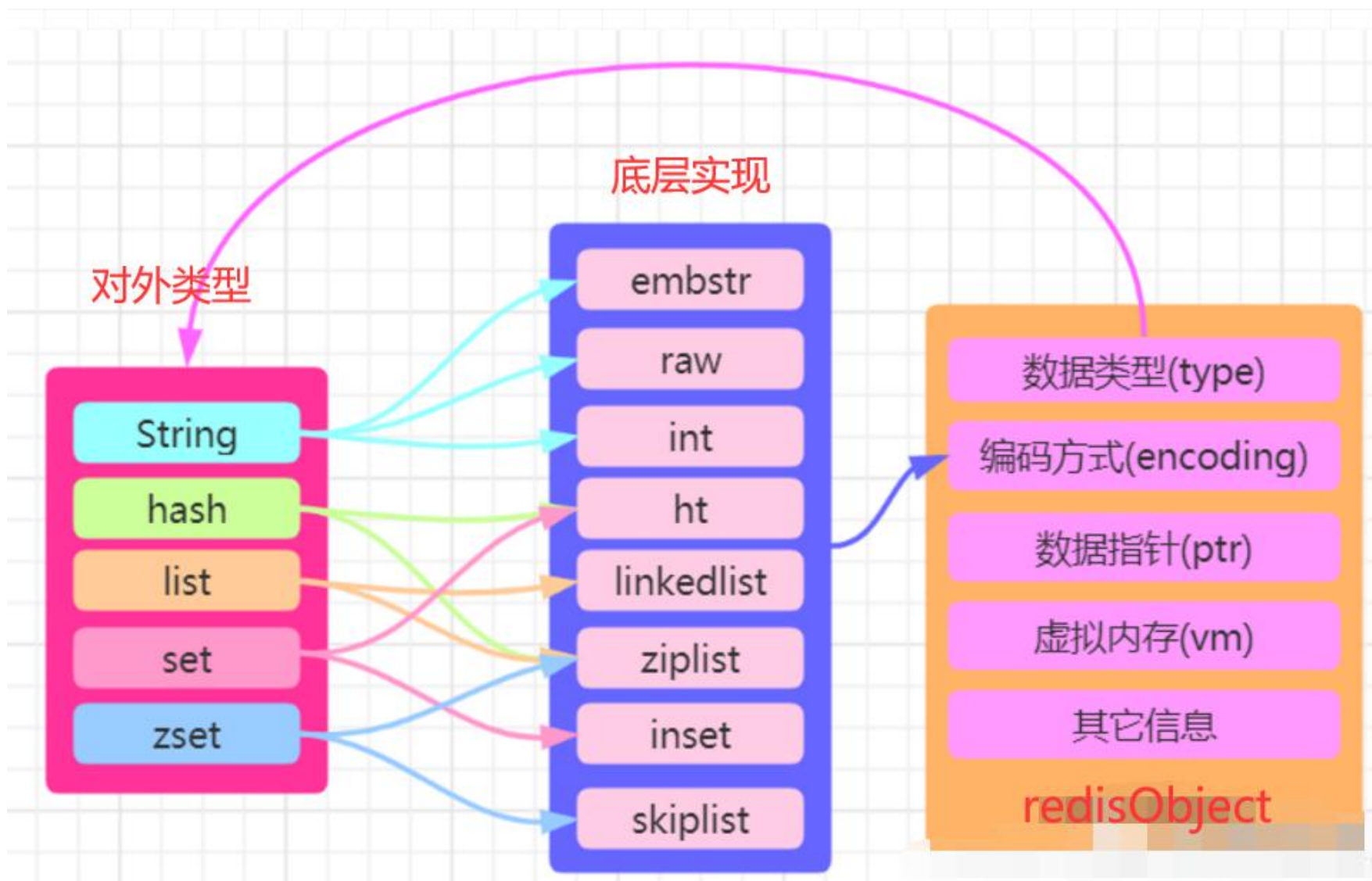


redis

Redis概述

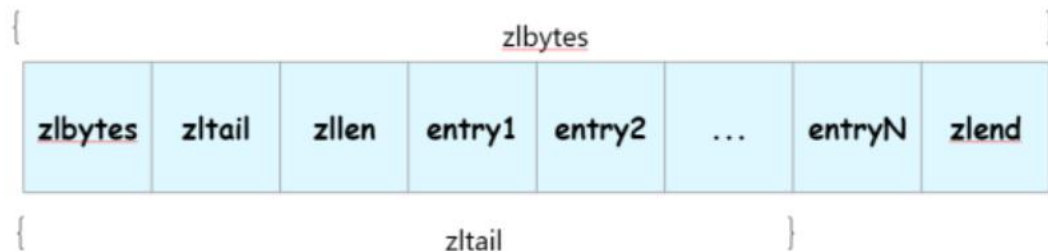
- 基于内存运行，性能高效
- 支持分布式，理论上可以无限扩展
- key-value存储系统
- 单进程单线程模型
- 丰富的数据类型
- 操作具有原子性
- 持久化
- 高并发读写
- 支持lua脚本

数据结构



DEBUG OBJECT key

ziplist



`zlbytes` 记录着整个压缩表的长度

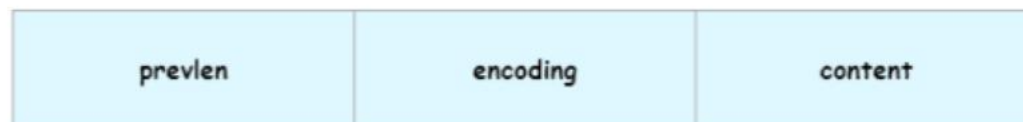
`zltail` 记录着最后一个元素的偏移量，这是为了倒序遍历整个zlist

`zllen` 用来记录压缩表中节点的数量

`entryX` 列表中的节点，节点用来存储具体的数据。

`zlend` 一个特殊值0xFF，用来标记压缩列表已经结束

ziplist node

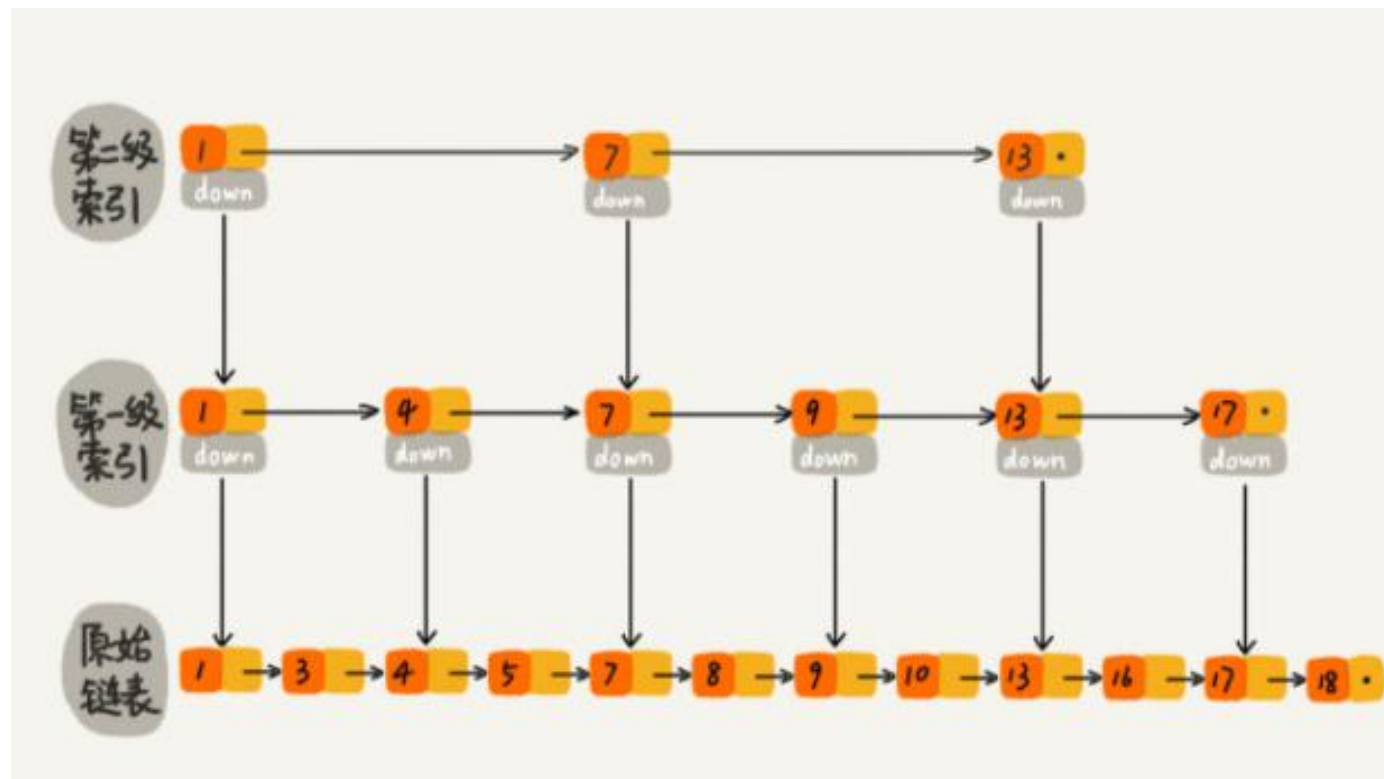


prevlen: 用来记录上一个节点的长度，因为压缩表可能需要倒序遍历，所以需要记录prevlen才能够定位出上一个entry的位置。

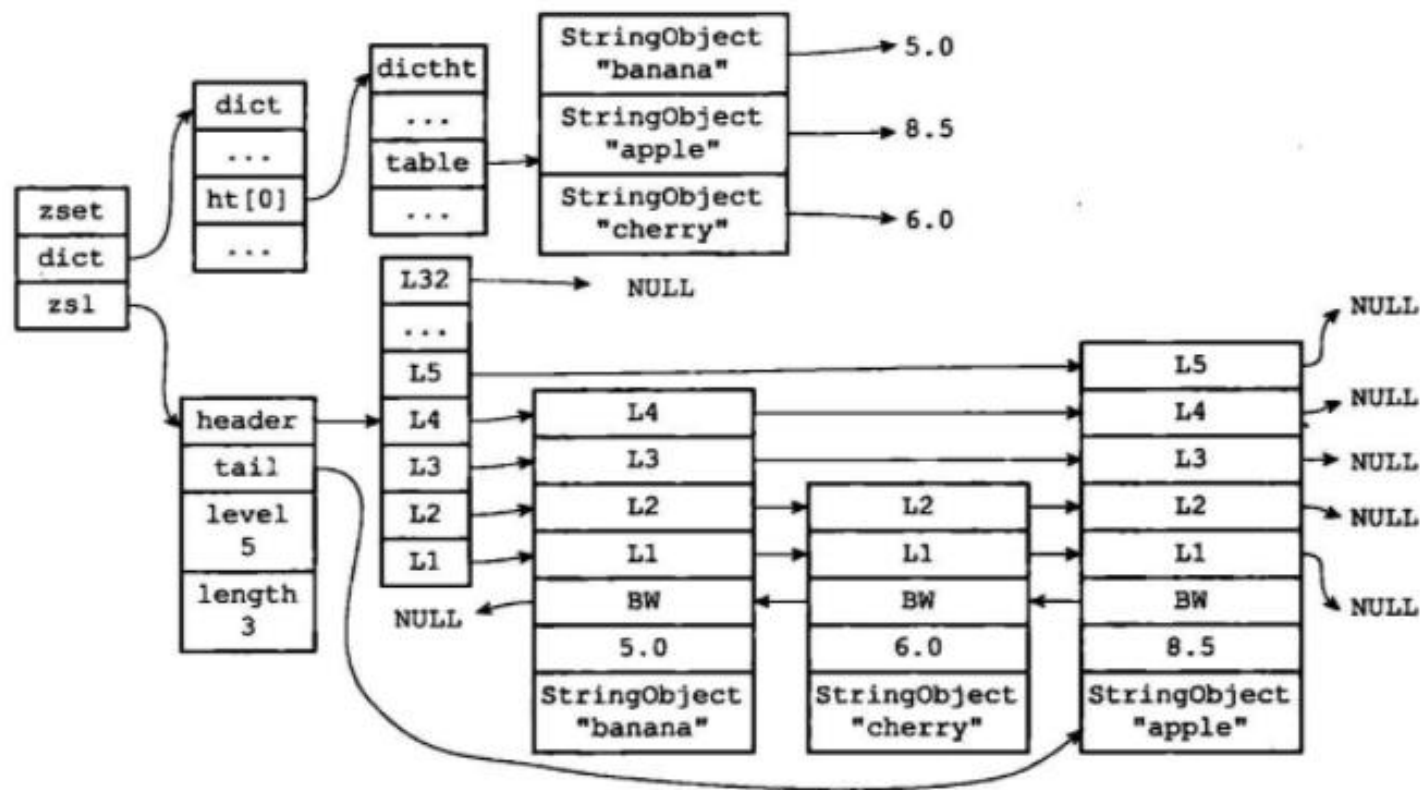
encoding : 为了节省压缩表占用的内存，Redis对Encoding进行了极致的设计。一般都是读取前8个字节，用来判断存储的数据是什么。举个简单的例子，如果前8个字节是00xxxxxx，这里的00开头，表示的是这是个非常短的字符串，后面的6个x表示字符串的长度， 2^6-1 等于63，所以，就表示，content是一个非常短的字符串，长度最多为63位。

content 用来存放具体的数据

skiplist



zset

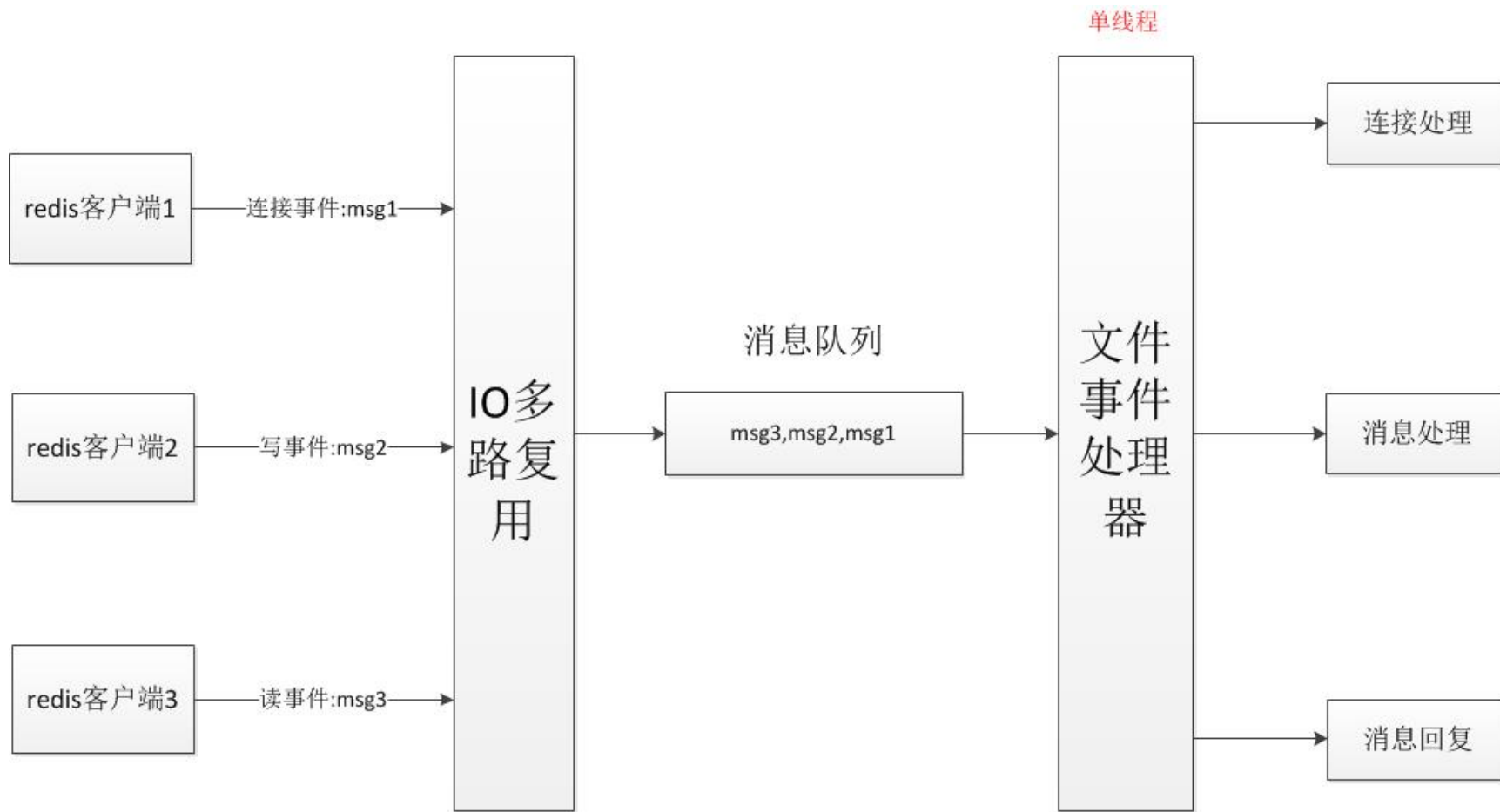


zset由一个dict和skiplist组成，数据结构如上图

字典的作用：可以在 $O(1)$ 时间复杂度内通过对象查找到分值，如zscore命令

跳表的作用：避免每次按分值区间查找（zrange）或者（zrank）的时候都要进行排序

Redis线程模型



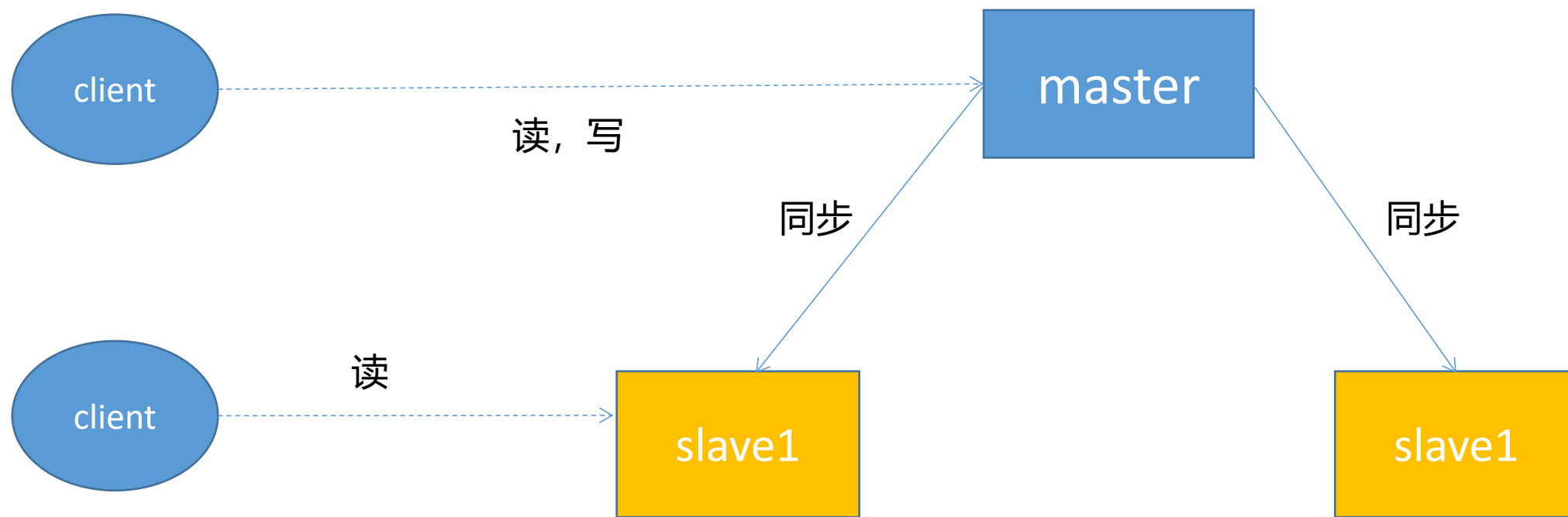
Redis是单线程的吗？

- 1、命令执行是单线程
- 2、网络连接是多线程 (redis6.0)
- 3、大key的删除是多线程(redis4.0)
- 4、未来会在更多的地方引入多线程

redis集群架构

- 1、主从模式
- 2、sentinel模式
- 3、cluster模式

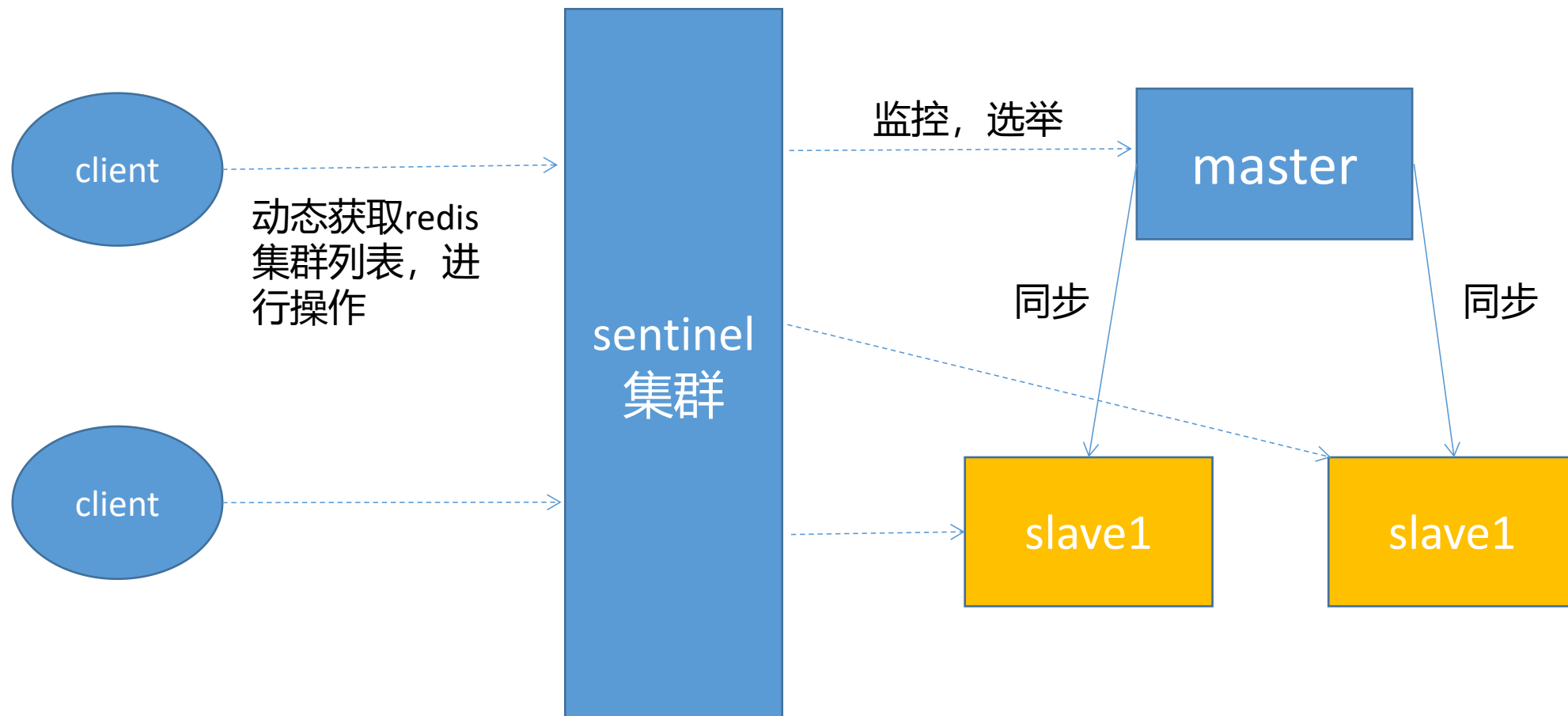
主从模式



主从模式弊端

- 1、不支持故障转移，master挂了后，集群不支持写入
- 2、集群中每个节点存的都是全量数据，存储能力无法横向扩展

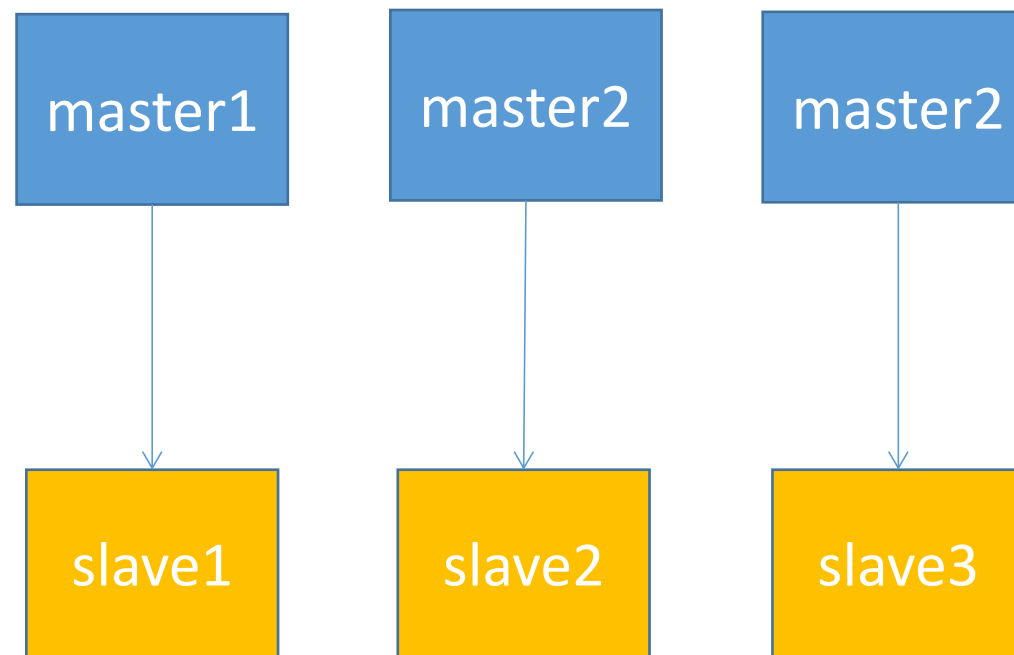
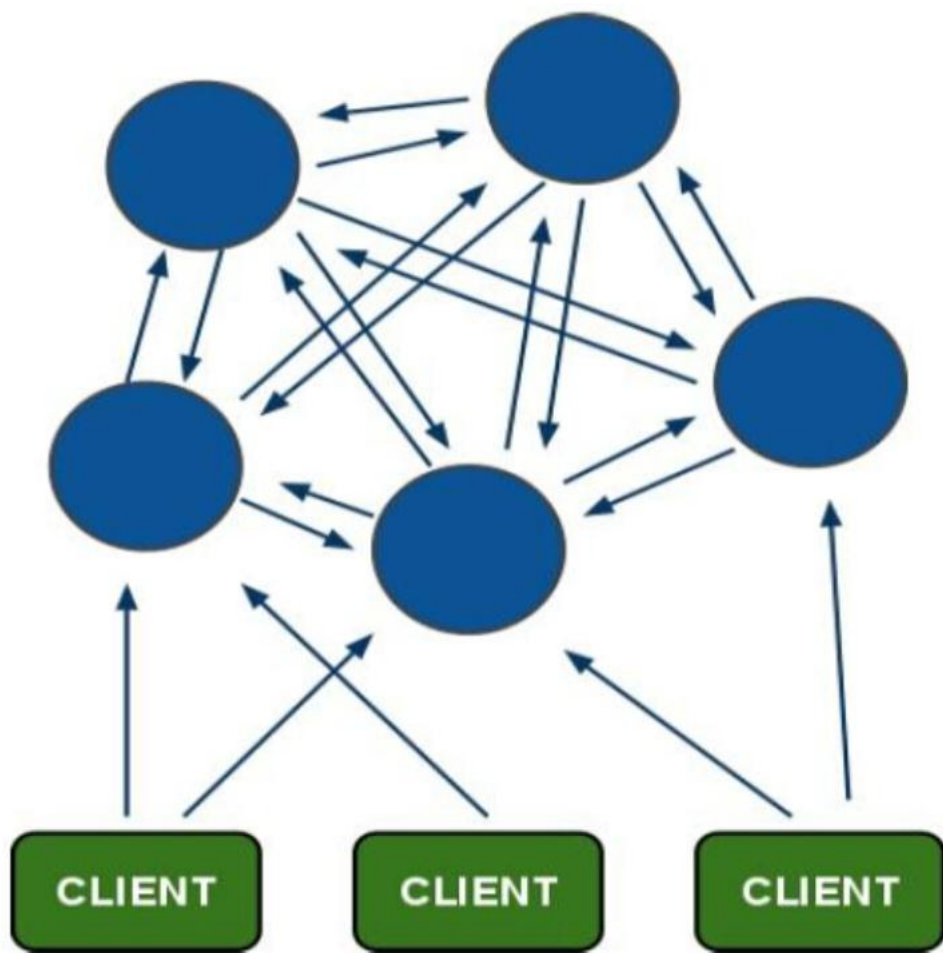
Sentinel模式



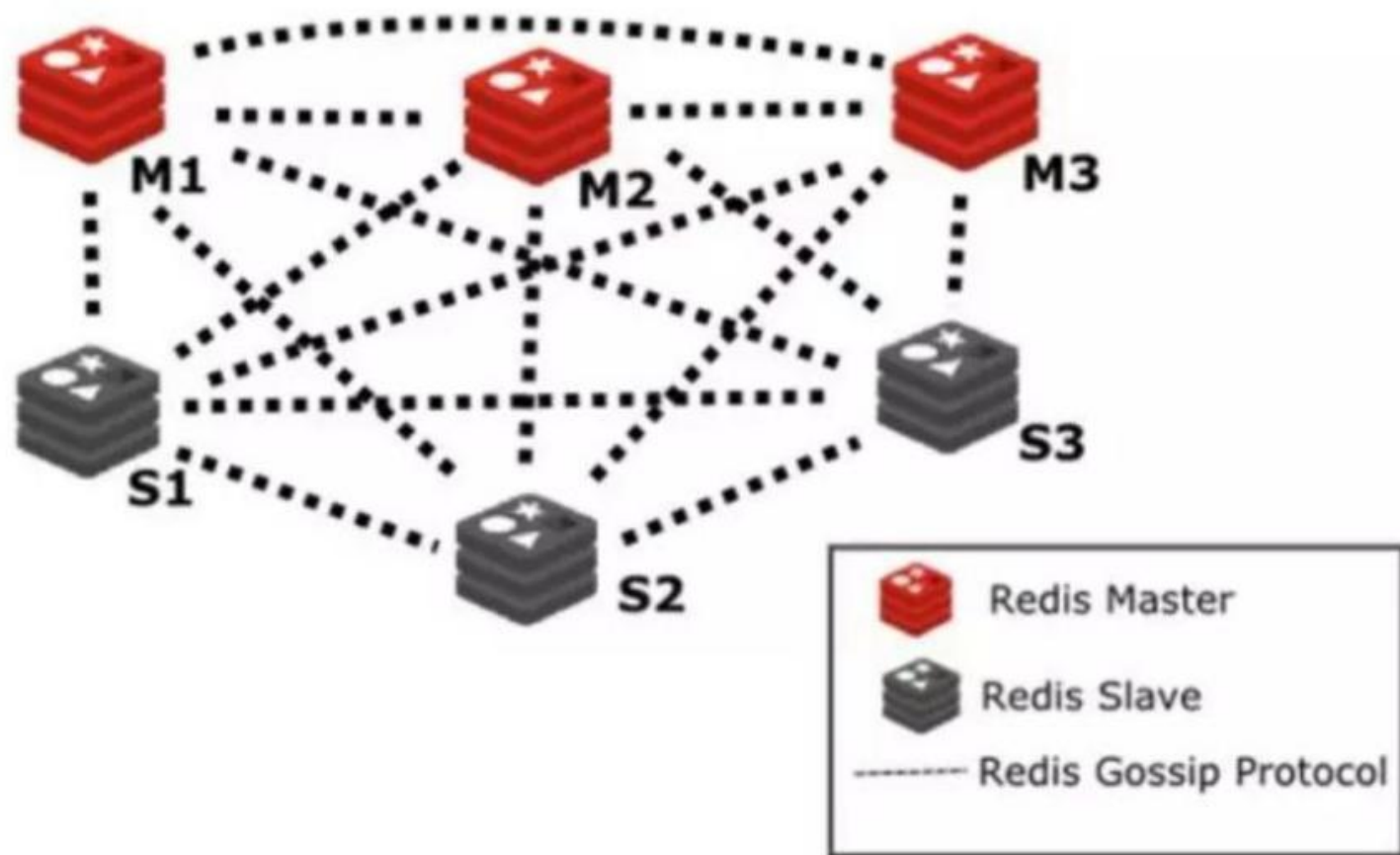
sentinel模式弊端

- 1、支持故障转移，master挂了后，重新选举；但需要引入sentinel集群，来管理真正的redis集群
- 2、集群中每个节点存的都是全量数据，存储能力无法横向扩展

cluster模式



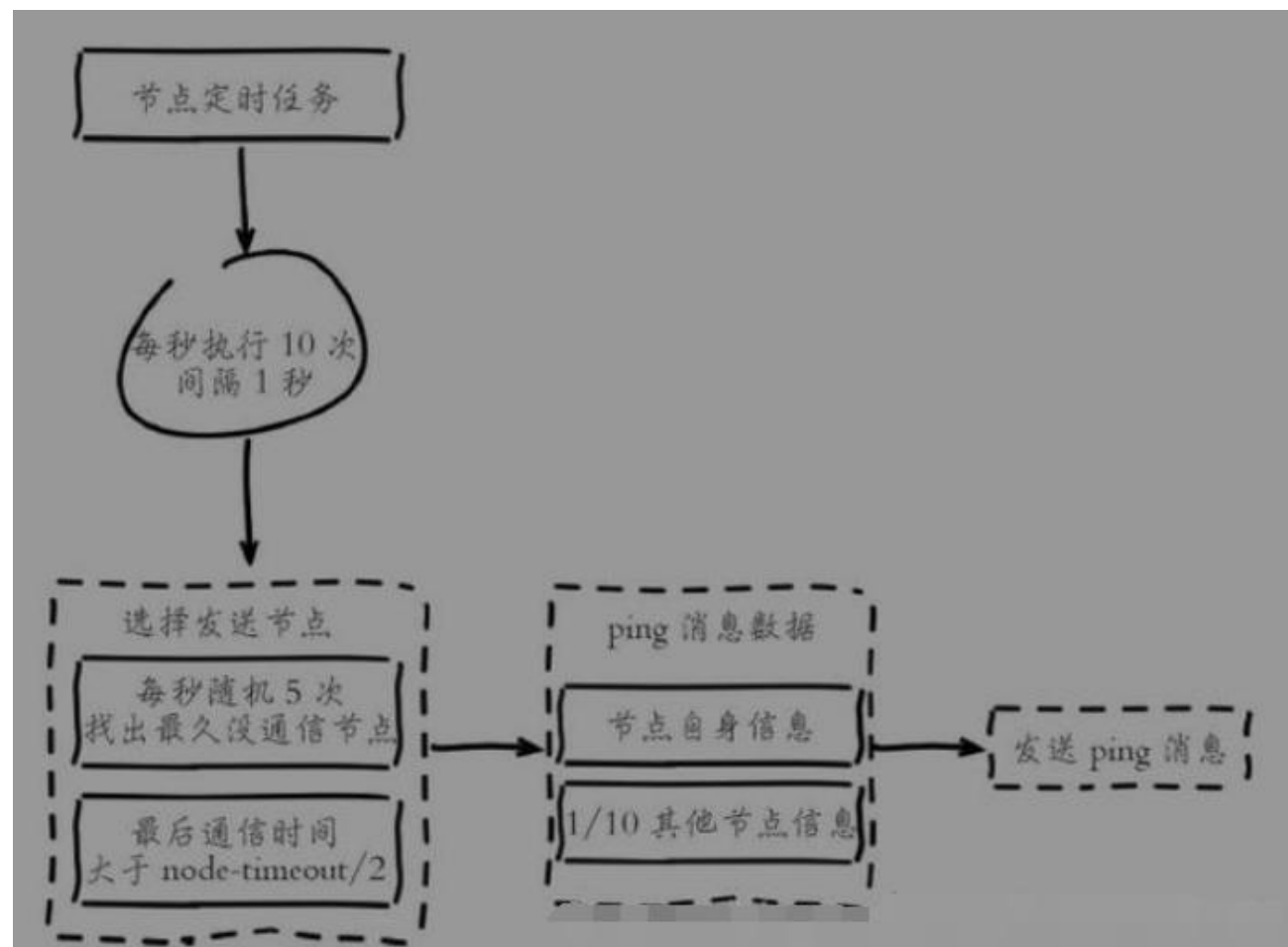
Redis Cluster 集群节点通信



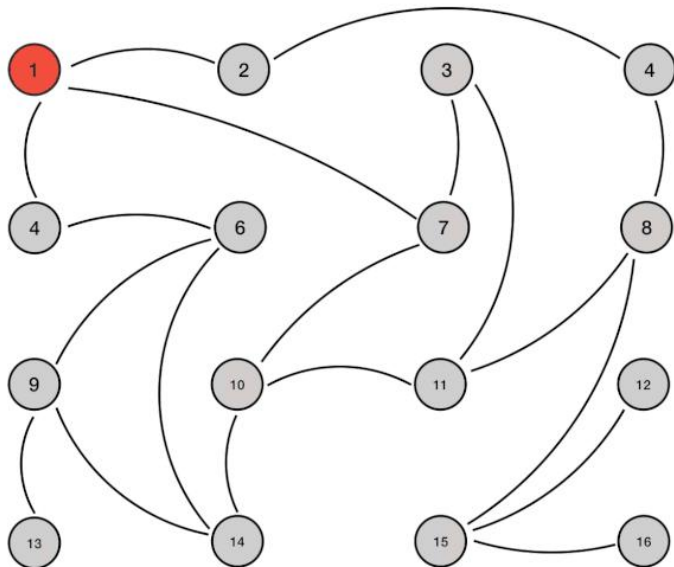
Gossip消息

- Gossip协议的主要职责就是信息交换。信息交换的载体就是节点彼此发送的Gossip消息，常用的Gossip
- 消息可分为：ping消息、pong消息、meet消息、fail消息
- meet消息：用于通知新节点加入。消息发送者通知接收者加入到当前集群，meet消息通信正常完成后，接收节点会加入到集群中并进行周期性的ping、pong消息交换
- ping消息：集群内交换最频繁的消息，集群内每个节点每秒向多个其他节点发送ping消息，用于检测
- 节点是否在线和交换彼此状态信息。ping消息发送封装了自身节点和部分其他节点的状态数据
- pong消息：当接收到ping、meet消息时，作为响应消息回复给发送方确认消息正常通信。pong消息内部封装了自身状态数据。节点也可以向集群内广播自身的pong消息来通知整个集群对自身状态进行更新
- fail消息：当节点判定集群内另一个节点下线时，会向集群内广播一个fail消息，其他节点接收到fail消息之后把对应节点更新为下线状态

ping

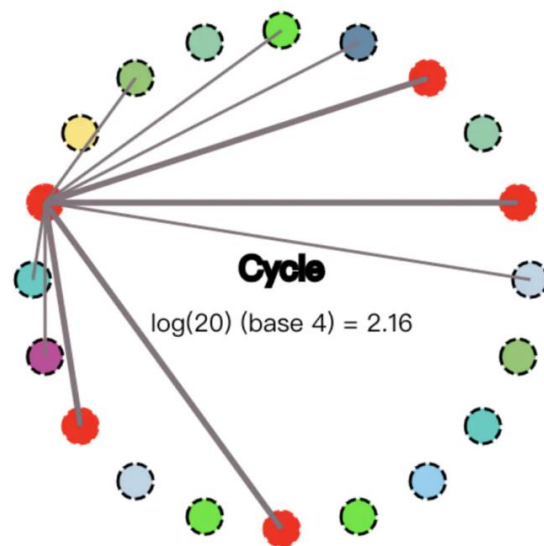
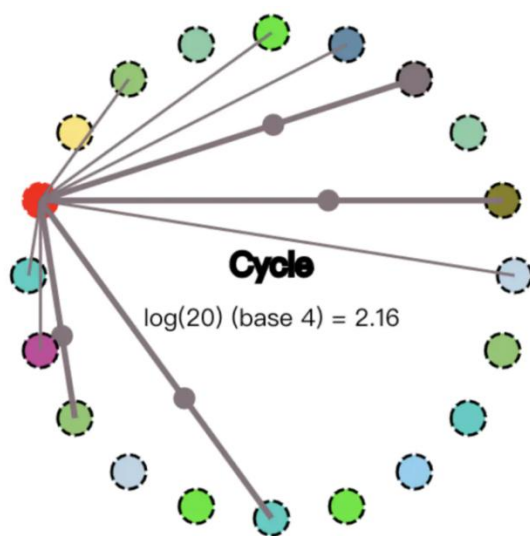


Gossip算法



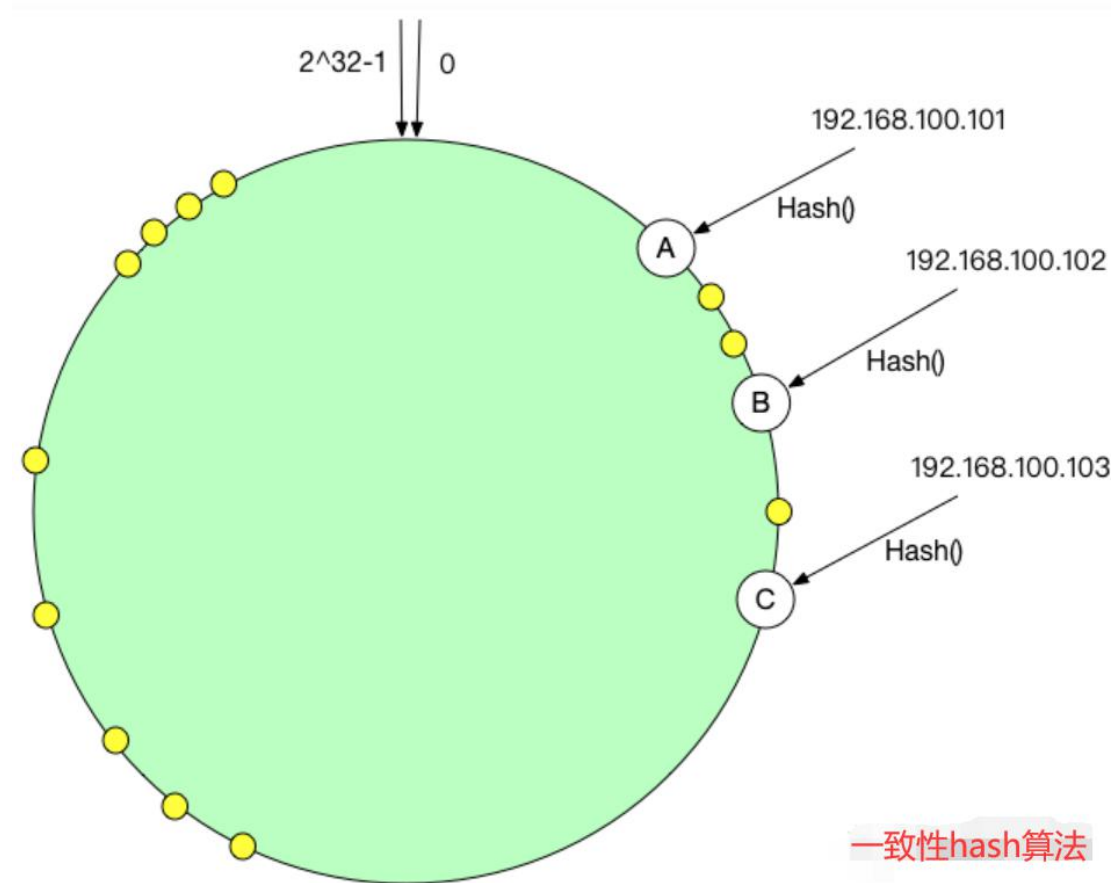
- 1、Gossip 是周期性的散播消息，把周期限定为 1 秒
- 2、被感染节点随机选择 k 个邻接节点 (fan-out) 散播消息，这里把 fan-out 设置为 3，每次最多往 3 个节点散播。
- 3、每次散播消息都选择**尚未发送过的节点**进行散播
- 4、收到消息的节点不再往发送节点散播，比如 $A \rightarrow B$ ，那么 B 进行散播的时候，不再发给 A 。

消息传播过程

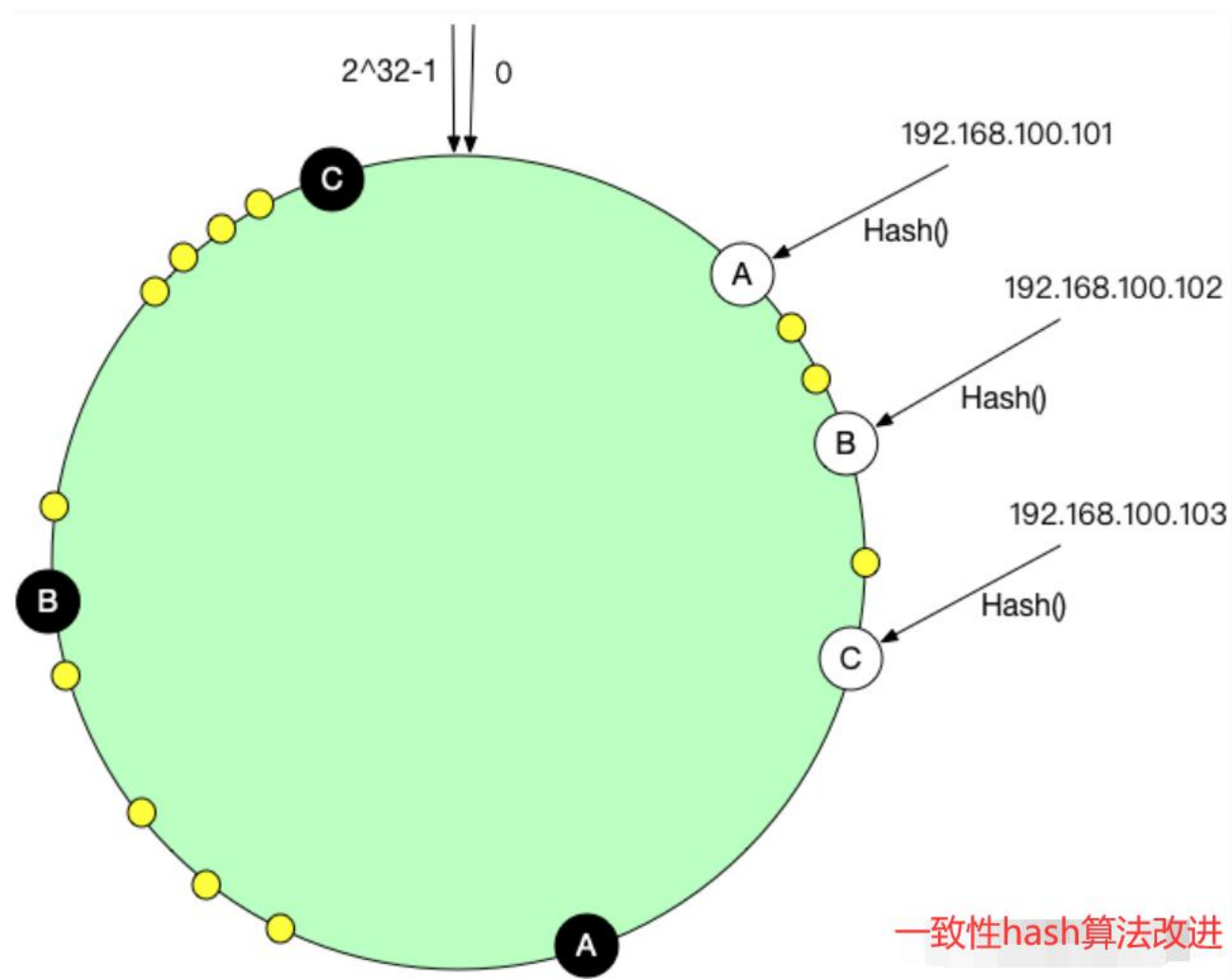


(20个节点，每个节点每次传播4个其他节点)

集群分片策略(一致性hash算法)

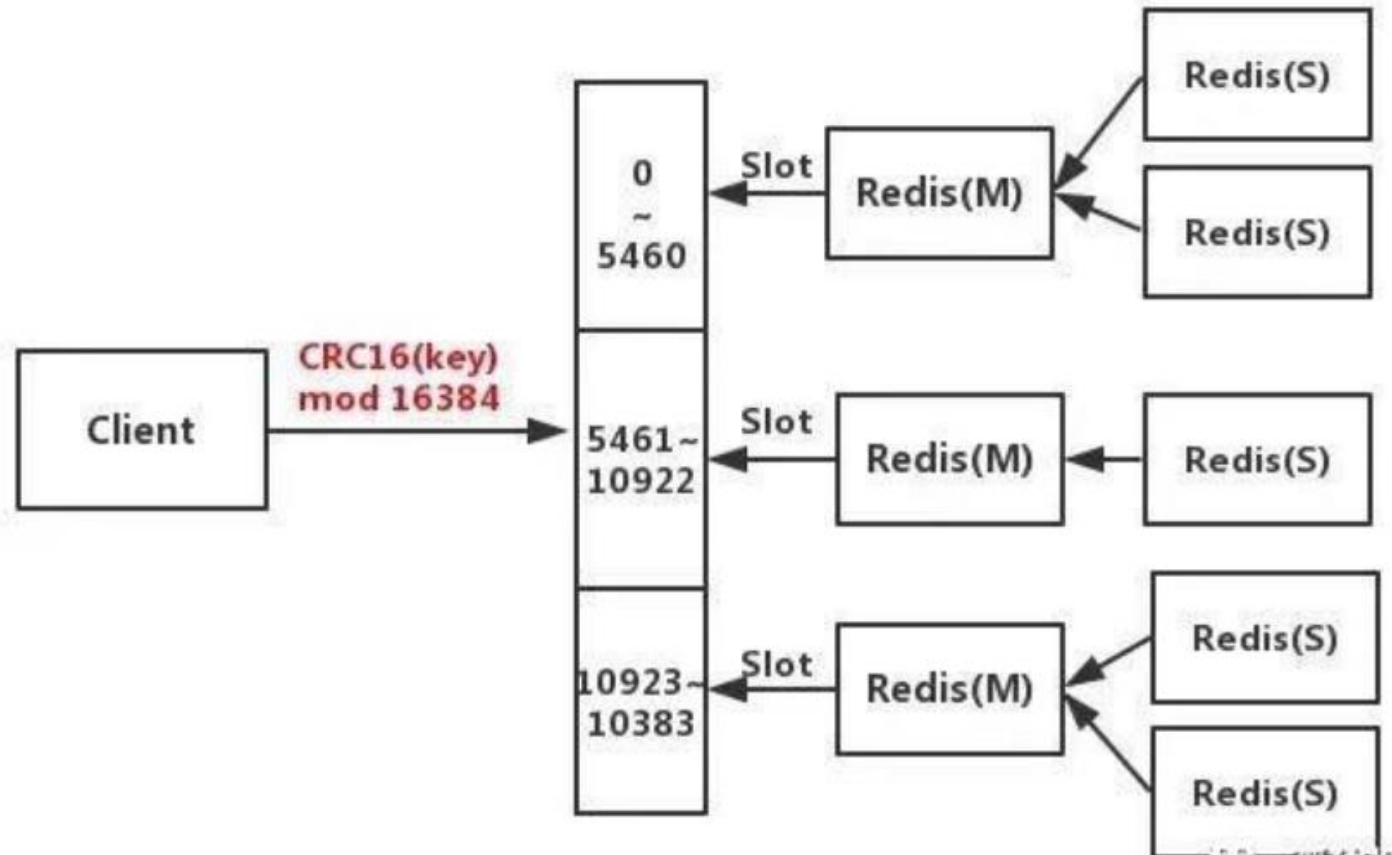


集群分片策略(一致性hash算法)



集群分片策略(hash slot)

$\text{slot} = \text{CRC16}(\text{key}) \% 16384$



why 16384?

- 由于使用CRC16算法，该算法可以产生 $2^{16}-1=65535$ 个值，可是为什么哈希槽的数量设置成了16384？
- **1.如果槽位为65536，发送心跳信息的信息头达8k，发送的心跳包过于庞大。**
- 如上所述，在消息头中，最占空间的是 `myslots[CLUSTER_SLOTS/8]`。当槽位为65536时，这块的大小是： $65536 \div 8 \div 1024 = 8\text{kb}$ 。因为每秒钟，redis节点需要发送一定数量的ping消息作为心跳包，如果槽位为65536，这个ping消息的消息头太大了，浪费带宽。
- **2.redis的集群主节点数量基本不可能超过1000个。**
- 如上所述，集群节点越多，心跳包的消息体内携带的数据越多。如果节点过1000个，也会导致网络拥堵。因此redis作者，不建议redis cluster节点数量超过1000个。那么，对于节点数在1000以内的redis cluster集群，16384个槽位够用了。没有必要拓展到65536个。
- **3.槽位越小，节点少的情况下，压缩率高**
- Redis主节点的配置信息中，它所负责的哈希槽是通过一张bitmap的形式来保存的，在传输过程中，会对bitmap进行压缩，但是如果bitmap的填充率 slots / N 很高的话(N 表示节点数)，bitmap的压缩率就很低。如果节点数很少，而哈希槽数量很多的话，bitmap的压缩率就很低。

redis cluster架构细节

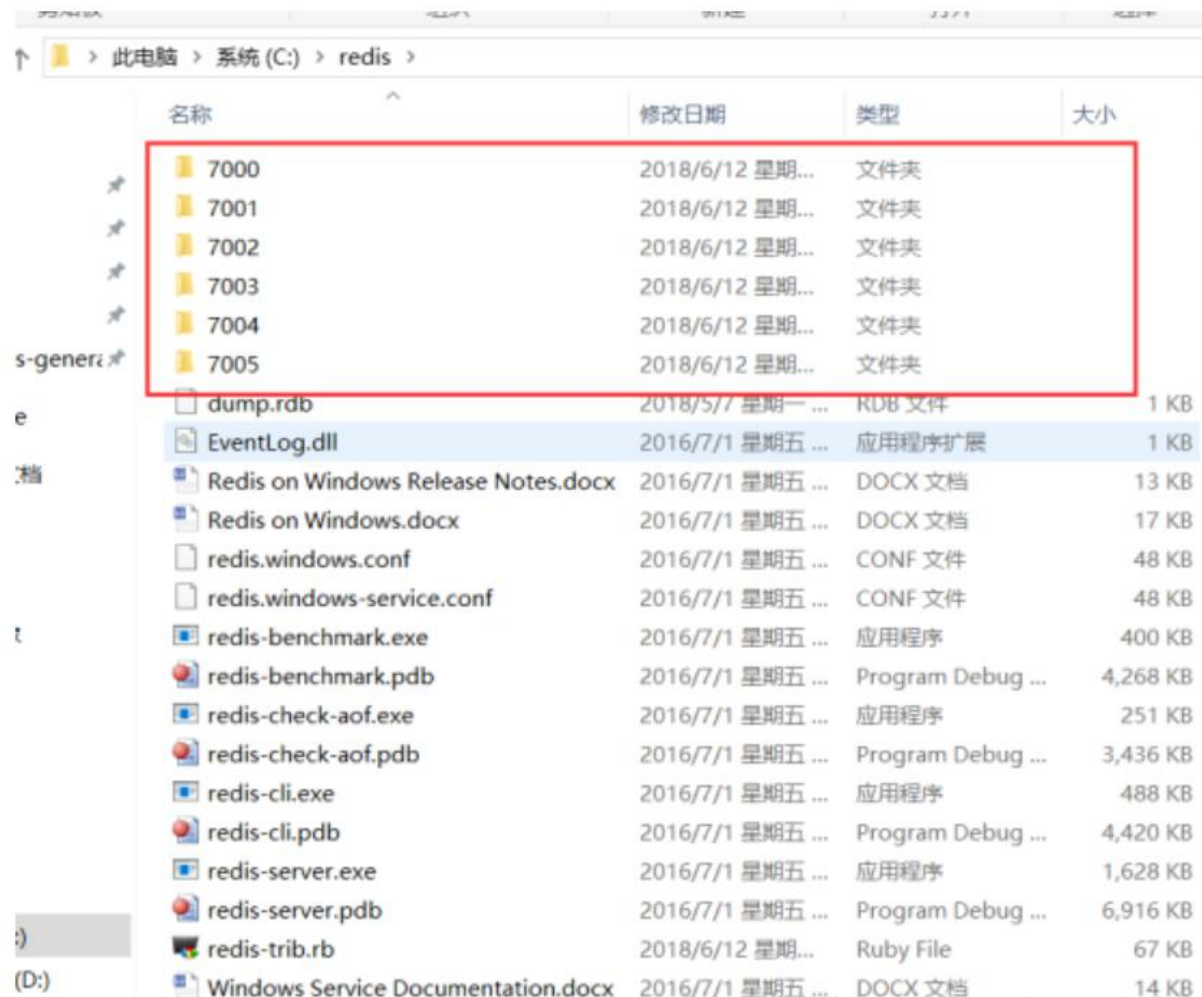
- 1、所有的 redis 节点彼此互联(PING-PONG 机制),内部使用二进制协议优化传输速度和带宽.
- 2、节点的 fail 是通过集群中超过半数的节点检测失效时才生效.
- 3、客户端与 redis 节点直连,不需要中间 proxy 层.客户端不需要连接集群所有节点,连接集群中任何一个可用节点即可
- 4、redis-cluster 把所有的物理节点映射到[0-16383]slot 上,cluster 负责维护
 - node<->slot<->value

windows版 redis集群搭建

- 1、安装单机版redis,安装包见附件:Redis-x64-3.0.504.msi
- 2、安装ruby运行环境, 见附件: rubyinstaller-2.2.4-x64.exe
- 3、使用ruby本地安装redis集群管理工具:
- `gem install --local D:/xxx(你的本地路径)/redis-3.2.2.gem`

redis集群配置

1、在redis安装目录下创建6个节点的文件夹：7000-7005，如右图



redis集群配置

1、将redis安装目录下面的文件复制到节点7000文件夹下面（其他的也是这样操作）

> 此电脑 > 系统 (C:) > redis > 7000				
名称	修改日期	类型	大小	
appendonly.aof	2018/6/12 星期...	AOF 文件	1 KB	
dump.rdb	2018/6/12 星期...	RDB 文件	1 KB	
EventLog.dll	2016/7/1 星期五 ...	应用程序扩展	1 KB	
nodes-7000.conf	2018/6/13 星期...	CONF 文件	1 KB	
Redis on Windows Release Notes.docx	2016/7/1 星期五 ...	DOCX 文档	13 KB	
Redis on Windows.docx	2016/7/1 星期五 ...	DOCX 文档	17 KB	
redis.windows.conf	2018/6/12 星期...	CONF 文件	48 KB	
redis.windows-service.conf	2016/7/1 星期五 ...	CONF 文件	48 KB	
redis-benchmark.exe	2016/7/1 星期五 ...	应用程序	400 KB	
redis-benchmark.pdb	2016/7/1 星期五 ...	Program Debug ...	4,268 KB	
redis-check-aof.exe	2016/7/1 星期五 ...	应用程序	251 KB	
redis-check-aof.pdb	2016/7/1 星期五 ...	Program Debug ...	3,436 KB	
redis-cli.exe	2016/7/1 星期五 ...	应用程序	488 KB	
redis-cli.pdb	2016/7/1 星期五 ...	Program Debug ...	4,420 KB	
redis-server.exe	2016/7/1 星期五 ...	应用程序	1,628 KB	
redis-server.pdb	2016/7/1 星期五 ...	Program Debug ...	6,916 KB	
start.bat	2018/6/12 星期...	Windows 批处理...	1 KB	
Windows Service Documentation.docx	2016/7/1 星期五 ...	DOCX 文档	14 KB	

redis集群配置

1、修改7000文件夹下redis.windows.conf 文件配置，其他节点类似

```
redis.windows.conf  renting.sql  redis.windows-service.conf  footer.jsp  login.jsp
56 bind 127.0.0.1
57
58 # Protected mode is a layer of security protection, in order to avoid that
59 # Redis instances left open on the internet are accessed and exploited.
60 #
61 # When protected mode is on and if:
62 #
63 # 1) The server is not binding explicitly to a set of addresses using the
64 #    "bind" directive.
65 # 2) No password is configured.
66 #
67 # The server only accepts connections from clients connecting from the
68 # IPv4 and IPv6 loopback addresses 127.0.0.1 and ::1, and from Unix domain
69 # sockets.
70 #
71 # By default protected mode is enabled. You should disable it only if
72 # you are sure you want clients from other hosts to connect to Redis
73 # even if no authentication is configured, nor a specific set of interfaces
74 # are explicitly listed using the "bind" directive.
75 protected-mode yes
76
77 # Accept connections on the specified port, default is 6379 (IANA #815344).
78 # If port 0 is specified Redis will not listen on a TCP socket.
79 port 7000
80
81 # TCP listen() backlog.
82 #
83 # In high requests-per-second environments you need an high backlog in order
84 # to avoid slow clients connections issues. Note that the Linux kernel
85 # will silently truncate it to the value of /proc/sys/net/core/somaxconn so
86 # make sure to raise both the value of somaxconn and tcp_max_syn_backlog
87 # in order to get the desired effect.
88 tcp-backlog 511
```

其他的也是，端口需要对应节点，7000,7001，
记得删除最前面的 #，左边不要留空格

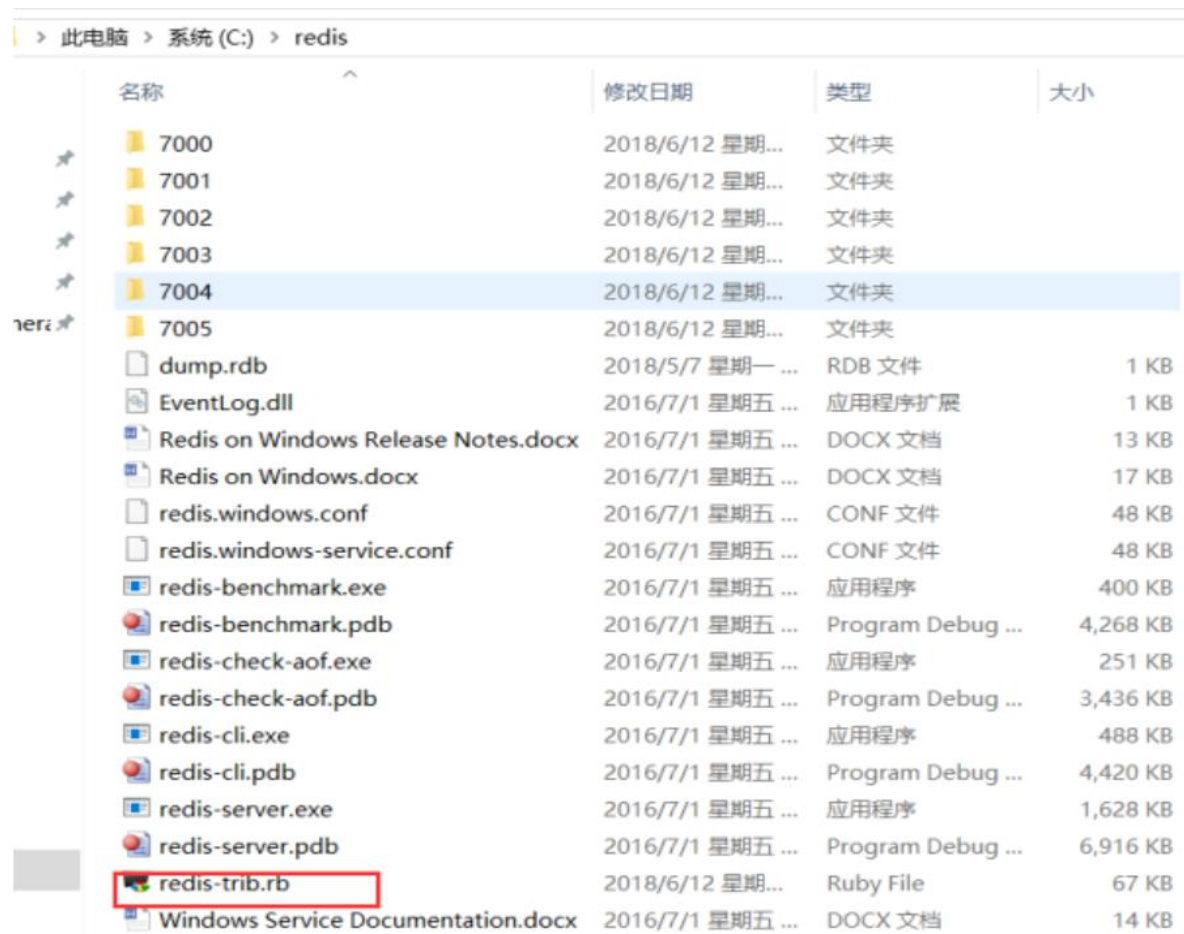
```
redis.windows.conf  renting.sql  redis.windows-service.conf  footer.jsp  login.jsp
693 lua-time-limit 5000
694
695 ##### REDIS CLUSTER #####
696 #
697 # +-----+
698 # WARNING EXPERIMENTAL: Redis Cluster is considered to be stable code, however
699 # in order to mark it as "mature" we need to wait for a non trivial percentage
700 # of users to deploy it in production.
701 # +-----+
702 #
703 # Normal Redis instances can't be part of a Redis Cluster; only nodes that are
704 # started as cluster nodes can. In order to start a Redis instance as a
705 # cluster node enable the cluster support uncommenting the following:
706 #
707 cluster-enabled yes
708
709 # Every cluster node has a cluster configuration file. This file is not
710 # intended to be edited by hand. It is created and updated by Redis nodes.
711 # Every Redis Cluster node requires a different cluster configuration file.
712 # Make sure that instances running in the same system do not have
713 # overlapping cluster configuration file names.
714 #
715 cluster-config-file nodes-7000.conf
716
717 # Cluster node timeout is the amount of milliseconds a node must be unreachable
718 # for it to be considered in failure state.
719 # Most other internal time limits are multiple of the node timeout.
720 #
721 cluster-node-timeout 15000
722
723 # A slave of a failing master will avoid to start a failover if its data
724 # looks too old.
725 #
726 # There is no simple way for a slave to actually have a exact measure of
727 # its "data age", so the following two checks are performed:
728 #
729 # 1) If there are multiple slaves able to failover, they exchange messages
730 # in order to try to give an advantage to the slave with the best
731 # replication offset (more data from the master processed)
732 # Slaves will try to get their rank by offset, and apply to the start
```

编写单个节点的启动脚本

- 编写一个bat来启动redis，在每个节点目录下创建start.bat,内容如下（端口记得对应）
- title redis-7000
- redis-server.exe redis.windows.conf

安装集群配置脚本

- 将附件中的集群配置脚本 redis-trib.rb，拷贝到redis 安装目录下



File Explorer window showing the contents of the 'redis' directory. The file 'redis-trib.rb' is highlighted with a red rectangle.

名称	修改日期	类型	大小
7000	2018/6/12 星期...	文件夹	
7001	2018/6/12 星期...	文件夹	
7002	2018/6/12 星期...	文件夹	
7003	2018/6/12 星期...	文件夹	
7004	2018/6/12 星期...	文件夹	
7005	2018/6/12 星期...	文件夹	
dump.rdb	2018/5/7 星期一 ...	RDB 文件	1 KB
EventLog.dll	2016/7/1 星期五 ...	应用程序扩展	1 KB
Redis on Windows Release Notes.docx	2016/7/1 星期五 ...	DOCX 文档	13 KB
Redis on Windows.docx	2016/7/1 星期五 ...	DOCX 文档	17 KB
redis.windows.conf	2016/7/1 星期五 ...	CONF 文件	48 KB
redis.windows-service.conf	2016/7/1 星期五 ...	CONF 文件	48 KB
redis-benchmark.exe	2016/7/1 星期五 ...	应用程序	400 KB
redis-benchmark.pdb	2016/7/1 星期五 ...	Program Debug ...	4,268 KB
redis-check-aof.exe	2016/7/1 星期五 ...	应用程序	251 KB
redis-check-aof.pdb	2016/7/1 星期五 ...	Program Debug ...	3,436 KB
redis-cli.exe	2016/7/1 星期五 ...	应用程序	488 KB
redis-cli.pdb	2016/7/1 星期五 ...	Program Debug ...	4,420 KB
redis-server.exe	2016/7/1 星期五 ...	应用程序	1,628 KB
redis-server.pdb	2016/7/1 星期五 ...	Program Debug ...	6,916 KB
redis-trib.rb	2018/6/12 星期...	Ruby File	67 KB
Windows Service Documentation.docx	2016/7/1 星期五 ...	DOCX 文档	14 KB

启动各个节点

- 到各个目录（7000--7005）下双击运行启动脚本，运行redis

名称	修改日期	类型	大小
redis on windows release notes.docx	2016/7/1 15:52	DOCX 文档	17 KB
Redis on Windows.docx	2016/7/1 15:52	DOCX 文档	17 KB
redis.windows.conf	2020/9/20 9:13	CONF 文件	43 KB
redis.windows-service.conf	2020/9/19 23:06	CONF 文件	43 KB
redis-benchmark.exe	2016/7/1 15:55	应用程序	397 KB
redis-benchmark.pdb	2016/7/1 15:55	VisualStudio.pdb.f9...	4,268 KB
redis-check-aof.exe	2016/7/1 15:55	应用程序	251 KB
redis-check-aof.pdb	2016/7/1 15:55	VisualStudio.pdb.f97c64a8	3,436 KB
redis-check-dump.exe	2016/7/1 15:55	应用程序	262 KB
redis-check-dump.pdb	2016/7/1 15:55	VisualStudio.pdb.f9...	3,404 KB
redis-cli.exe	2016/7/1 15:55	应用程序	471 KB
redis-cli.pdb	2016/7/1 15:55	VisualStudio.pdb.f9...	4,412 KB
redis-server.exe	2016/7/1 15:55	应用程序	1,517 KB
redis-server.pdb	2016/7/1 15:55	VisualStudio.pdb.f9...	6,748 KB
start.bat	2020/9/20 21:38	Windows 批处理文件	1 KB
Windows Service Documentation.docx	2016/7/1 9:17	DOCX 文档	14 KB

```
C:\WINDOWS\system32\cmd.exe - redis-server.exe redis.windows.conf
Microsoft Windows [版本 10.0.16299.431]
(c) 2017 Microsoft Corporation. 保留所有权利。

C:\Users\Administrator>
C:\Users\Administrator>cd C:/redis
C:\redis>redis-server.exe redis.windows.conf

Redis 3.2.100 (00000000/0) 64 bit

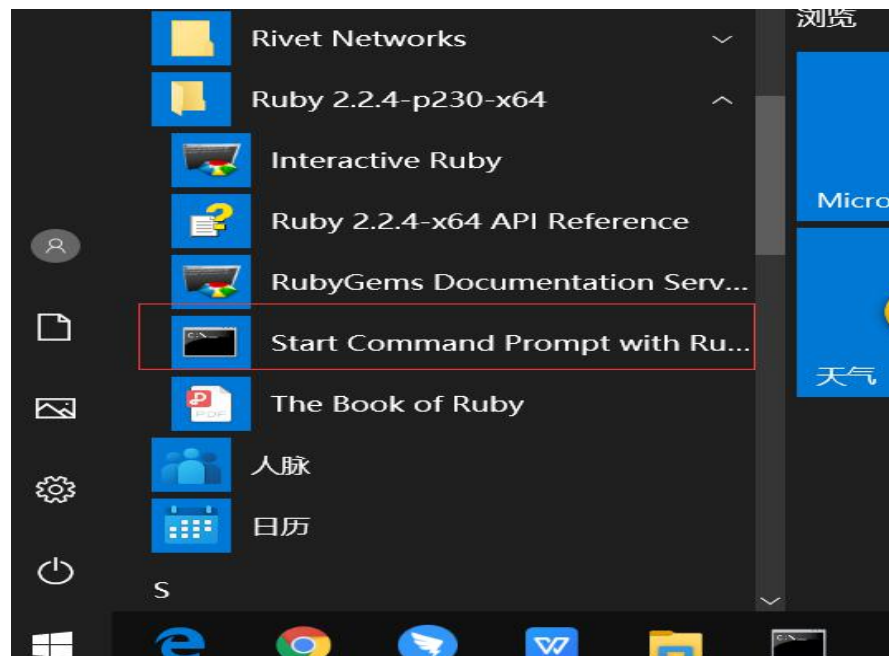
Running in standalone mode
Port: 6379
PID: 8664

http://redis.io

[8664] 13 Jun 11:27:51.136 # Server started. Redis version 3.2.100
[8664] 13 Jun 11:27:51.139 * DB loaded from disk: 0.002 seconds
[8664] 13 Jun 11:27:51.140 * The server is now ready to accept connections on
```


构建redis集群

- 1、启动ruby客户端:



- 2、运行如下命令构建redis集群:
- `ruby redis-trib.rb create --replicas 1 127.0.0.1:7000 127.0.0.1:7001 127.0.0.1:7002 127.0.0.1:7003 127.0.0.1:7004 127.0.0.1:7005`

集群信息如下

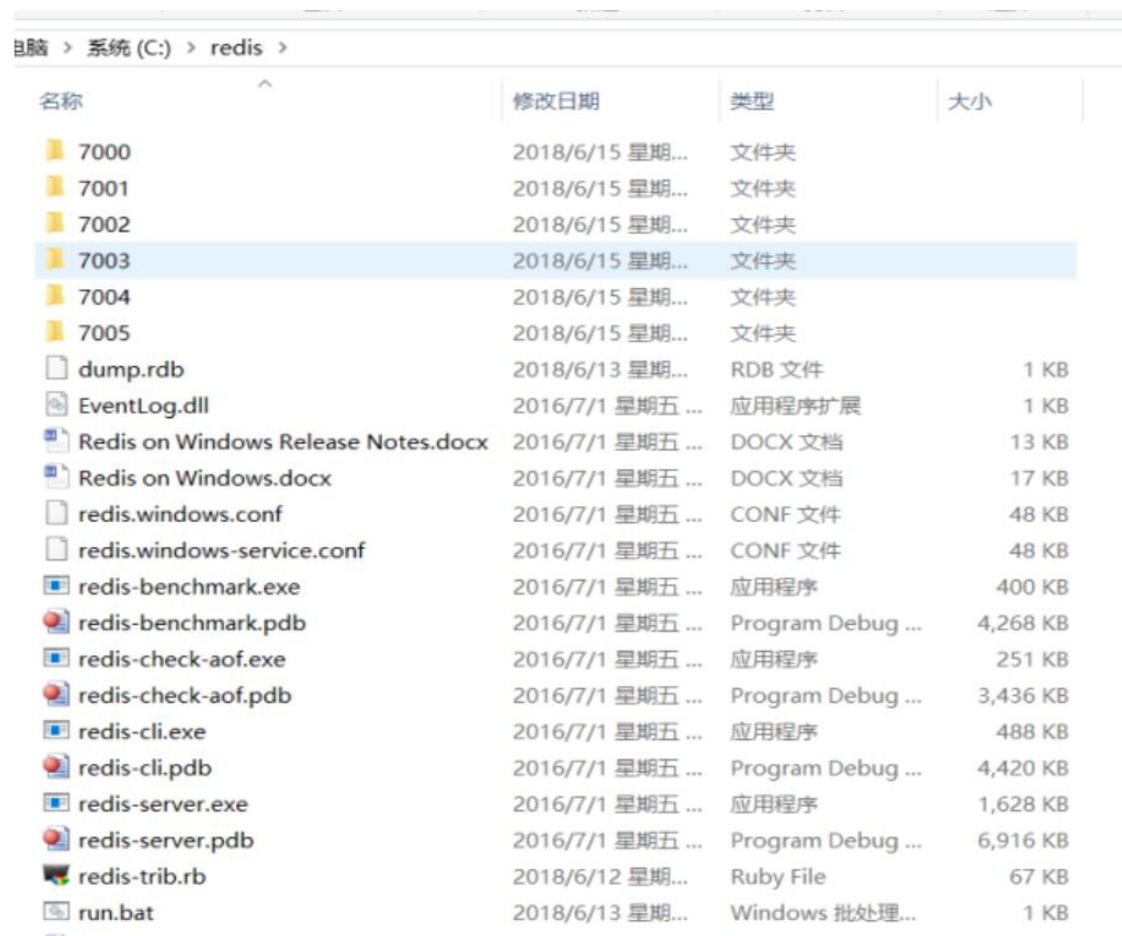
```
Windows PowerShell
PS D:\Program Files\Redis> ruby redis-trib.rb create --replicas 1 127.0.0.1:7000 127.0.0.1:7001 127.0.0.1:7002 127.0.0.1:7003 127.0.0.1:7004 127.0.0.1:7005
>>> Creating cluster
>>> Performing hash slots allocation on 6 nodes...
Using 3 masters:
127.0.0.1:7000
127.0.0.1:7001
127.0.0.1:7002
Adding replica 127.0.0.1:7003 to 127.0.0.1:7000
Adding replica 127.0.0.1:7004 to 127.0.0.1:7001
Adding replica 127.0.0.1:7005 to 127.0.0.1:7002
M: 3c1113bba7908eb4b1883d46ebdb0eed5fa14df7 127.0.0.1:7000
  slots:0-5460 (5461 slots) master
M: 9d5ae8824bb4c29808b3027ddcfabaadc4426b9a 127.0.0.1:7001
  slots:5461-10922 (5462 slots) master
M: f1b81e5e58bf44b6a883e2b64b4580bd81795c83 127.0.0.1:7002
  slots:10923-16383 (5461 slots) master
S: f75dfc2ee4b0230ece4445d856e87e6326d5c884 127.0.0.1:7003
  replicates 3c1113bba7908eb4b1883d46ebdb0eed5fa14df7
S: 9fbb32027da47562cabbfaled52284bfb1fe6420 127.0.0.1:7004
  replicates 9d5ae8824bb4c29808b3027ddcfabaadc4426b9a
S: 250e8426b8bea24896f4e1392fd09447072fca2f 127.0.0.1:7005
  replicates f1b81e5e58bf44b6a883e2b64b4580bd81795c83
Can I set the above configuration? (type 'yes' to accept): yes
```

redis cluster信息

```
M: 3c1113bba7908eb4b1883d46ebdb0eed5fa14df7 127.0.0.1:7000
  slots:0-5460 (5461 slots) master
M: 9d5ae8824bb4c29808b3027ddcfabaadc4426b9a 127.0.0.1:7001
  slots:5461-10922 (5462 slots) master
M: flb81e5e58bf44b6a883e2b64b4580bd81795c83 127.0.0.1:7002
  slots:10923-16383 (5461 slots) master
S: f75dfc2ee4b0230ece4445d856e87e6326d5c884 127.0.0.1:7003
  replicates 3c1113bba7908eb4b1883d46ebdb0eed5fa14df7
S: 9fbb32027da47562cabbfaled52284bfb1fe6420 127.0.0.1:7004
  replicates 9d5ae8824bb4c29808b3027ddcfabaadc4426b9a
S: 250e8426b8bea24896f4e1392fd09447072fca2f 127.0.0.1:7005
  replicates flb81e5e58bf44b6a883e2b64b4580bd81795c83
Can I set the above configuration? (type 'yes' to accept): yes
>>> Nodes configuration updated
>>> Assign a different config epoch to each node
>>> Sending CLUSTER MEET messages to join the cluster
Waiting for the cluster to join...
>>> Performing Cluster Check (using node 127.0.0.1:7000)
M: 3c1113bba7908eb4b1883d46ebdb0eed5fa14df7 127.0.0.1:7000
  slots:0-5460 (5461 slots) master
M: 9d5ae8824bb4c29808b3027ddcfabaadc4426b9a 127.0.0.1:7001
  slots:5461-10922 (5462 slots) master
M: flb81e5e58bf44b6a883e2b64b4580bd81795c83 127.0.0.1:7002
  slots:10923-16383 (5461 slots) master
M: f75dfc2ee4b0230ece4445d856e87e6326d5c884 127.0.0.1:7003
  slots: (0 slots) master
  replicates 3c1113bba7908eb4b1883d46ebdb0eed5fa14df7
M: 9fbb32027da47562cabbfaled52284bfb1fe6420 127.0.0.1:7004
  slots: (0 slots) master
  replicates 9d5ae8824bb4c29808b3027ddcfabaadc4426b9a
M: 250e8426b8bea24896f4e1392fd09447072fca2f 127.0.0.1:7005
  slots: (0 slots) master
  replicates flb81e5e58bf44b6a883e2b64b4580bd81795c83
[OK] All nodes agree about slots configuration.
>>> Check for open slots...
>>> Check slots coverage...
[OK] All 16384 slots covered.
```

连接某一个redis节点

- 1、cd到redis安装目录：
- 2、运行以下命令：
- `redis-cli.exe -c -p 7000`



电脑 > 系统 (C:) > redis >

名称	修改日期	类型	大小
7000	2018/6/15 星期...	文件夹	
7001	2018/6/15 星期...	文件夹	
7002	2018/6/15 星期...	文件夹	
7003	2018/6/15 星期...	文件夹	
7004	2018/6/15 星期...	文件夹	
7005	2018/6/15 星期...	文件夹	
dump.rdb	2018/6/13 星期...	RDB 文件	1 KB
EventLog.dll	2016/7/1 星期五 ...	应用程序扩展	1 KB
Redis on Windows Release Notes.docx	2016/7/1 星期五 ...	DOCX 文档	13 KB
Redis on Windows.docx	2016/7/1 星期五 ...	DOCX 文档	17 KB
redis.windows.conf	2016/7/1 星期五 ...	CONF 文件	48 KB
redis.windows-service.conf	2016/7/1 星期五 ...	CONF 文件	48 KB
redis-benchmark.exe	2016/7/1 星期五 ...	应用程序	400 KB
redis-benchmark.pdb	2016/7/1 星期五 ...	Program Debug ...	4,268 KB
redis-check-aof.exe	2016/7/1 星期五 ...	应用程序	251 KB
redis-check-aof.pdb	2016/7/1 星期五 ...	Program Debug ...	3,436 KB
redis-cli.exe	2016/7/1 星期五 ...	应用程序	488 KB
redis-cli.pdb	2016/7/1 星期五 ...	Program Debug ...	4,420 KB
redis-server.exe	2016/7/1 星期五 ...	应用程序	1,628 KB
redis-server.pdb	2016/7/1 星期五 ...	Program Debug ...	6,916 KB
redis-trib.rb	2018/6/12 星期...	Ruby File	67 KB
run.bat	2018/6/13 星期...	Windows 批处理...	1 KB

redis事务

- 见word

Redis分布式锁

- 1、获取锁
 - (1) 设置超时时间（防止死锁）
 - (2) 设置客户端标识（标记锁的所有者）
- 2、释放锁
 - (1) 根据客户端标记释放（防止释放别人的锁）
 - (2) 删除key时，要确保事务（防止释放别人刚获取的锁）

分布式锁代码

- 见demo