HDFS B+ 索引

郭平雷(index, query) 万强 (B+)

详细介绍和图片见ppt

系统结构

1. Index 索引生成 (tree-index module)
2. Query 查询 (独立于Hive的Server) (query module)

Index分成4个phase

1. 预处理, 将表结构数据整理成 k-v 结构, 如果生成二级索引(对分唯一的列生成索引) 可以在这一步压缩, 或者在sort压缩, 对于主键索引, 不需要压缩
2. 排序, 使用RandomSampler来确定分区, 使用 ToaltOrderPartitioner 来保证全局排序, 生成分区结果和meta, meta 记录了每个分区的id, 起始键值, 键数量
3. 生成索引, 根据排序生成的meta, 在reducer里读取partiton的数据，在本地生成索引文件，上传到HDFS, 同时输出新的meta，在之前的meta上增加索引文件的位置(HDFS上的)
4. 把新的meta合并成metable

Query 分成以下几种情况

1. 点查询
2. 范围查询
3. 聚合查询

最简单的

1. 恰好的范围聚合查询, 比如 5000 < age < 1000 刚好对应 metastable, 就可以直接返回

不需mapreduce的

1. 点查询, 直接复制一个sub-index到本地来查询
2. 在一个sub-index范围内的范围查询, 同上

需要mapreduce的

跨多个sub-index的查询, 根据需要的sub-index个数生成mr, 每个node查询一个sub-index之后合并结果, 查询的结果存到临时文件或者cache或者外部缓存(ie: redis)里, 再由查询server取回

B+ 库 使用

<https://github.com/at15/mellowtech-core> jdk7版本的fork, 原始库需要jdk8