# HBASE学习

1. 是什么

HBASE是一个hadoop应用，是专为分布式数据准备的数据库

1. 为什么

需要满足大量数据对RDBMS性能的要求（但是HBASE并不是关系数据库）

1. 怎么办

# 概念学习

1. 简要的总结：最基本的单位是列，一列或多列形成一行，并由唯一的行键来确定存储。反过来，一个表中有若干行，其中每列可能有多个版本，在每个单元格中存储了不同的值。（行序按照行键字典序进行排序）
2. HBASE是以列簇为组织形式的，只要预先给出了一个列簇，就可以动态的增加该列簇的列；
3. HBASE的实现：正如hadoop的分布式方法一样，HBASE也有自己的主从机结构以及其管理方式；其中，主机叫作master，而从机叫作regionserver
4. 目标：将HBASE的可伸缩性变得简单，以不同的方法访问HBASE，了解HBASE架构的细节，包括存储格式、预写日志、后台进程等；在HBase中集成MapReduce框架；了解如何调节集群、设计模式、拷贝表、导入批量数据和删除节点等。并且在读源码的同时掌握JAVA编码技巧。
5. 协处理器：在服务端地址空间执行来自客户端的代码。

## 结构

1、自动分区，region本质上是以行键排序的连续存储区间，太大了系统会自动拆分，太小的时候则会自己合并。一个region服务器可以加载多个region，但是一个region只能被一个region服务器加载

2、更新数据流程，每次更新时，都会先将数据记录在提交日志中，然后再写入内存的memstore中，内存保存的数据超过一定值的时候会刷写到磁盘，这时会用新的memstore获取更新数据而将满的memstore刷写入磁盘。

3、删除数据是先将数据标记为删除的，此时将对用户不可见，在执行一次major合并的时候会忽略删除的数据，从而达到更新的目的。（major合并是带压缩的，而minor合并不）

4、Hbase中有三个主要组件：客户端库、一台主服务器、多台region服务器。可以动态的增加或移除region服务器，以适应不断变化的负载要求。主服务器利用ZooKeeper管理

# 深入代码

## 一、客户API基础知识

Hbase保持了行的原子性，因此，多个客户端同时对行进行操作可能会出现锁的问题，推荐的方法是采用批处理的办法。

只创建一个HTable的实例，或者使用HTablePool类，因为创建有代价。

同时，应该为每个线程创建一个实例，并且一般在程序开始时创建。

### 1、CRUD（create、read、update、delete操作）

行键必须指定，

它最好代表了实际的意义，如订单号，用户等；

它是一个btye[]；

它的内容可以是简单的数字，也可以是一个UUID（Universally Unique Identifer）

提供了一些转化方法供别的类型转化为btye[];

# 要点总结

1. 以前因为技术的原因无法分析每个数据点，而现在，不分析所有数据点的公司将失去竞争力。
2. 大数据为什么大：数十亿行\*数百万列\*数千个版本=TB级或PB级的存储。

# 资料总结

1、论文：BigTable：A Distributed Storage System for Structured Data