节点通过cluster meet ip port向指定的节点握手，握手成功之后就可以加入指定节点所在的集群。每个节点都会保存集群的状态比如在线还是下线，集群的配置纪元，以及集群内其他节点的信息。其他节点的信息是通过字典来保存的。

cluster meet命令的实现，节点A向指定ip port的其他节点B发送MEET消息，B给A回复一个PONG消息，表示已经成功收到MEET消息。

A再向B发送一条PING消息，B收到PING消息之后就确认A已经收到了PONG消息，握手完成。

redis有16384个槽位，当每个槽位都有节点在处理的时候，整个集群就处于上线的状态。有任何一个槽位没有处理，集群就出于下线的状态。

通过向节点发送CLUSTER ADDSLOTS命令，可以给节点指派一个或者多个槽位。完成上面的握手过程集群还没有上线，只有完成了槽位的指派，集群才能上线。

每个节点通过一个slots二进制位数组来记录槽位信息，数组长度为16384/8=2048个字节。数组的每个成员都有8个bit位，每个bit位就对应一个槽位，如果节点负责该槽位，则该bit位为1。

numslots则记录节点处理槽位的数量。

集群中的每个节点都会向其他节点通告自己的槽位信息，每个节点有closterNode \*slots[16384]的数组，用来保存每个槽位当前是由哪个节点负责的。集群内所有节点都会定时的发送ping + pong消息来互相通告信息。每个节点的槽位信息就是携带在ping消息内。

对于1000个以内的redis集群，16384个槽位已经够用了，而槽位数量过多，则为增加ping消息的大小。加上集群内节点是两两互相发消息的，这样就会成倍的增加集群中节点通信所消耗的带宽。

当客户端向节点发送命令的时候，接收命令的节点会计算当前操作的key属于哪个槽位，并检查这个槽位是否由自己负责。如果不属于自己负责，则通过MOVED消息告诉客户端槽位当前是由哪个节点负责。

槽迁移

redis-trib向源节点发送一个MIGRATE的命令，先将对应槽位中保存的所有键值对迁移到目的节点。迁移完成之后再通过cluster setslot命令将对应槽位指派给目的节点。这个命令会发送给集群内的所有节点。

在槽迁移过程中，如果客户端操作槽中的键值对，节点会先判断该键值对是否还在本地，如果在的话就执行操作，如果不在的话说明已经迁移，则向客户端返回ASK消息告诉客户端应该向哪个redis发送操作命令。

ASK和MOVED的区别

MOVED是告诉客户端槽已经迁移到新的节点，后续客户端要向新的节点发送命令。而ASK只是告诉客户端本次请求要向新节点发，但是后续的请求还是会发给当前的节点。相当于永久重定向和临时重定向。