

# 背景

随着缓存数量的增加，单服务器无法承载的时候，就出现了redis-cluster，将数据分区放到不同的服务器中，用分布式的缓存来承接高并发。

## 核心原理

redis cluster 实现数据分区

分布式缓存节点之间的通信

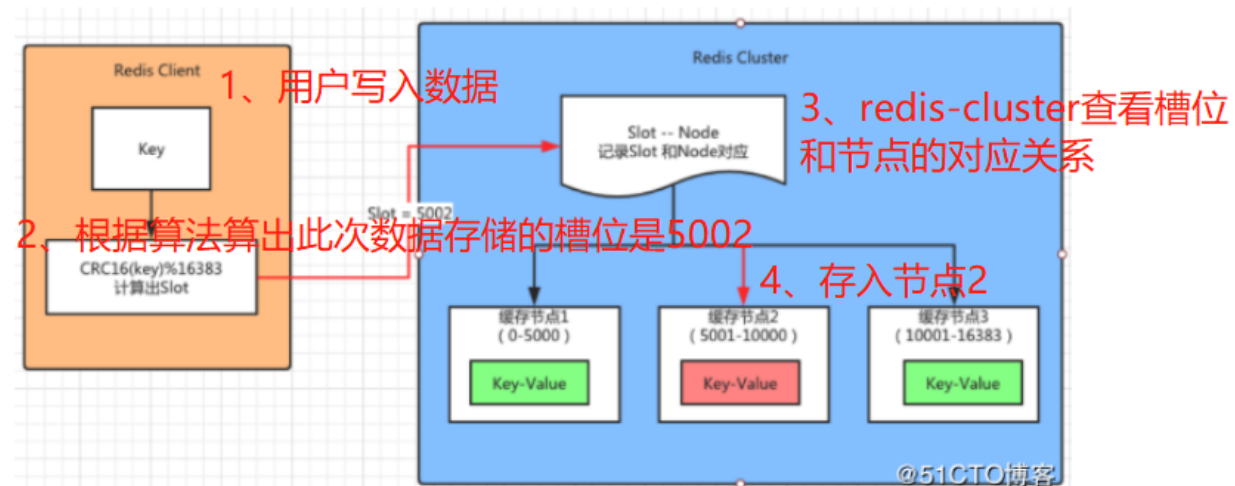
请求分布式缓存的路由

缓存节点的扩展收缩

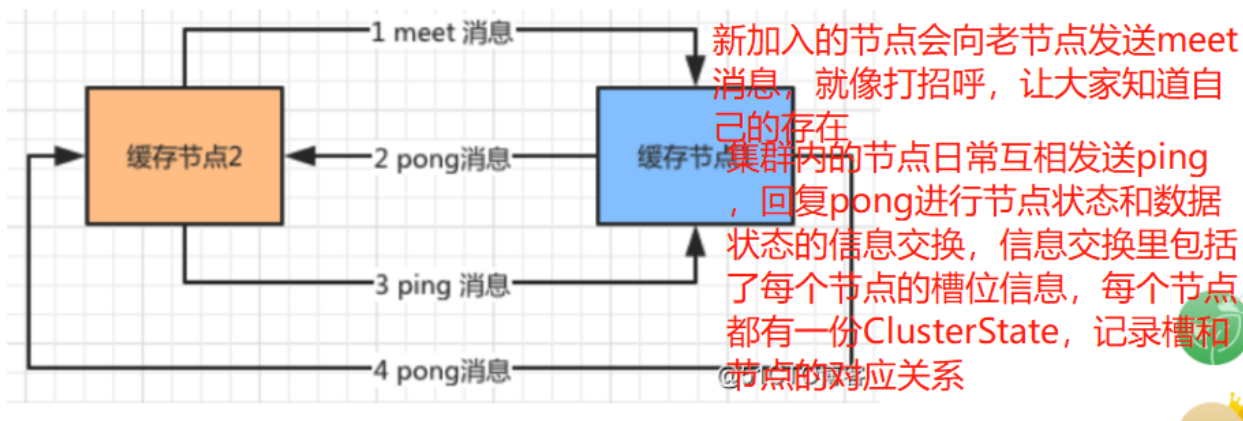
故障发现和恢复

## redis cluster 实现数据分区

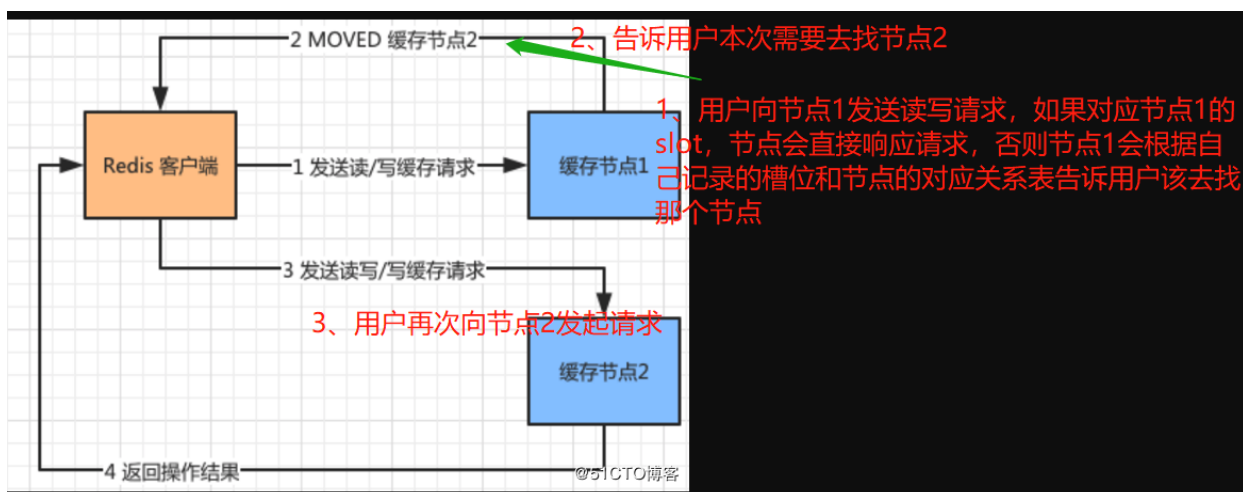
在redis-cluster中，将存储空间分成了16384个槽，槽是用来存放缓存信息的单位，redis-cluster的槽的范围是 0-16383。缓存信息通常是key-value的形式存放的，存储信息时，redis-cluster会根据算法 得到存储信息的存放槽位，而槽又被分散在不同的节点上，所以实现了数据被存储在不同的节点上。



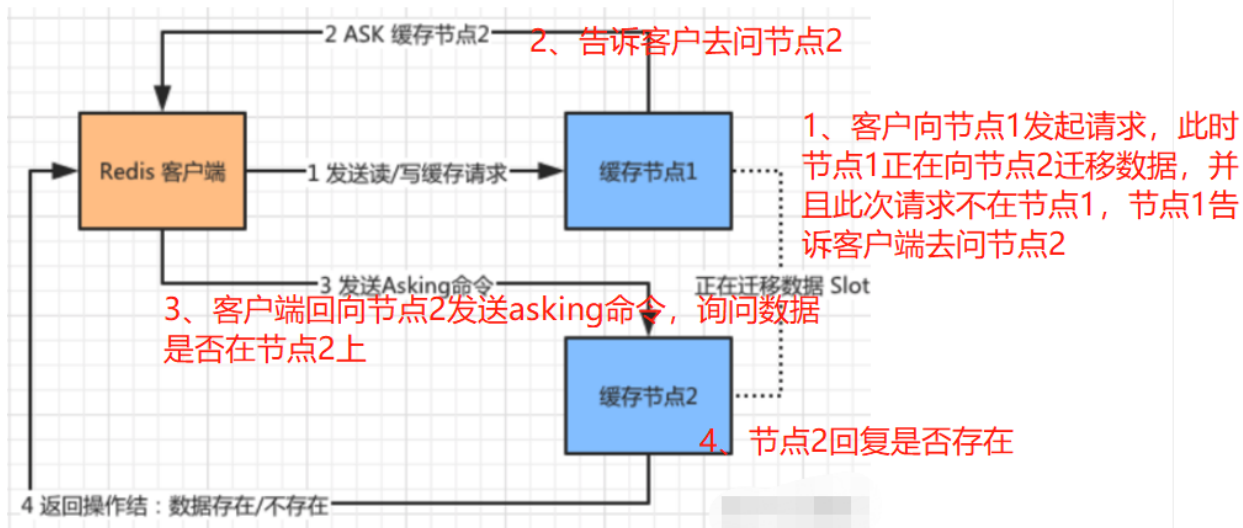
## 分布式缓存节点之间的通信



## 请求分布式缓存的路由MOVED 重定向请求



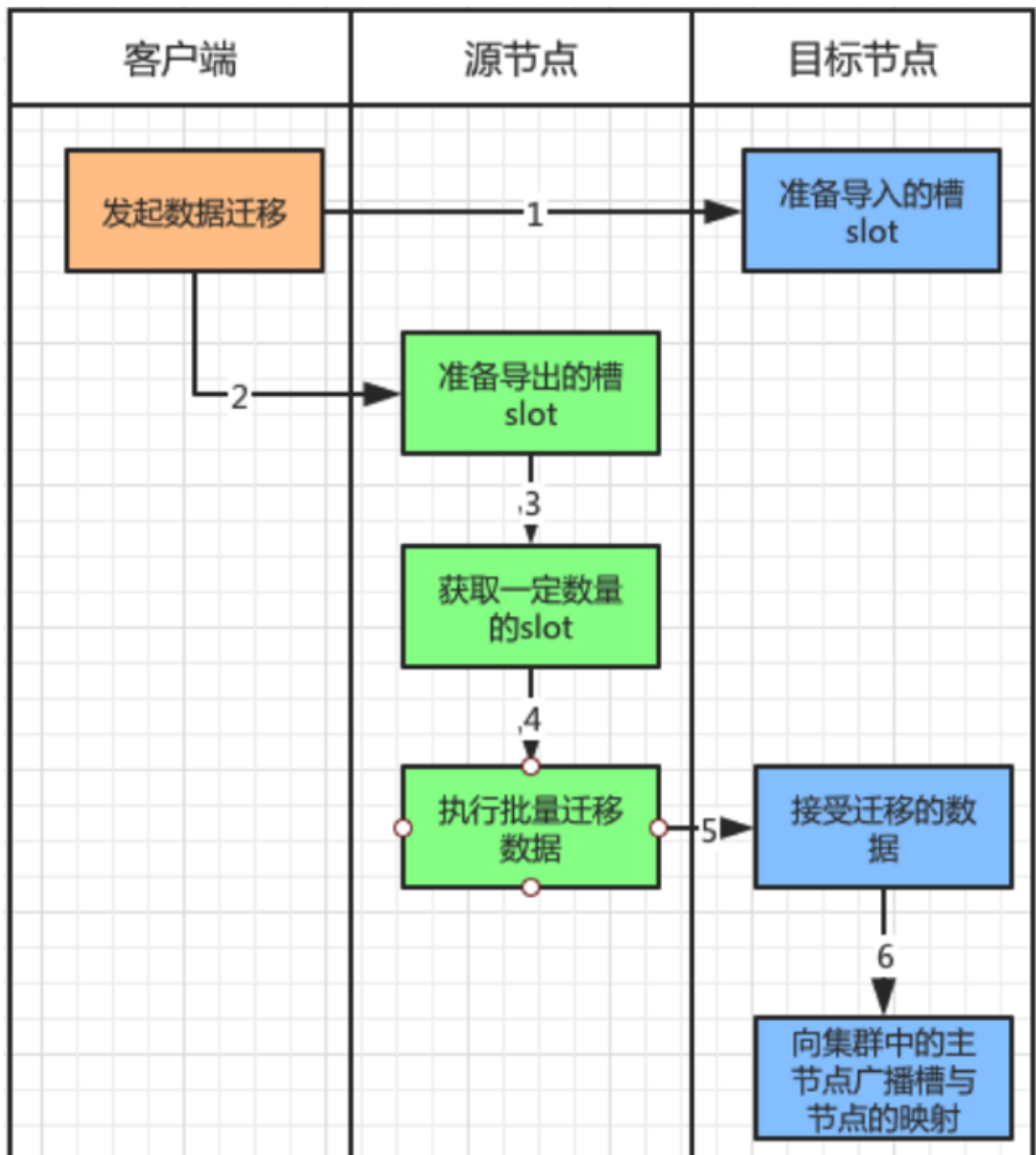
那么如果两个缓存节点正在做节点的数据迁移，此时客户端请求会如何处理呢？



## 缓存节点的扩展收缩

根据下图的步骤回想扩容主节点的步骤

由客户端发起节点之间的槽数据迁移，数据从源节点往目标节点迁移



## 故障发现和恢复

机器内半数以上的节点联系不上故障节点，故障节点就会被客观下线，如果故障节点是从节点，不会触发选举，如果故障节点是主节点就会触发选举，选举出新的节点替代故障节点。

参考资料：<https://blog.51cto.com/14279308/2484807#h3>