**《大数据管理》第三章作业 2023.10.20**

**大数据2101班 李嘉鹏 U202115652**

1.简述Memcached系统中数据分布策略的前后两种策略的基本步骤，并说明为何后期采用的第二种策略？DynamoDB系统的数据分布策略和Memcached的第二种策略有何异同？

**早期，Memcached采用标准的分布式方法（对键的存储根据服务器台数的余数进行分散），基本步骤是：①求得键的整数哈希值；②将这个值除以服务器台数，根据其余数来选择服务器；③当选择的服务器无法连接时进行rehash操作——将连接次数添加到键之后再次计算哈希值并尝试连接。**

**后期，Memcached采用的数据分布策略是一致性hash（Consistent Hashing）抽象算法，基本步骤是：①求出服务器节点的哈希值，将其配置到编号为0～232的圆上；②用同样的方法求出存储数据的键的哈希值并映射到圆上；③从数据映射到的位置开始顺时针查找，将数据保存到找到的第一台服务器上；④如果超过232仍然找不到服务器，就保存到第一台服务器上。**

**后期采用第二种策略的原因是当添加或移除服务器时，不会涉及到缓存的重组，代价较小，且数据的分散性较好。**

**DynamoDB系统的数据分布策略和Memcached的第二种策略的相同之处在于：二者都使用了Consistent Hashing算法。**

**不同之处在于：前者是将数据分布在不同节点上，读数据时依据key的哈希值寻找对应节点，节点对应的不再是一个确定的hash值，而是一个hash值的范围，同时还设计了虚拟节点的大小与位置分配策略。而后者是将服务器分布在不同节点上，节点对应的是一个确定的hash值。**

2.简述DynamoDB的数据模型及其主要特点，并写出该系统读写数据的基本步骤序列。

**Dynamo是一个采用键值对存储模型的数据管理系统。在数据组织上，有3层概念，分别是键值对key-value、节点node和实例instance。在数据结构上，主要包括表table、数据项item、属性attribute等组件，其中表是所有数据的集合，每个表包含多个数据项，每个数据项包含一组属性。其主要特点是存放的数据量大、支持高并发、高度可用、查询速度快、采用最终一致性进行数据更新等。**

**对于读写数据，客户端的请求最终交给preference list中的一个节点处理，该节点称为coodinator。**

**写数据（Put）流程：①coodinator生成新的数据版本及vector clock分量；②本地保存新数据；③向preference list中的所有节点发送写入请求；④收到W-1个确认后向用户返回成功。**

**读数据（Get）流程：①coodinator向preference list中所有节点请求数据版本；②等到R-1个答复；③coodinator通过vector clock处理有因果关系的数据版本；④将不相容的所有数据版本返回用户。**

3.（1）写出时钟向量更新的基本步骤；

**VC向量的更新算法分三步：①节点i本地发生一个事件时，VCi[i]加1；②节点i给节点j发送消息m时，将整个VCi存在消息内；③节点j收到消息m时，VCj[k]=max(VCj[k],VCi[k])（其中k是所有的节点），同时VCj[j]加1。**

（2）时钟向量中的服务器时钟为何可能会导致丢失数据，而时钟向量剪枝又存在什么问题？

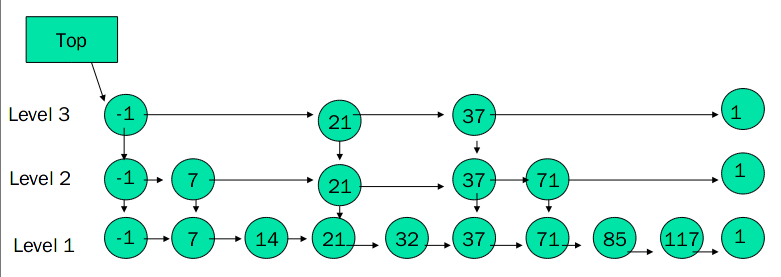
**服务器时钟可能因为网络传输延时而丢失数据。**

**时钟向量剪枝可能存在的问题是某个客户端保持了一个很久之前的向量时钟，并且继承该向量时钟提交了一个数据。此时由于这个很久之前的向量时钟长度大于big\_vclock门限值，因此早已被剪枝，客户端提交的此次数据在服务端无法找到一个祖先，此时Riak会创建一个sibling。**

4.（1）简述Redis系统的数据模型；

**Redis是一个key-value存储系统。每个数据库中的所有数据都是key-value对，其中value支持5种类型，分别是字符串（string）、哈希表（hash）、链表（list）、集合（set）和有序集合（order set）。**

（2）列出在Redis的有序集合的核心数据结构上进行查找的主要步骤，并结合下图示例简述查找数据117的路径及其过程。



**查找的主要步骤：首先从左上角开始，在当前层水平向右查找。如果当前节点的值比目标数据的值小，则继续向右查找；如果当前节点的下一节点（右侧）的值比目标数据的值大，则向下一层继续查找，直到找到某个节点的值等于目标数据的值。**

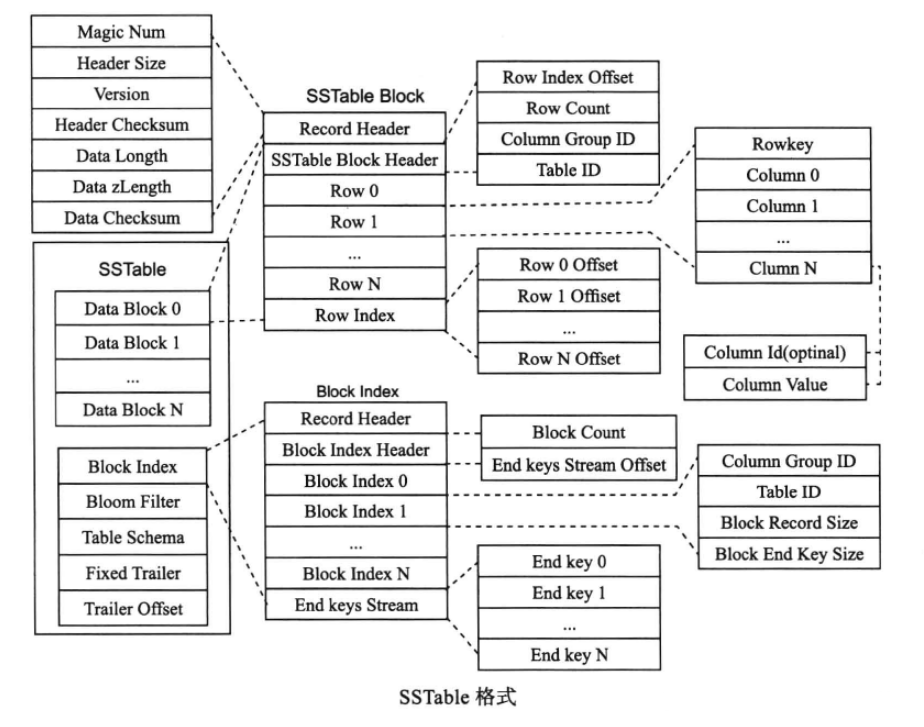
**对于此图，查找数据117的过程是：从Level 3开始，经过-1→21→37，然后向下到Level 2，经过37→71，最后向下到Level 1，经过71→85→117（目标），查询结束。**

5.简述RAMCloud系统的内存数据组织中的两种核心数据结构及其之间的关联。

**RAMCloud采用的简单键值对数据模型中，两种核心数据结构分别是数据项（object）和表（table）。其中数据项是连续存储的，每个数据项对应一个唯一的键；多个数据项组成一个表，表可以跨多个服务器节点存放，数据项只能以整体的方式进行读写。**

6.简要描述Bigtable系统底层存储采用的SSTable的基本结构，并说明为何SSTable本质上是一个两级索引结构？

**SSTable的结构如下图所示。**



**数据按照主键key排序后存放在连续的数据块（Block）中，块之间是有序的。对于每个数据块，其索引由该block最后一行的主键组成（这个索引的存放位置位于SSTable全部块的后面），用于数据查询中具体block的定位。在块索引之后，存放布隆过滤器和表的schema信息，最后存放了固定大小的trailer和trailer的偏移位置。在实际使用中，如果要查找SSTable，首先从子表的索引信息中读取SSTable Trailer的偏移位置，接着获取Trailer信息，从而可以获取块索引的大小和偏移，并将整个块索引加载到内存中。**

**本质上SSTable是一个两级索引结构：块索引+行索引。根据块索引记录的每个block的最后一行的主键，可以通过二分定位查找到目标block。然后将这个block加载到内存中，通过二分查找block中记录的行索引查找到具体的某一行。**