

一页纸精华 | Hive

原创 2016-03-03 牛家浩 中兴大数据

>>>> 这是 **中兴大数据** 第212篇原创文章

要入门大数据，最好的办法就是理清Hadoop的生态系统。中兴大数据公众号将推出“一页纸精华”栏目，将用最精炼的语言，陆续为你介绍Hadoop生态系统的各个组件。本期为你介绍Hadoop数据仓库工具Hive。



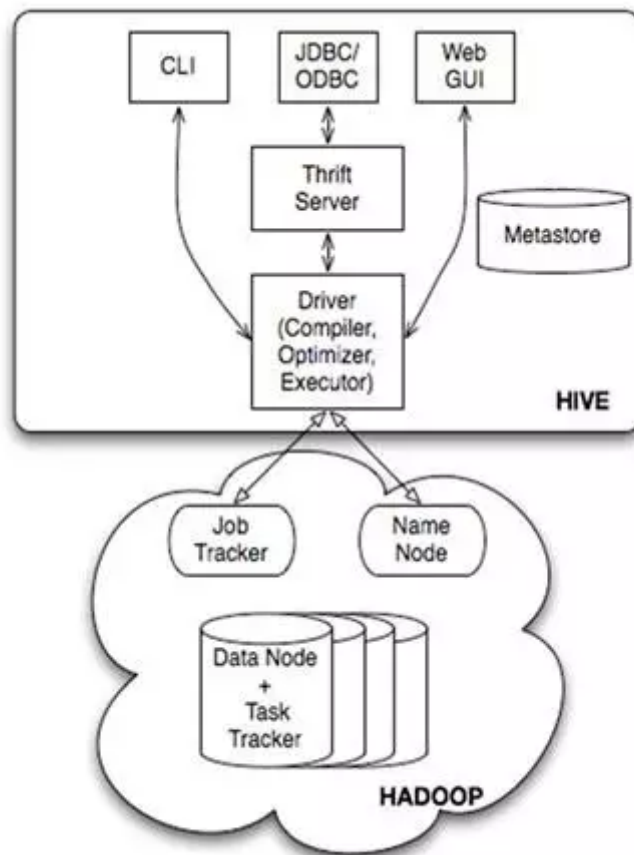
Hive是Apache Hadoop的正式子项目，可以将结构化的数据文件映射为一张数据库表，并提供简单的SQL查询功能，可以将SQL语句转换为MapReduce任务进行运行。其优点是学习成本低，可以通过类SQL语句快速实现简单的MapReduce统计，不必开发专门的MapReduce应用，十分适合数据仓库的统计分析。

Hive 是建立在 Hadoop 上的数据仓库基础构架。它提供了一系列的工具，可以用来进行数据提取转化加载（ETL），这是一种可以存储、查询和分析存储在 Hadoop 中的大规模数据的机制。Hive 定义了简单的类 SQL 查询语言，称为HQL，它允许熟悉 SQL 的用户查询数据。同时，这个语言也允许熟悉 MapReduce 开发者的开发自定义的 mapper 和 reducer 来处理内建的 mapper 和 reducer 无法完成的复杂的分析工作。它具备下面几个基本特性：

1. **查询语言**：由于 SQL 被广泛的应用在数据仓库中，因此，专门针对Hive 的特性设计了类 SQL的查询语言 HQL。熟悉SQL 开发的开发者可以很方便的使用 Hive 进行开发。
2. **数据存储位置**：Hive 是建立在 Hadoop 之上的，所有 Hive 的数据都是存储在 HDFS 中的。而数据库则可以将数据保存在块设备或者本地文件系统中。
3. **数据格式**：Hive 中没有定义专门的数据格式，数据格式可以由用户指定，用户定义数据格式需要指定三个属性：列分隔符（通常为空格、“\t”、“x001”）、行分隔符（“\n”）以及读取文件数据的方法（Hive 中默认有三个文件格式 TextFile，SequenceFile 以及 RCFile）。由于在加载数据的过程中，不需要从用户数据格式到 Hive 定义的数据格式的转换，因此，Hive 在加载的过程中不会对数据本身进行任何修改，而只是将数据内容复制或者移动到相应的 HDFS 目录中。
4. **执行**：Hive 中大多数查询的执行是通过 Hadoop 提供的 MapReduce 来实现的（类似 select * from tbl 的查询不需要 MapReduce）。而数据库通常有自己的执行引擎。
5. **执行延迟**：之前提到，Hive 在查询数据的时候，由于索引功能还不够完善，需要扫描整个表，因此延迟较高。另外一个导致Hive 执行延迟高的因素是 MapReduce 框架。由于 MapReduce 本身具有较高的延迟，因此在利用 MapReduce 执行 Hive 查询时，也会有较高的延迟。相对的，数据库的执行延迟较低。当然，这个低是有条件的，即数据规模较小，当数据规模大到超过数据库的处理能力的时候，Hive 的并行计算显然能体现出优势。
6. **可扩展性**：由于 Hive 是建立在 Hadoop 之上的，因此 Hive 的可扩展性是和 Hadoop 的可扩展性是一致的而数据库由于 ACID 语义的严格限制，扩展行非常有限。目前最先进的并行数据库ORACLE 在理论上的扩展能力也只有 100 台左右。
7. **数据规模**：由于 Hive 建立在集群上并可以利用 MapReduce 进行并行计算，因此可以支持很大规模的数据；对应的，数据库可以支持的数据规模较小。

Hive体系结构

Hive架构如下图所示：



Hive系统总体上分为以下部分：

- **UI**：用户提交查询请求与获得查询结果。包括三个接口：命令行（CLI）、Web GUI（Hue）和客户端。
- **Driver**：总体功能接受查询请求经过处理返回查询结果。在它内部又再分为2个组件：**Compiler**：编译器，分析查询SQL语句，在不同的查询块和查询表达式上进行语义分析，并最终通过从Metastore中查找表与分区的元信息生成执行计划。**ExecutionEngine**：执行引擎，执行由Compiler创建的执行计划，执行引擎管理不同阶段的依赖关系，通过MapReduce执行这些阶段。
- **Metastore**：元数据储存，元数据存储在MySQL或derby等数据库中。元数据包括Hive各种表与分区结构化信息，包括列与列类型信息，序列化器与反序列化器，从而能够读写HDFS中的数据。

Hive数据模型

Hive的数据模型包括：database、table、partition和bucket：

1. **Database**：相当于关系数据库里的命名空间（namespace），它的作用是将用户和数据库的应用隔离到不同的数据库或模式中，Hive提供了create database dbname、use dbname以及drop database dbname这样的语句；
2. **表(table)**：Hive的表逻辑上由存储的数据和描述表格中的数据形式的相关元数据组成。表存储的数据存放在分布式文件系统里，例如HDFS，元数据存储的关系数据库里，当我们创建一张Hive的表，还没有为表加载数据的时候，该表在分布式文件系统，例如HDFS上就是一个文件夹（文件目录）。Hive里的表有两种类型一种叫托管表，这种表的数据文件存储在Hive的数据仓库里，一种叫外部表，这种表的数据文件可以存放在Hive数据仓库外部的分布式文件系统上，也可以放到Hive数据仓库里（注意：Hive的数据仓库也就是HDFS上的一个目录，这个目录是Hive数据文件存储的默认路径，它可以在Hive的配置文件里进行配置，最终也会存放到元数据库里；
3. **桶(bucket)**：分桶是将数据集分解成更容易管理的若干部分的另一个技术，上面的table和partition都是目录级别的拆分数据，bucket则是对数据源数据文件本身来拆分数据。使用桶的表会将源数据文件按一定规律拆分成多个文件，要使用bucket，我们首先要打开Hive对桶的控制。

Hive应用场景

Hive提供数据提取、转换、加载功能，并可用类似于SQL的语法，对HDFS海量数据库中的数据进行查询、统计等操作。形象地说Hive更像一个数据仓库管理工具，适用于结构化数据的应用，读多写少的应用，响应时间要求不高的场合。

Hive常用于以下几个方面：

1. 数据汇总（每天/每周用户点击数，点击排行）
2. 非实时分析（日志分析，统计分析）
3. 数据挖掘（用户行为分析，兴趣分区，区域展示）

