Redis官网对Redis的定义是：“Redis is an open source, BSD licensed, advanced key-value cache and store”，可以看出，Redis是一种键值系统，可以用来缓存或存储数据。Redis是“Remote Dictionary Server”（远程字典服务）的缩写，提供了字符串（string），列表（list），哈希（hash），集合（set）和有序集合（sorted set）等5种数据结构，这些数据结构使它成为一种便于使用的键值系统。

        作为Redis提供的最简单的数据结构，字符串示意图及其常用命令如下图所示：

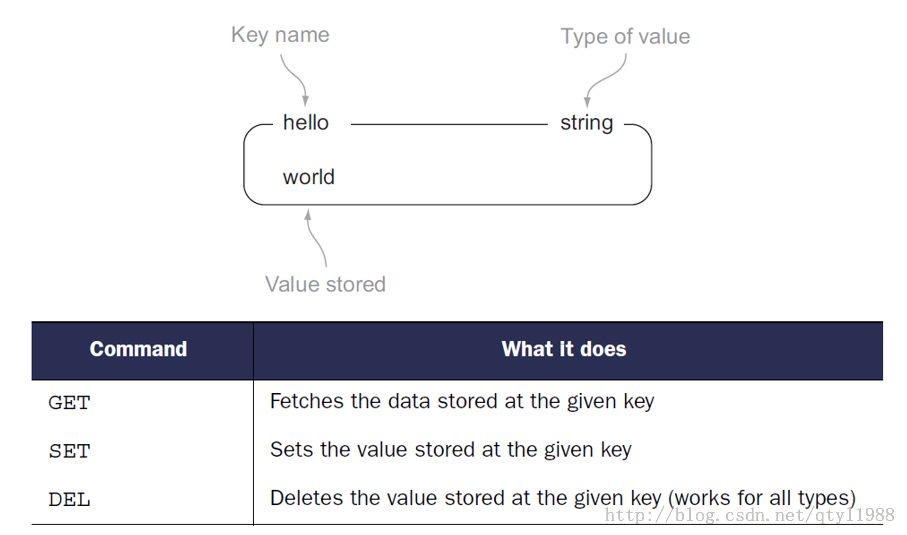


图1

        可见，Redis字符串类型实际上类似于C++中的map，一个键对应一个值。列表类型则是一种相对高级的数据结构，其示意图和常用命令如下图所示：

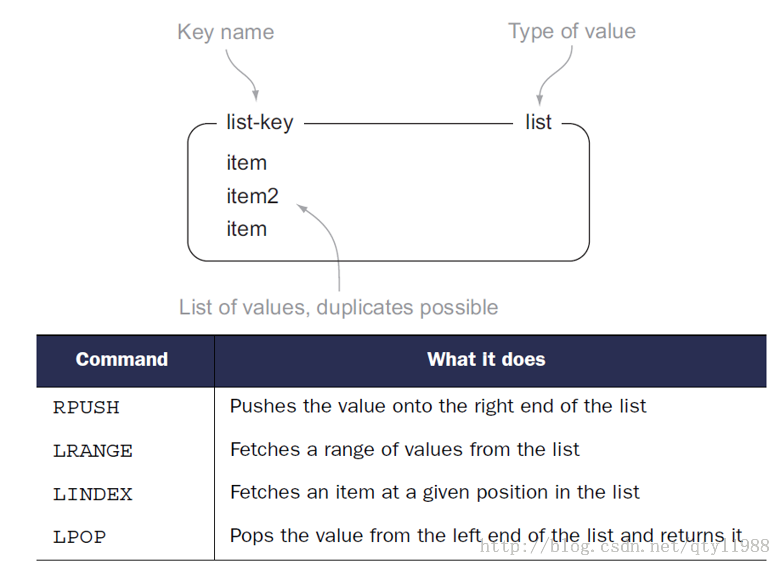


图2

        列表实际上由一个双向链表实现，所以在列表两端插入或删除数据效率极高，可以用于实现类似于微博、人人网上的“新鲜事”功能。 Redis提供的一个更有趣的结构称为哈希，其示意图和常用命令如下图所示：

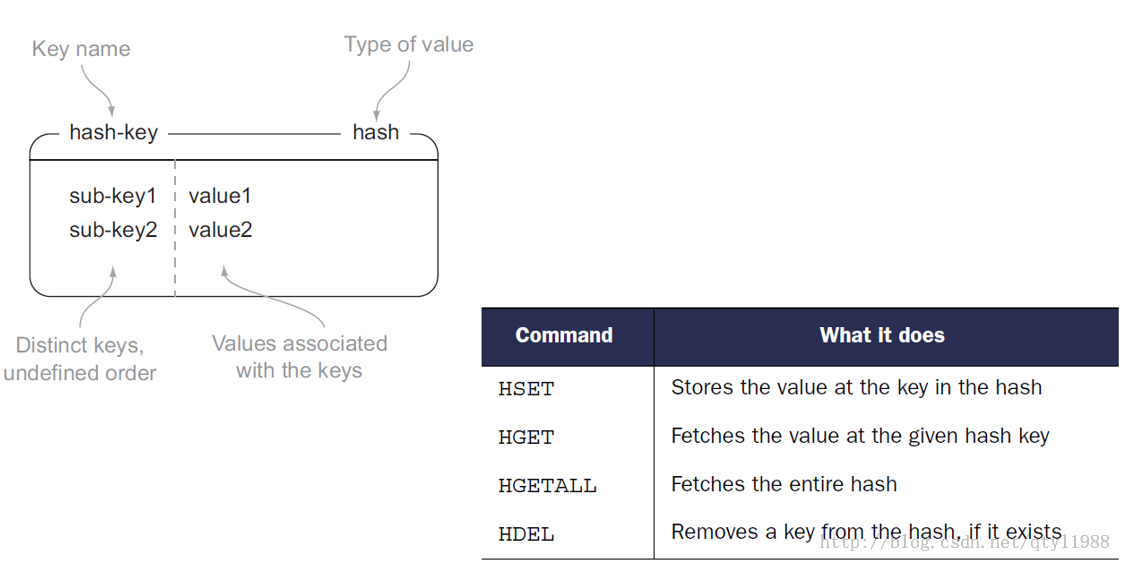


图3

        一个哈希实际上是一组键值对的集合，哈希本身的键是“父键”，哈希中包含的若干字段是“子键”，每个“子键”都有相应的值。“子键”之间并没有一定的联系，但是它们共同组成了一个完整的哈希结构。我们可以把一个哈希看做关系数据库中的一行，哈希的每个子键对应行的一个字段。因此，当把关系数据库中的数据缓存至Redis时，使用哈希结构可能会带来方便。需要注意的是，哈希结构内部的子键之间是没有顺序关系的。Redis提供的另一个无顺序关系的结构是集合，其示意图和常用命令如下图所示：

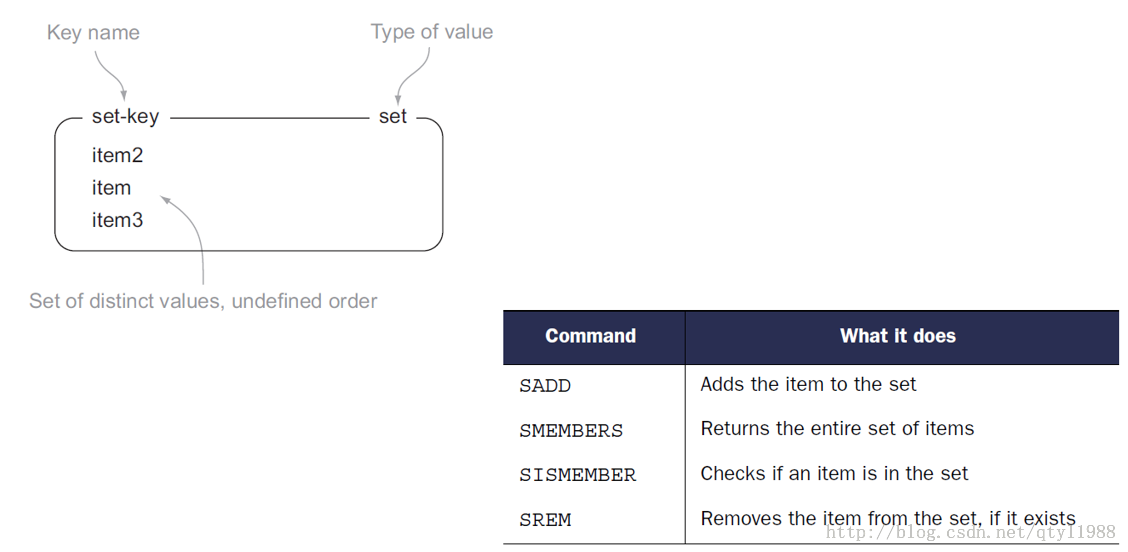


图4

        集合中的数据满足唯一性和无序性。集合在Redis内部是用哈希表实现的，所以插入和删除操作的时间复杂度均为O(1)。Redis为集合提供了求交并差等运算命令，使用起来非常方便。与集合密切相关的另一个数据结构是有序集合，其示意图和常用命令如下图所示：

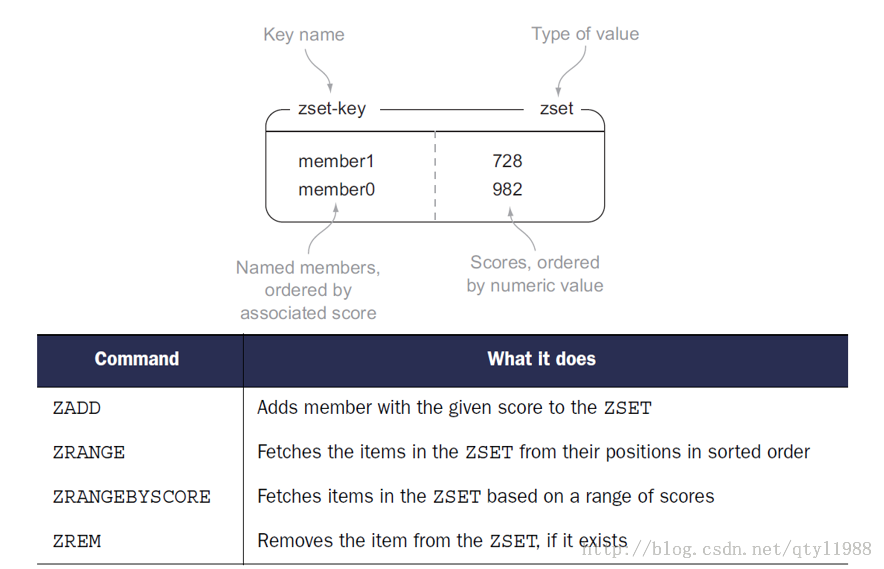


图5

        有序集合与集合的最大差异在于有序集合中的每个数据都有一个对应的分数，集合中的元素就是按照这些分数进行排序的。从结构功能和使用方法上来看，有序集合可能是Redis提供的5种数据结构中最高级的一种。

        Redis是一种内存数据库，这使其在数据存取效率方面表现突出。由于内存中的数据时刻面临丢失的危险，Redis提供了两种持久化机制，及时将内存中的数据写入硬盘。第一种机制为RDB，利用存储快照的方式把内存数据定期写入硬盘；第二种机制为AOF，利用写日志的方式，每执行一条更改Redis数据的命令，就在日志里附加上该命令并保存在硬盘上。Redis默认开启RDB模式，关闭AOF模式，这是因为AOF更加耗时。但是，如果对数据安全要求极高，无法承担任何数据丢失的后果，AOF模式就变成了持久化的首选。

        当然，即使Redis具备持久化机制，但是一旦本地硬盘损坏，数据丢失依然在所难免。所以，Redis又提供了复制功能，作用是将一个主数据库（master）的数据自动同步到多个从数据库（slave），从而尽可能防止数据丢失。Redis主从机制可以简单描述为：从数据库启动后，先向主数据库发送SYNC命令；主数据库接到SYNC命令后就开始保存快照，在此期间，所有发给主数据库的命令都被缓存起来；快照保存完成后，主数据库把快照和缓存的命令一起发给从数据库；从数据库保存主数据库发来的快照文件，并依次执行主数据库发来的缓存命令。在同步过程中，从数据库不会阻塞，它默认使用同步之前的数据继续响应客户端发来的命令。

        事务是Redis提供的特性之一。首先利用MULTI命令通知Redis，接下来的若干命令属于同一事务；然后输入若干命令，这些命令都被放入命令队列而不会被立即执行；最后，利用EXEC命令通知Redis，属于同一事务的所有命令均已输入完成，现在开始执行该事物。属于同一个事务的所有有效命令或者全部被执行，或者全部不执行，且在执行过程中不会插入其他命令。管道（pipeline）是Redis提供的另一个特性。该特性使Redis能够一次性接收多个命令，执行之后再一次性返回结果。这样能够减少客户端与Redis服务器的通信次数，从而降低往返时延。

        除事务和管道以外，Redis还提供了很多其他特性，如设置键的过期时间，利用BLPOP/BRPOP命令实现优先级队列，利用PUBLISH/SUBSCRIBE命令实现消息订阅和发布等，这里不再赘述。

        Redis由ANSI C写成，能够运行在包括Linux和OS X在内的大部分POSIX系统上，官方推荐的操作系统为Linux。Redis没有官方的Windows版本，但是微软移植并维护了一个能够运行在Windows上的Redis并放在github上（https://github.com/MSOpenTech/redis）。

        Redis的编程接口被称为客户端（clients），绝大部分主流编程语言都有官方推荐的客户端。下图中，具有Redis官方推荐客户端的语言标为绿色，没有官方推荐客户端的标为蓝色：

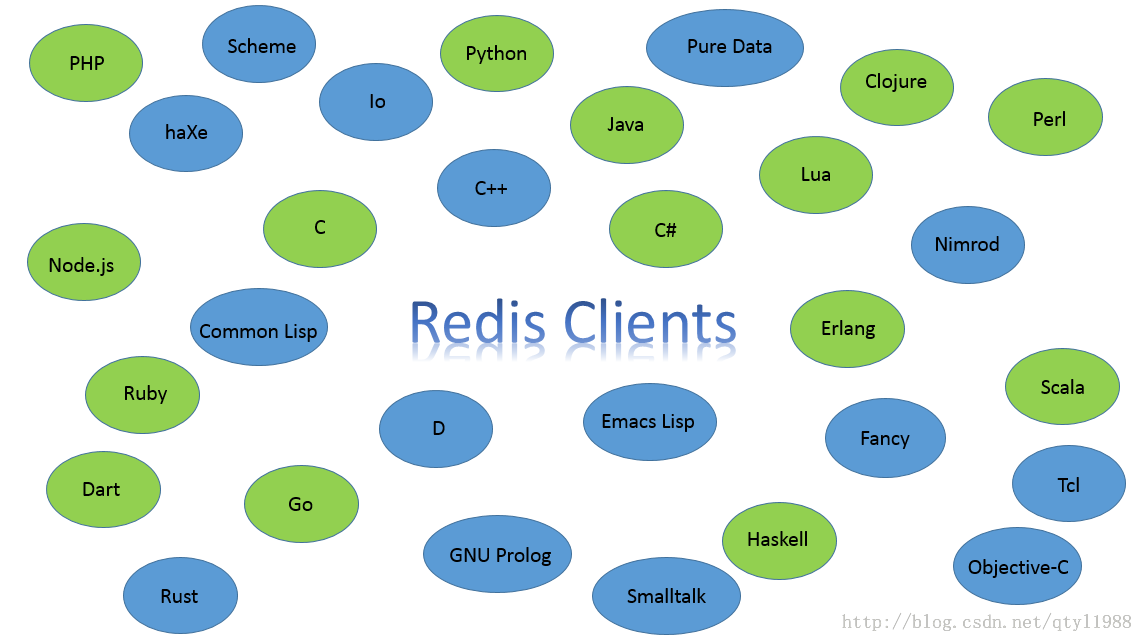


图6

        由于Redis没有官方推荐的C++客户端，因此可以使用C客户端予以代替。官方推荐的C客户端是hiredis，可以从github上找到（https://github.com/redis/hiredis）。

        作为一款功能强大，效率极高的NoSQL数据库，Redis已被包括新浪微博，知乎，Stackoverflow，暴雪和Flickr在内的多家公司成功用于实战。

注：图1-5均来自《Redis in Action》,Josiah L.Carlson, Manning Publications, 2013

参考文献：

1. 李子骅，Redis入门指南，人民邮电出版社，2013年5月第1版

2.Josiah L.Carlson, Redis in Action, Manning Publications, 2013

3. Redis官方网站http://www.redis.io