



下载APP



## 复习课（六）| Hive

2021-12-01 徐文浩

《大数据经典论文解读》

课程介绍 &gt;



讲述：王惠

时长 07:46 大小 7.13M



你好，我是黄金。今天这期复习课，我们一起来回顾和总结下，Facebook 在 2009 年所发表的 Hive 的论文。

### Hive 介绍

2008 年以前，Facebook 的数据仓库构建在商用的 RDBMS 上。随着数据量的增加，一些需要每天执行的批处理作业，单次运行时间已经超过了一天，因此优化数据仓库成为迫在眉睫的任务。后来 Facebook 把数据搬到了 Hadoop 上，原来需要花一天多才能跑完的作业，现在花几个小时就能跑完，执行速度快了很多。



不过使用 Hadoop 并不容易，尤其是对那些不熟悉 MapReduce 的人来说，即使是写一个简单的查询，也要花上几个小时。工程师们就在想，能不能把分析师熟悉的表、分区之类

的 SQL 概念，引入 Hadoop 世界，一来可以让分析人员使用自己已经掌握的工具，二来可以把编写脚本的时间，从几个小时缩短成几分钟。

于是，就有了我们今天要介绍的 Hive，而它所使用的查询语言就是和 SQL 非常类似的 HiveQL。

## 数据模型

想要把 MapReduce 任务变成 SQL 语句，需要先把数据结构化，才能用 SQL 语句查询。像传统的 RDBMS 一样，Hive 的数据通过**表**来抽象，数据由多行记录构成，一行记录包含多个字段，每个字段有特定的类型。

Hive 天然支持的基础类型包括整数、浮点数和字符串，天然支持的复杂类型包括数组、字典和结构体，这些类型足以让我们应付大多数业务场景。同时，Hive 也允许用户自定义字段类型，便于我们应付特殊的业务场景。

和 RDBMS 不同的地方在于，Hive 通常采用宽表结构，把一个对象的所有字段保存在一张表中。这么做的目的是为了**避免表与表之间的 Join 操作**，Join 操作一般需要跨服务器交互，成本高而且性能差。

Hive 的数据表支持**分区和分桶**操作。

分区是根据指定字段的值，把数据分到不同的区，每个值一个区。如果以后根据这个字段的值来查询数据，就可以只扫描对应分区的数据，而不用读取全部数据。一般我们会按日期对数据进行分区，因为常常需要分析指定时间段的数据。

分桶是根据指定字段的值，把数据分到固定个数的桶中，比如指定 32 个桶，通过哈希的方式让数据分布到不同的桶中。分桶可以用于采样分析，根据某几个桶的数据快速估算数据的整体情况。

那么，对比 Hive 和 Bigtable 的分区方式，Bigtable 采用动态分区，Hive 采用静态分区，这是否意味着 Hive 的分区方式不够高级呢？

我想这和它们要解决的问题领域相关。**Bigtable 善于处理随机读写**，一次操作仅仅涉及几条记录，动态分区的方式有助于快速定位记录所在的分区，分散访问热点。**Hive 主要执行**

**批处理作业**，一次扫描大量连续的数据，数据只要按照过滤条件静态分区即可。这是因为过滤条件下的所有数据都要读，不会因为横向拆分成更多文件，就可以少读一些数据。

## 数据存储

Hive 的**数据表存储在 HDFS 上**，一张表对应了 HDFS 的一个目录。对数据表分区，就是以分区字段和值为目录名，创建子目录，分区数据保存在子目录中。对数据表分桶，就是把原来的一个文件，拆分成多个文件，有几个桶就拆成几个文件。至于分桶后的文件放在哪里，就看是在表上分桶，还是在分区上分桶。如果是在表上分桶，就存储在表的根目录，如果是在分区上分桶，就存储在对应的分区目录。

Hive 通过**序列化器 Serializer 和反序列化器 Deserializer**，指定应用层数据模型和存储层数据模型之间的转换规则。比如以什么作为行分隔符，又以什么作为字段分隔符。

Hive 通过**文件格式**指定数据如何存储到文件中。比如是以文本格式存储，还是以二进制格式存储，还是使用列式存储。

## 系统架构

Hive 的整个系统架构并不复杂，总共分成三个部分。

第一部分是**对外接口**。对外接口包括命令行、Web 界面、Thrift 接口，以及 JDBC 和 ODBC 驱动。通过 Thrift 接口，Hive 可以支持其他语言的客户端的调用。

第二部分是**驱动器**。驱动器管理了 HiveQL 语句的生命周期，它通过编译器和优化器创建执行计划，通过执行器执行计划。

第三部分是 **Metastore**。Metastore 存储了 Hive 的各种元数据，包括表的名称和位置、列的名称和类型等。

使用 HiveQL，而不是 MapReduce，除了简单易用，我们还能够充分利用系统来提升执行性能。当分析师分析数据时，难免要把不同的表 Join 到一起执行查询操作，而作为声明式语言的 HiveQL，会自动找到合适的 Join 方式。

如果是一张大表和一张小表 Join，应该先找个规则，把大表的数据分区，为每一个分区启动一个任务，把小表的数据拷贝到每个任务中执行 Join 操作，这种方法被称为**广播哈希 Join**。如果是一张大表和另一张大表 Join，应该通过 Join 字段，把两张表的数据按相同的方式分区，在每个分区上分别执行 Join 操作，这种方法被称为**分区哈希 Join**。

Hive 的优化器，会自动识别应该采用哪一种方法执行 Join 操作。随着 Hive 的版本升级，已经投入使用的 HiveQL 还可以利用新版的优化器获得更好的性能，这些好处是直接写 MapReduce 任务所不具备的。

当然，HiveQL 最终还是要翻译成 MapReduce 任务，每一个 Mapper 或 Reducer 任务读取数据后，都需要 Metastore 中的元数据信息来解析数据。为了降低 Metastore 被请求的频率，驱动器在生成执行计划时，就已经把所有需要用到的元数据，写到了执行计划中，Mapper 或 Reducer 任务只需要通过执行计划文件，就能得到元数据信息。

通过观察 Hive 的系统架构，我们可以发现，**Hive 并没有对 MapReduce 形成强依赖**，所以我们今天可以看到 Hive on Spark 这样的项目。借助 Spark，HiveQL 能够执行得更快。这是因为 Spark 采用了内存管理中间数据，不像 MapReduce 一样，每个任务都要把结果写到文件系统。而且 Spark 是数据流引擎，不用等到上一个任务全部结束，才开始执行下一个任务，只要上一个任务有输出一点内容，下一个任务就可以开始执行。

好了，到这里 Hive 的核心内容我们就复习完了。如果你有其他关于 Hive 的论文的学习思考，期待在留言区一起探讨。

分享给需要的人，Ta 订阅后你可得 **20 元现金奖励**

 生成海报并分享

 赞 1     提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇    复习课（五）| Chubby

下一篇    25 | 从 S4 到 Storm（一）：当分布式遇上实时计算

训练营推荐

# Java 学习包免费领<sup>NEW</sup>

面试题答案均由大厂工程师整理

阿里、美团等  
大厂真题

18 大知识点  
专项练习

大厂面试  
流程解析

可复用的  
面试方法

面试前  
要做的准备

## 精选留言

写留言

由作者筛选后的优质留言将会公开显示，欢迎踊跃留言。