HBase基本说明与性能测试

|  |  |
| --- | --- |
| **HBase** | |
| 数据库类型 | NoSql—列式数据库 |
| 运行所需要的环境 | Linux |
| 是否可以搭建集群 | 天然的分布式数据库，具有自动分片功能 |
| 可扩展性 | 强，无缝支持水平拓展 |
| 插入 | 与设置的参数关系很大，批量插入和单条插入差别大，单台机器能够实现1w~3w之间的插入速度 |
| 更新 |  |
| 删除 |  |
| 查询 | 只支持按照rowkey来查询或者全表扫描 |
| 范围查询 | 不支持 |
| 模糊匹配 | 不支持 |
| 时间范围查询 | 不支持 |
| 分页查询 | 可以做到 |
| 数据库安全性 | 低 |
| 大数据量下的查询响应时间 | 各个数据级别下的响应时间： (均为随机读取，不命中缓存)  1、2万-3万/s  2、3亿-------------------5ms(单行)  3、3亿-------------------124ms(30行) |
| 大数据量下占用的磁盘空间 | 各个数据级别下的磁盘占用空间（以出租车表为例，17个字段，一行200个字节）：  1、1亿-------------------18G（使用GZ压缩） |
| 是否有良好的技术支持 | 社区活跃，但是配置复杂，参数繁多，学习代价比较大 |
| 数据导入和导出 | 有从RDBMS导入数据的工具Sqoop |
| 热备份 |  |
| 异步复制 |  |
| 是否需要商业付费 | 否 |
| 是否开源 | 是 |
| 优点 | 1、  支持高效稳定的大数据存储，上亿行、上百万列、上万个版本，对数据自动分片  2、  列式存储保证了高效的随机读写能力  3、  列数可以动态增长  4、  水平拓展十分容易  5、  拥有良好的生态系统，Sqoop用户数据的导入，Hadoop作为分布式计算平台 |
| 缺点 | 1、  学习复杂  2、  不支持范围查询、条件查询等查询操作 |
| 适用场景 | 1. 实时、高频查询 2. 行随机读操作或者随机写操作、大数据上高并发操作 3. 简单读写 |
|  |  |

基本操作：

| **操作** | **API** | **注意点** | **与版本的关系** |
| --- | --- | --- | --- |
| Get | Table.get | 返回指定行的属性；Scan的第一行 | 若没有指定版本，则返回版本值最大（但可能不是最新的）的数据；可以通过设置MaxVersion的值修改返回的数据条数 |
| Scan | Table.scan | 返回满足条件的多行 | 同上 |
| Put | Table.put | Key存在则更新Key不在则插入;通过 Table.put (写缓存) 或者Table.batch (没有写缓存) | 默认使用系统时间;只要key、column和version相同就可以实现覆盖；插入时可以指定版本 |
| Delete | Table.delete | 1.删除指定列；2.删除列的所有版本；3.删除特定列族的所有列 | 1. 删除操作不会立刻执行，而是给该数据设置墓碑标签，在空间清理的时候再执行死亡数据和墓碑的清除工作；2.通过在 hbase-site.xml.中hbase.hstore.time.to.purge.deletes属性来设置TTL（生存时间） |