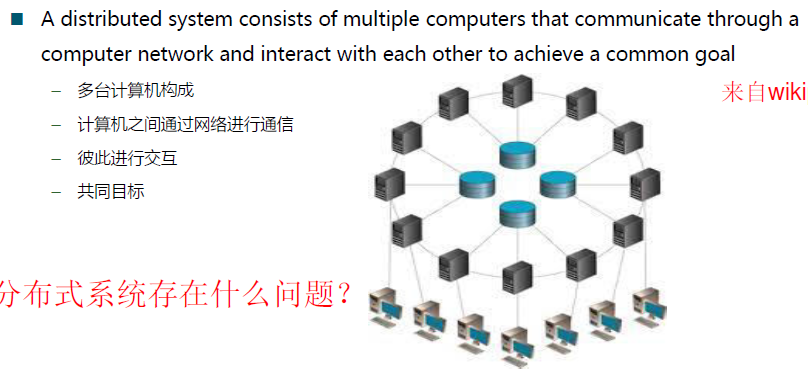
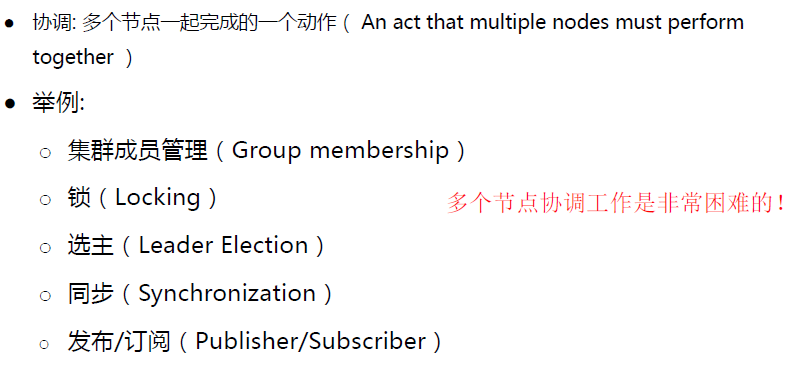
Zookeeper

# 基本介绍

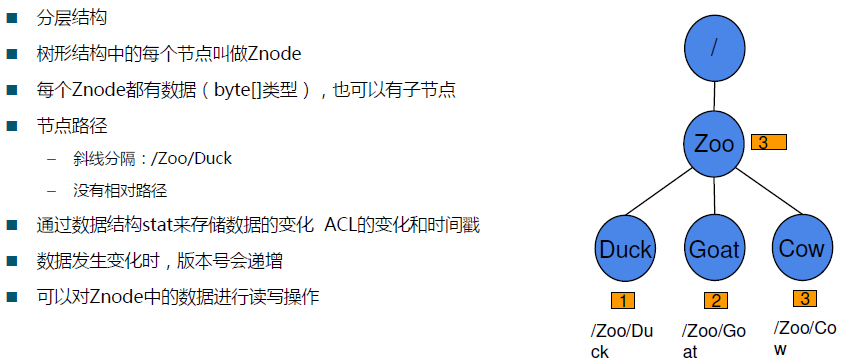
Zookeeper是一个高性能，分布式的，开源分布式应用协调服􀔇。它提供了简单原始的功能，分布式应用可以基于它实现更高级的服务，比如同步，配置管理，集群管理，名空间。它被设计为易于编程，使用文件系统目录树作􀐞数据模型。服务端跑在java，提供java和C的客户端 API。



## zookeeper 分布式系统协调



## 数据模型



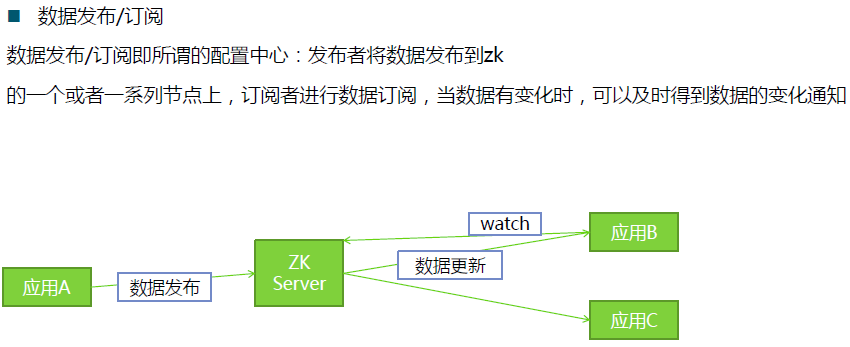
所有操作都是基于绝对路径来操作，没有相对路径的概念。

数据有版本号

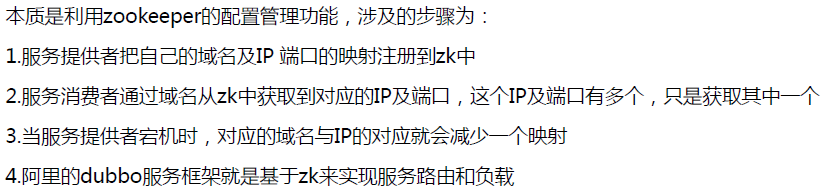
## 典型应用场景

ZK 和应用B/C之间是长连接。 ZK本身是一个集群，高可用。

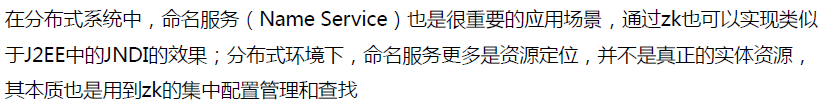
### 数据发布/订阅



### 负载均衡



### 命名服务



### 分布式协调/通知

– 通过watcher和通知机制实现

– 分布式锁

– 分布式事务

### 集群管理

– 当前集群中的机器数量

– 集群中机器的运行时状态

– 集群中节点的上下线操作

– 集群节点的统一配置

### master 选举

– 临时节点

– 顺序节点

### 分布式锁

排他锁

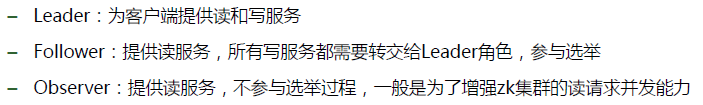
共享锁

### 分布式队列

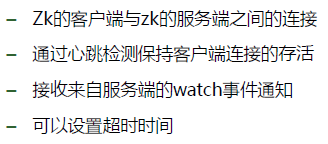
FIFO

## 基本概念

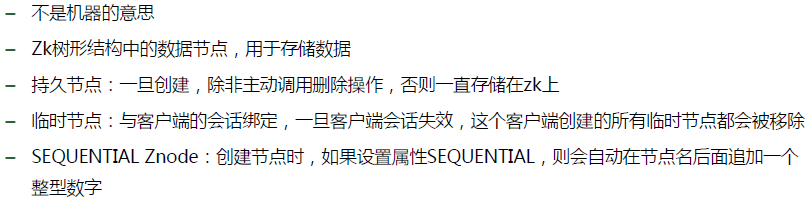
### 角色



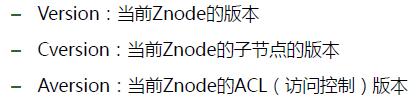
### 会话



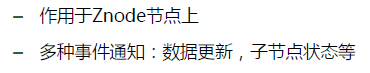
### 数据节点



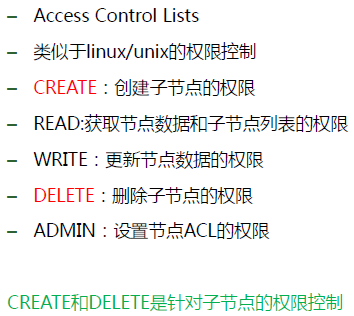
### 版本



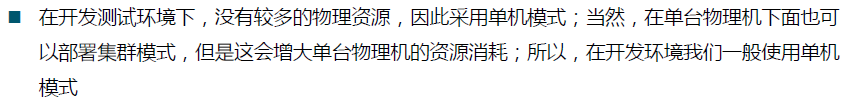
### Watcher



### ACL



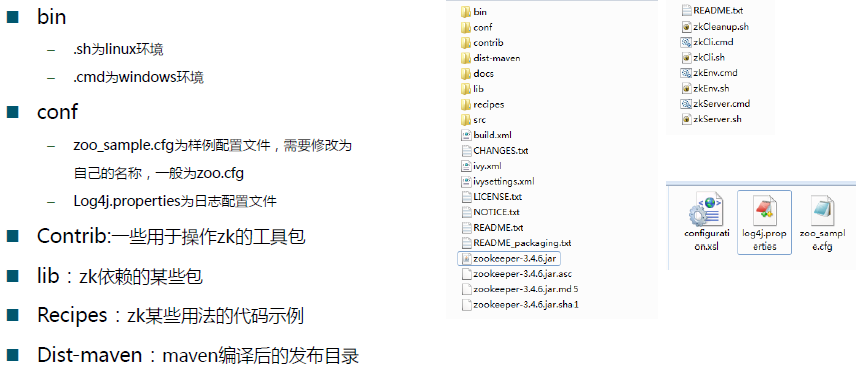
## 单机模式

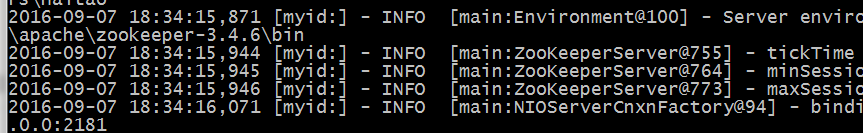
### 安装



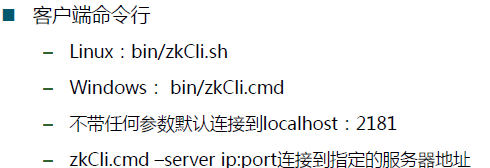
### 目录结构

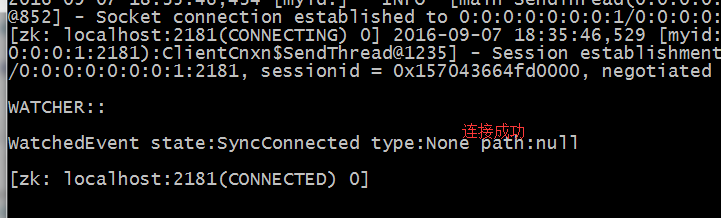


### 运行配置

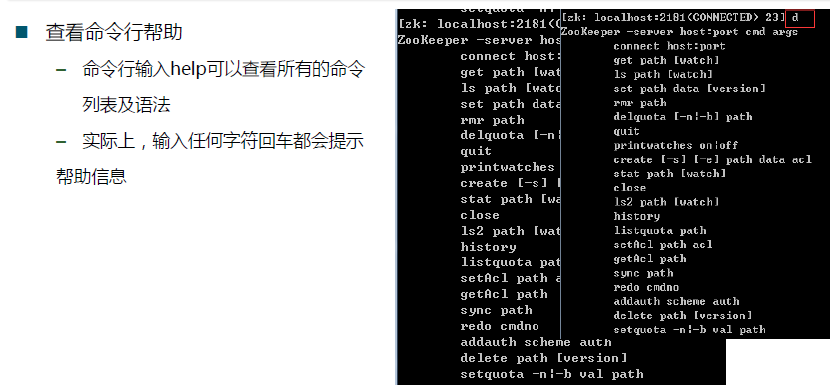
 

### 客户端连接

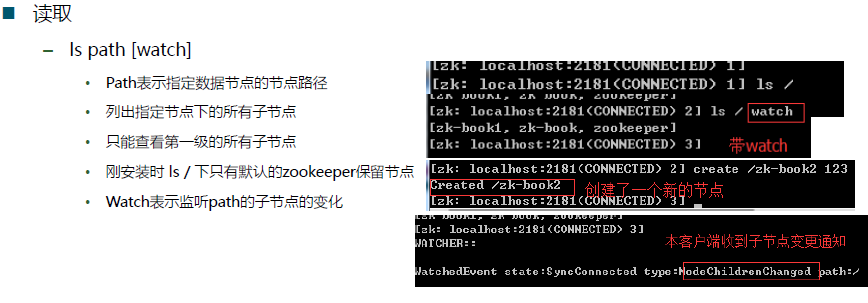




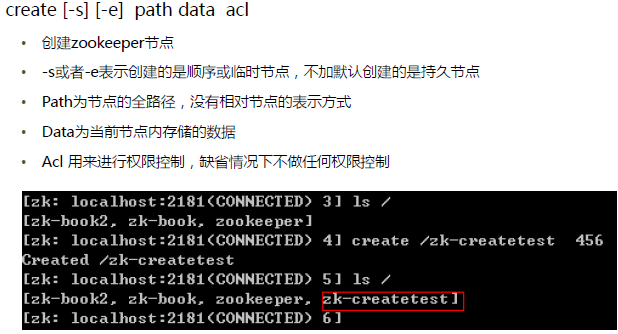
### 客户端命令



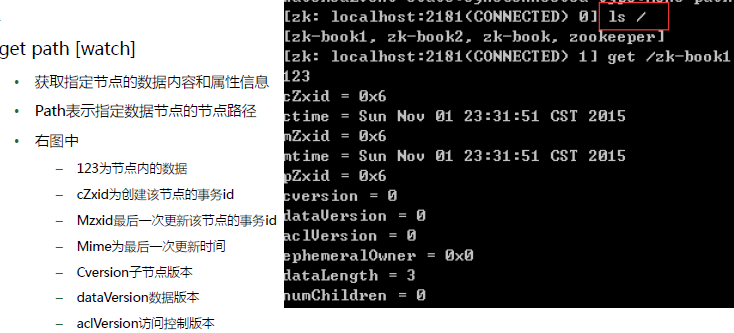
#### 读取



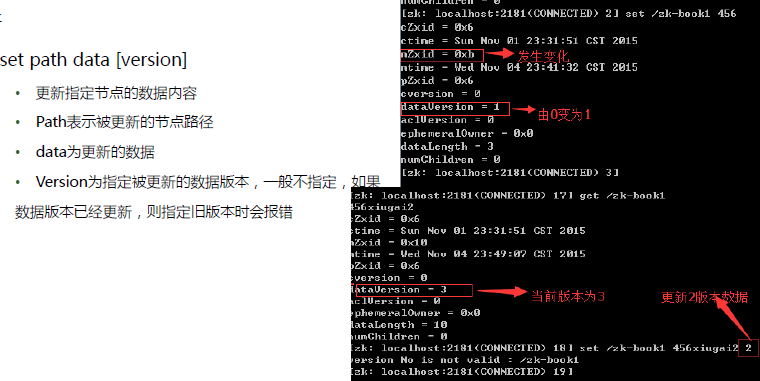
#### 创建



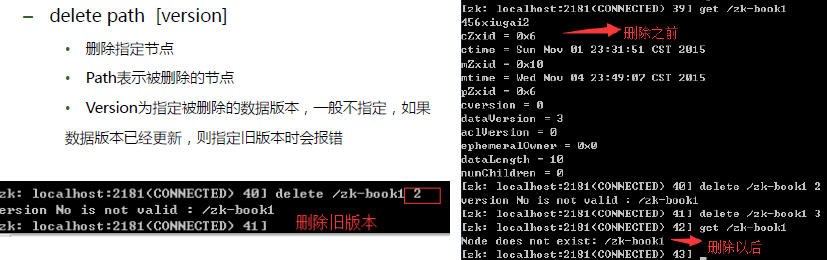
#### 读取



#### 更新

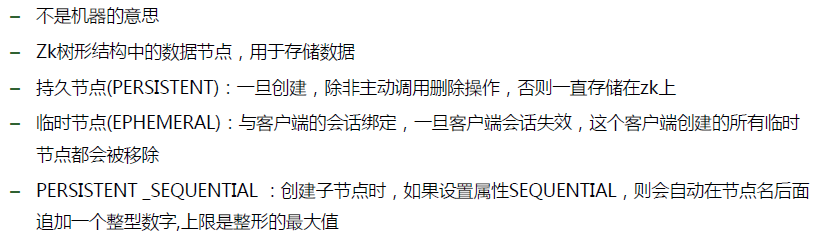


#### 删除



# ZK 进阶

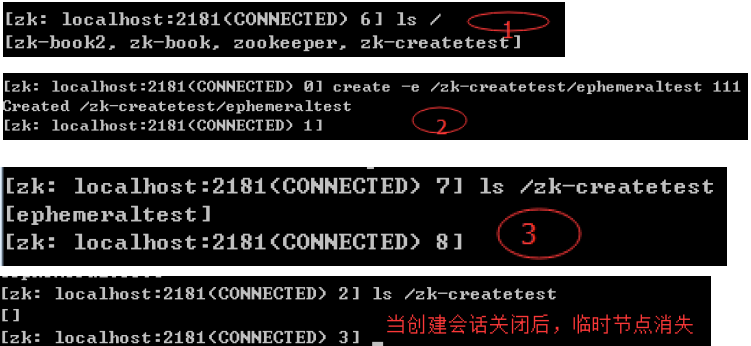
## 数据节点 Znode



### 临时节点

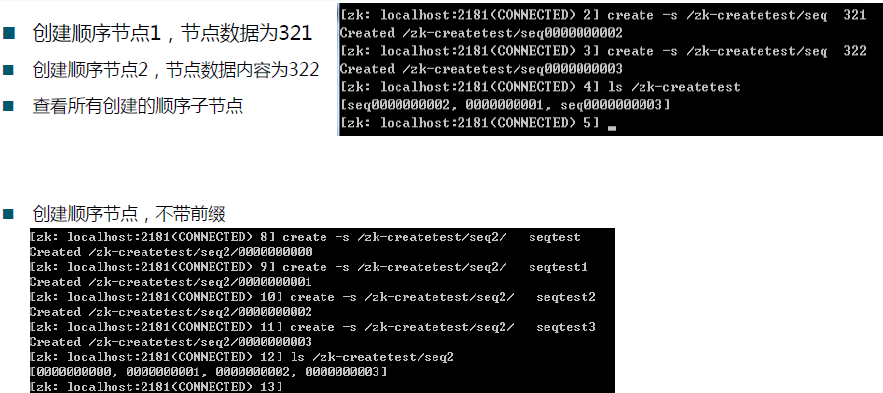
与客户端的会话绑定。

可以根据临时节点的特点判断某个服务是否存活。 创建时加参数-e



### 顺序节点

创建时加参数 –s



## Watcher

