结构

2、集群角色：前面已经介绍过（leader、follower、observer）

3、会话

客户端和服务器之间的一次TCP长连接



4、数据节点

持久化节点： 节点创建后会一直存在zookeeper服务器上，直到主动删除

持久化有序节点：每个节点都会为它的一级子节点维护一个顺序

临时节点： 临时节点的生命周期和客户端的会话保持一致。当客户端会话失效，该节点自动清理

临时有序节点： 在临时节点上多勒一个顺序性特性

5、版本

ZooKeeper的每个节点上都会存储数据，对应于每个ZNode，ZooKeeper都会维护一个叫Stat的数据结构，Stat记录了ZNode的三个数据版本，分别是：

cversion = 0 子节点的版本号

aclVersion = 0 表示acl的版本号，修改节点权限

dataVersion = 1 表示的是当前节点数据的版本号

6、watcher

zookeeper提供了分布式数据发布/订阅,zookeeper允许客户端向服务器注册一个watcher监听。当服务器端的节点触发指定事件的时候

会触发watcher。服务端会向客户端发送一个事件通知

watcher的通知是一次性，一旦触发一次通知后，该watcher就失效

7、ACL（Access Cotrol Lists）

zookeeper提供控制节点访问权限的功能，用于有效的保证zookeeper中数据的安全性。避免误操作而导致系统出现重大事故。zookeeper提供了world、auth、digest、ip和supper的模式

CREATE /READ/WRITE/DELETE/ADMIN

CREATE：创建子节点的权限

READ：获取节点数据和子节点列表的权限

WRITE：更新节点数据的权限

DELETE：删除子节点的权限

ADMIN：设置节点ACL权限

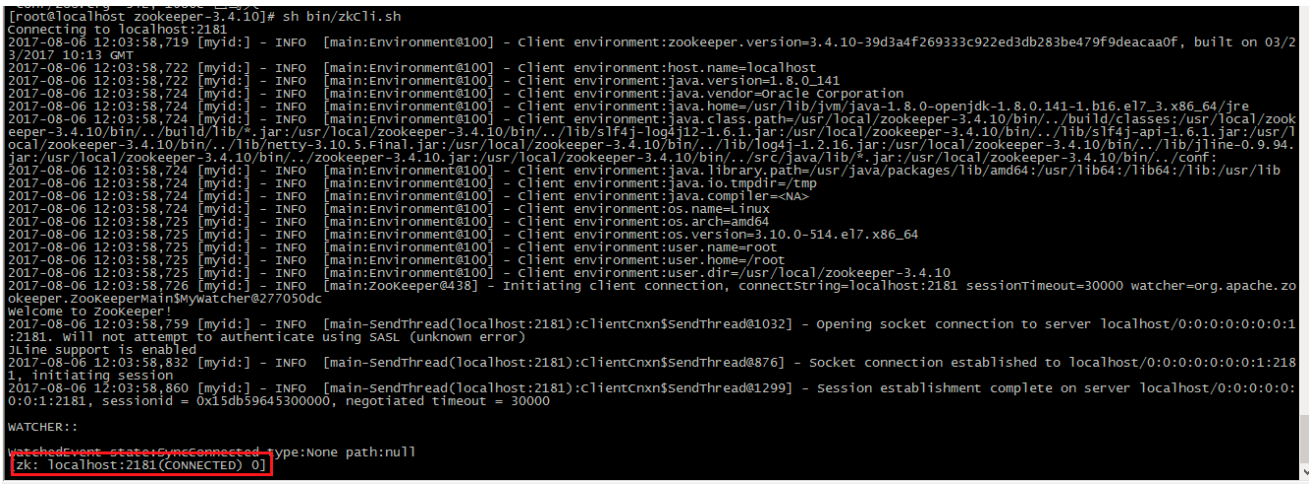
三、ZooKeeper的命令操作

在{ZOOKEEPER_HOME}目录，连接zkCli客户端，命令格式如下

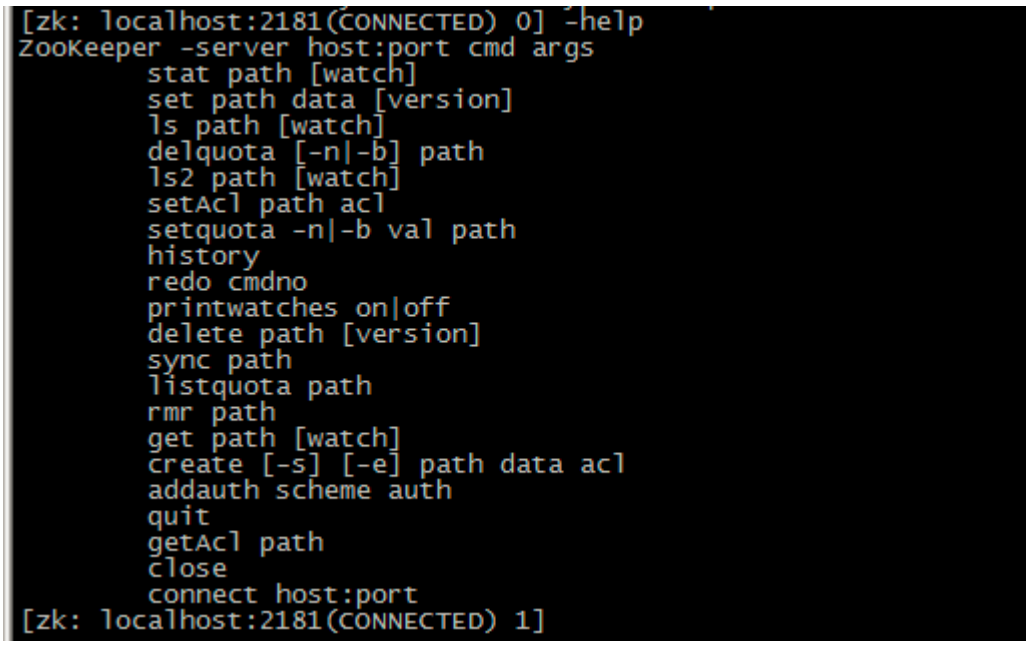
```
sh bin/zkCli.sh -server ip:port
```

现在连接到本机

sh bin/zkCli.sh



键入-help查看有哪些命令



1、create [-s][-e] path data acl

-s 表示节点是否有序

-e 表示是否为临时节点

默认情况下，是持久化节点

演示：

默认创建node节点

```
create /node1``create /node1/node1-1``create /node1/node1-2
```

创建有序节点

```
create -s /node2
```

创建临时节点

```
create -e /node_tmp1
```

会话关闭之后，再重新连接，发现临时节点没有了

2、get path [watch]

获得指定 path的信息

3、set path data [version]

修改节点的信息

修改节点 path对应的data

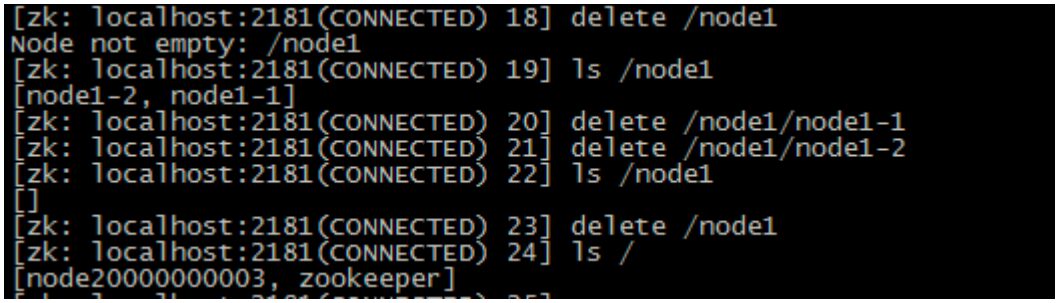
乐观锁的概念

数据库里面有一个 version 字段去控制数据行的版本号

`#修改node1的值为mic``set /node1 mic````#查看修改后的值``get /node1`

4、delete path [version]

删除节点，它是从叶子节点开始删除，不能删除非空节点



四、stat信息

cversion = 0 子节点的版本号

aclVersion = 0 表示acl的版本号，修改节点权限

dataVersion = 1 表示的是当前节点数据的版本号

czxid 节点被创建时的事务ID

mzxid 节点最后一次被更新的事务ID

pzxid 当前节点下的子节点最后一次被修改时的事务ID

ctime = Sat Aug 05 20:48:26 CST 2017

mtime = Sat Aug 05 20:48:50 CST 2017

下面是一个节点的stat信息

cZxid = 0x500000015

ctime = Sat Aug 05 20:48:26 CST 2017

mZxid = 0x500000016

mtime = Sat Aug 05 20:48:50 CST 2017

pZxid = 0x500000015

cversion = 0

dataVersion = 1

aclVersion = 0

ephemeralOwner = 0x0 创建临时节点的时候，会有一个sessionId 。 该值存储的就是这个sessionid

dataLength = 3 数据值长度

numChildren = 0 子节点数

五、Java操作ZooKeeper

（一）Java API的使用

1、导入jar包

```
<dependency>
    <groupId>org.apache.zookeeper</groupId>
    <artifactId>zookeeper</artifactId>
    <version>3.4.10</version>
</dependency>
```

2、代码演示

GitHub地址为：

（二）zkClient

（三）curator