

6.3.3 缓存节点之间的通信

如果说 Redis 集群的虚拟槽算法解决的是数据拆分和存放问题,那么各缓存节点之间是如何通信的呢?接下来我们讨论这个问题。在 Redis 集群中,缓存节点会被分配到一个或者多个服务器上,还可能根据缓存的数据量和支持的并发量扩展缓存节点的数目。如图 6-29 所示,假设 Redis 集群中本来只有缓存节点 1,此时由于业务扩展,新增了缓存节点 2。缓存节点 2 会通过 Gossip 协议向老节点,也就是缓存节点 1 发送一个 meet 消息。收到消息以后,缓存节点 1 礼貌性地回复一个 Pong 消息。此后缓存节点 2 就会定期给缓存节点 1 发送 Ping 消息,同样地,缓存节点 1 每次都会回复 Pong 消息。

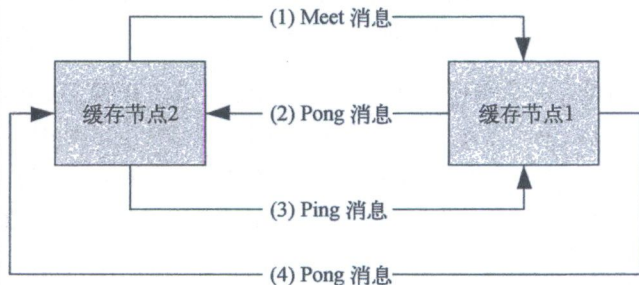


图 6-29 新上线的缓存节点 2 和缓存节点 1 通信

上面这个例子说明,Redis 集群中的缓存节点是通过 Gossip 协议相互通信的。节点之间通信的目的是维护彼此保存的元数据信息,元数据就是每个节点包含哪些数据、是否出现了故障。各节点通过 Gossip 协议不断地交互这些信息,就像一群人围坐在一起各种八卦,用不了多久每个节点就都会知道集群中所有节点的情况。整个传输过程大致分为以下几步。

- (1) Redis 集群中的每个缓存节点都会开通一个独立的 TCP 通道,用于和其他节点通信。
- (2) 存在一个节点定时任务,负责每隔一段时间从系统中选出发送节点。这个发送节点按照一定频率(例如 5 次/秒)随机向最久没有通信的节点发送 Ping 消息。
- (3) 接收到 Ping 消息的节点向发送节点回复 Pong 消息。
- (4) 不断重复上述步骤,让所有节点保持通信。

Gossip 协议中有如下 4 种消息。

- Meet 消息:用于通知旧节点加入了新节点。比如上面例子中提到的,缓存节点 2 上线后会给老节点 1 发送 Meet 消息,表示有新成员加入。
- Ping 消息:这个消息使用得最为频繁,其中封装了节点自身和其他节点的状态数据,被有规律地发给其他节点。