29 HBase

目标：能够根据大尺寸非结构化和半结构化数据存储与分析的要求，设计并实现基于HBase的数据存储与分析方案

基本概念

- **HBase(Hadoop Database)**是一个高可靠性、高性能、面向列、可伸缩的**分布式存储系统。**就像Bigtable利用了Google文件系统（File System）所提供的分布式数据存储一样，HBase在Hadoop之上提供了类似于Bigtable的能力。

  HBase是Apache的Hadoop项目的子项目。HBase不同于一般的关系数据库，它是一个适合于非结构化数据存储的数据库。

- **列族**的概念：列族有几个关联比较大的列组成，里面的数据是一起存储的，不同的列族可以分开来存

  列的命名： station : identifier

  每一行都有一个独特的id，按id顺序存储

- 另一个不同的是HBase基于列的而不是基于行的模式

- 按列存：考虑到一个列的额数据相似，编码机制和压缩机制会更高效。
- 非结构化存储：会有空的地方——稀疏矩阵，但在实际存储的时候这些地方不会空着

- 元数据管理简单，表格中数据太大的时候会水平分割成两个region分布式存储，这样也方便version管理

左图描述Hadoop EcoSystem中的各层系统。其中，HBase位于结构化存储层，**Hadoop HDFS为HBase提供了高可靠性的底层存储支持**，

Hadoop MapReduce为HBase提供了高性能的计算能力，Zookeeper为HBase提供了稳定服务和failover机制。

1.面向列：Hbase是面向列的存储和权限控制，并支持独立索引。列式存储，其数据在表中是按照某列存储的，这样在查询只需要少数几个字段时，能大大减少读取的数据量。

2.多版本：Hbase每一个列的存储有多个Version。

3.稀疏性：为空的列不占用存储空间，表可以设计得非常稀疏。

4.扩展性：底层依赖HDFS。

5.高可靠性：WAL机制保证了数据写入时不会因集群异常而导致写入数据丢失，Replication机制保证了在集群出现严重的问题时，数据不会发生丢失或损坏。而且Hbase底层使用HDFS，HDFS本身也有备份。

6.高性能：底层的LSM数据结构和Rowkey有序排列等架构上的独特设计，使得Hbase具有非常高的写入性能。region切分，主键索引和缓存机制使得Hbase在海量数据下具备一定的随机读取性能，该性能真对Rowkey的查询能到达到毫秒级别。

HBase的使用

不同于关系型数据库

▼ 列族的数量尽量不要超过3个！

常用操作：

数据存储

Namespace

限制容量、安全权限、分组，引用的话在表之前再加一个：

Version管理

HBase和RDBMS的对比

- **HBase是一种分布式、面向列的数据存储系统**。

– 表模式反映了物理存储，为高效的数据结构序列化、存储和检索创建了一个系统

– 应用程序开发人员有责任以正确的方式使用此存储和检索

- **典型的RDBMS**

– 具有ACID属性和复杂SQL查询引擎的固定模式、面向行的数据库

– 重点在于强大的一致性、引用完整性、物理层的抽象以及通过SQL语言进行的复杂查询

– 您可以轻松创建二级索引，执行复杂的内部和外部联接，跨多个表、行和列对数据进行计数、求和、排序、分组和分页