# Hbase

## 作用

基于hdfs的.

lsm架构 高写入性能.

海量数据存储

百亿行,没有列的限制,你的表的列你可以弄几百万列...

准时的查询.

几十万行,百万行的数据就不要往里放了.

非关系型

上百万行 上百万列....横向纵向都很大.

面向列的, 不用设计表结构, 列是根据数据来的.实时动态增加列,可以对单独对列进行操作.

高可靠性.

比如那种航道上的实时的船的信息,和交通上的摄像头数据这种也是很大的.

还有就是那种大流量的支付,电商,这些都是非常大的.日志啊啥的.

## 和hive的区别

HDFS是Hadoop的一个模块，主要负责数据存储，而Hadoop另一个模式是MapReduce，用于离线数据计算。

Hive依赖于Hadoop，实际存储是依赖于Hadoop的HDFS模块，计算依赖于Hadoop的Mapreduce模块。

HBase则只是依赖于Hadoop的HDFS模块，是基于Hdfs实现对分布式数据文件的管理，但是跟Mapreduce没有关系，HBase的优势在于实时计算，通过API直接访问HBase，实现实时计算，由于使用的是nosql模式（列式结构），从而提高了查找性能，使其能运用于大数据场景，这是它跟MapReduce的区别。

Hive的数据源既可以来自HDFS，也可以来自于HBase。

————————————————

共同点:

1.hbase与hive都是架构在hadoop之上的。都是用hadoop作为底层存储duzhi

区别：

2.Hive是建立在Hadoop之上为了减少MapReduce jobs编写工作的dao批处理系统，HBase是为了支持弥补Hadoop对实时操作的缺陷的项目 。

3.想象你在操作RMDB数据库，如果是全表扫描，就用Hive+Hadoop,如果是索引访问，就用HBase+Hadoop 。

4.Hive query就是MapReduce jobs可以从5分钟到数小时不止，HBase是非常高效的，肯定比Hive高效的多。

5.Hive本身不存储和计算数据，它完全依赖于HDFS和MapReduce，Hive中的表纯逻辑。

6.hive借用hadoop的MapReduce来完成一些hive中的命令的执行

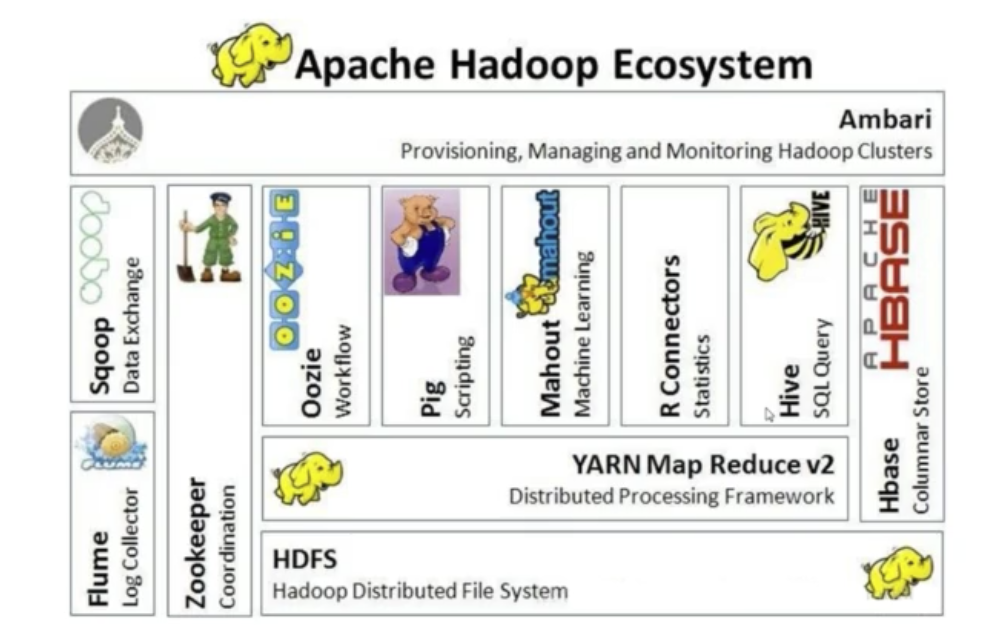
7.hbase是物理表，不是逻辑表，提供一个超大的内存hash表，搜索引擎通过它来存储索引，方便查询操作。

8.hbase是列存储。

9.hdfs作为底层存储，hdfs是存放文件的系统，而Hbase负责组织文件。

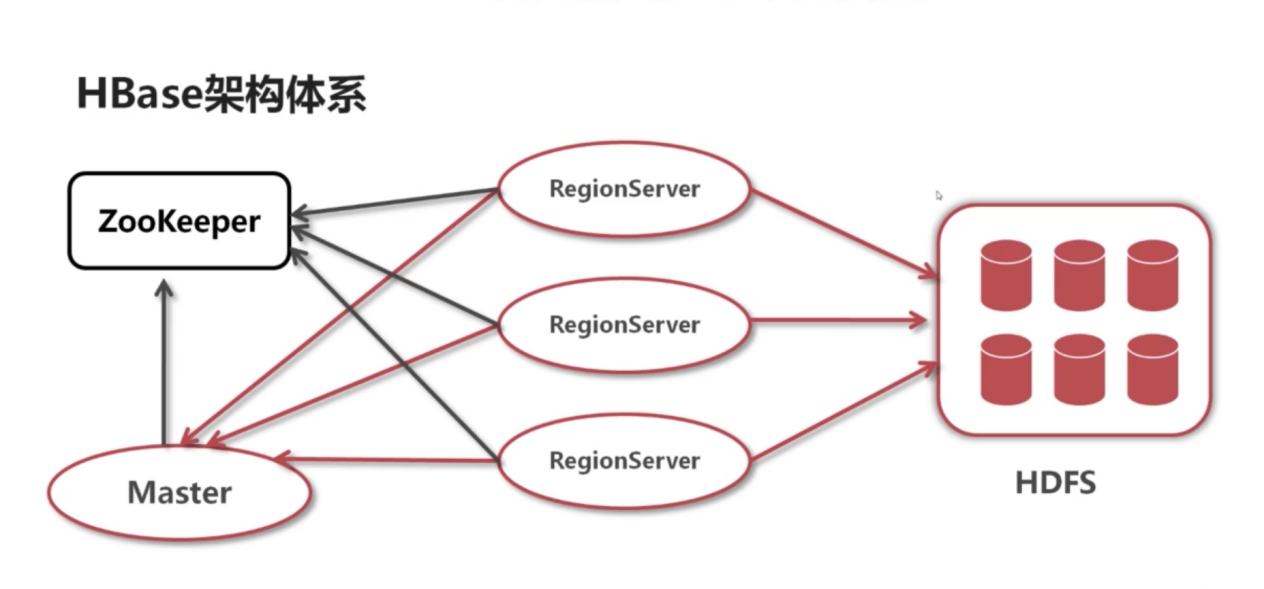
10.hive需要用到hdfs存储文件，需要用到MapReduce计算框架。

## hbase在hashdoop中的定位



唯一的真实的数据存储

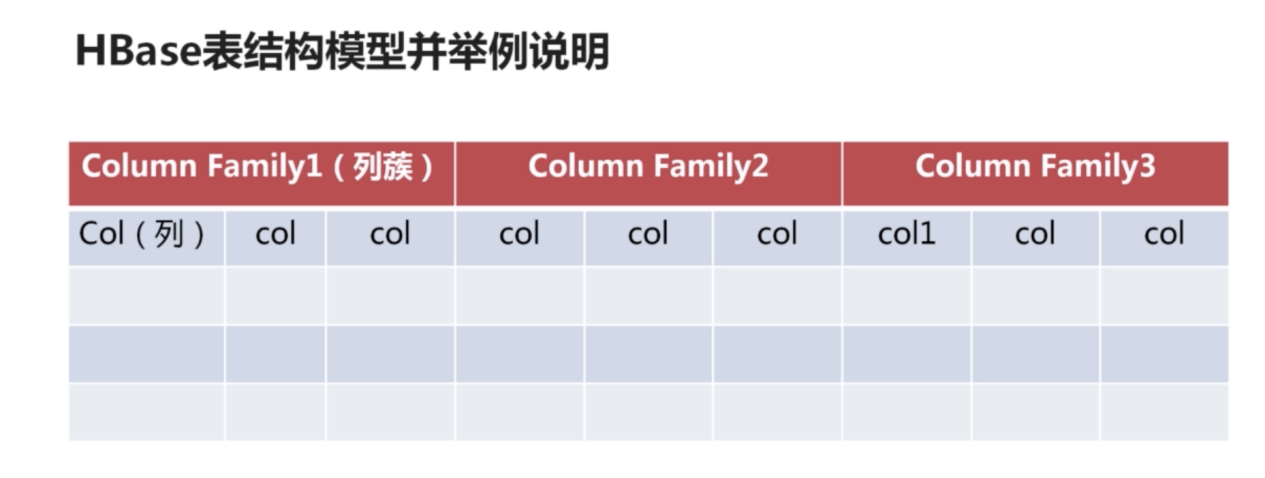
## 架构



regionserver就是region的服务, 什么是region呢,就是把一个很大的表给分区了.

master就是管理这些region的,维护这些region的信息,那些数据存到了哪个region里.

## 列簇





## 表数据



列簇越少越好

这个和头条的user表不是一个意思么,原来这就是所谓的面向列,头条的user表也是面向列的,列也是可以无限扩展的.

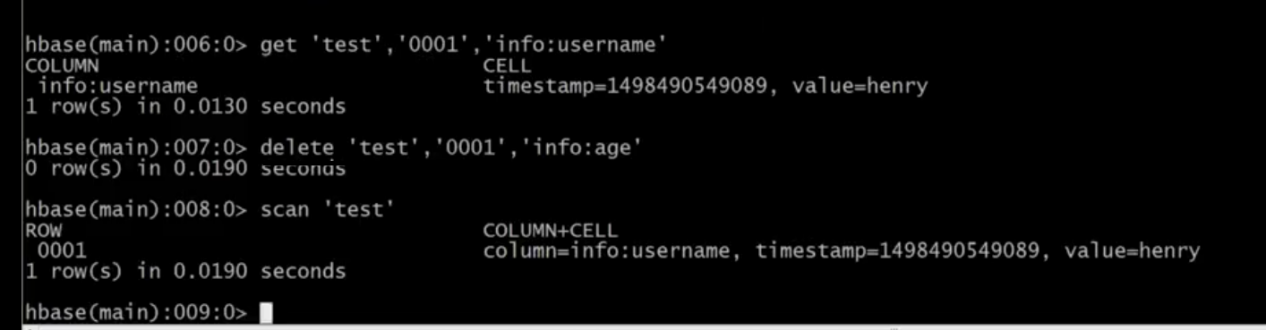
数据自动切分会,mysql是不会的.

这么多好处,,缺点就是不支持复杂的查询,不支持条件查询,只支持匹配查询.

## 命令







Get 表名 标志id 列簇名:列名

获取某个用户的某一列的数据

Delte 表面 标志id 列簇名:列名

删掉某个用户的某一列

Scan 表名 查看表中所有的数据