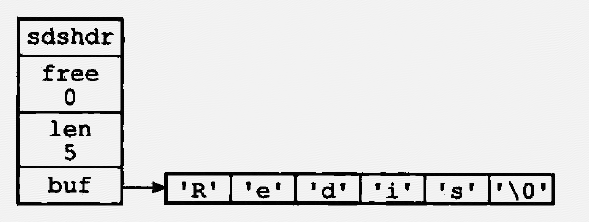
Redis设计与实现

第一部分：数据结构与对象

第二章：简单动态字符串（Simple Dynamic String）

Redis的默认字符串表示为SDS，即简单动态字符串。结构图为：



Len记录了SDS中记录的字符串的长度：查询长度的时间复杂度为O(1)。

buf记录了SDS中存放的数据；buf中的实际长度为len+free+1byte。

free表示了SDS中未使用的空间。实现了空间预分配和惰性空间释放两种策略。

1.空间预分配：

优化了SDS的字符串增长操作，减少了字符串增长带来的内存重分配次数。

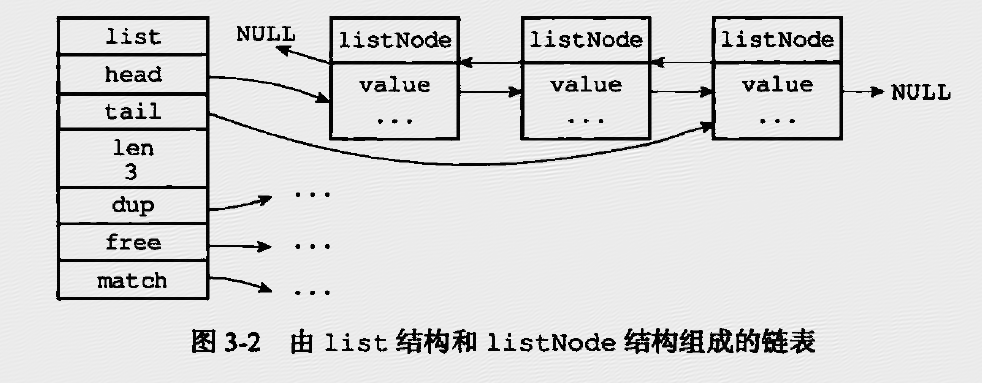
2.惰性空间释放：

优化了SDS的字符串缩短操作，减少了字符串缩减带来的内存重分配次数。

第三章：链表

链表提供了高效的节点重排能力，以及顺序性的节点访问方式，并且可以通过增删节点灵活的调整链表的长度。

链表在redis中应用举例：列表键的底层实现之一就是链表：当一个列表键包含了数量比较多的元素，或者列表中包含的元素都是比较长的字符串时，redis就会使用链表作为列表键的底层实现。发布与订阅，慢查询，监视器等功能也用到了链表，redis服务器使用链表保存多个客户端的状态信息，以及使用链表来构建客户端输出缓冲区。



Redis链表的实现特点：

1.双端：链表节点带有prev和next指针，获取某个节点的前置节点和后置节点复杂度都是O(1)

2.无环：表头节点的prev和表位节点的next指针都指向null，所以对链表的访问都是nulll为终点

3.带表头指针和表尾指针：通过list结构的head和tail指针，获取链表的表头节点和表尾节点的复杂度都为O(1)

4.带链表长度计数器：程序可以使用list的len属性直接获取链表长度，复杂度O(1)

第四章：字典