使用 Hive 构建数据仓库

省钱的数据仓库

Peter J. Jamack, 大数据分析顾问, Peter J Jamack

简介： 各个组织已围绕数据仓库展开了数年的争斗。您如何构建它？您可以集成哪些数据？您应该使用 Kimball 还是 Inmon，企业信息工厂 (CIF) 还是数据集市？这些问题已存在多年，甚至数十年。对于大数据，问题变得复杂得多，比如一个数据仓库是否够用？答案取决于具体企业。人们声称 Hive 是 Hadoop 的数据仓库。尽管从某个层面上讲这是真的，但这种说法也有几分虚假。不过，有时您必须使用您可以使用的工具，就此而言，可以将 Hive 用作一个数据仓库。

发布日期： 2013 年 8 月 16 日
级别： 中级
原创语言： 英文

有 3 个家伙来到了一家企业。第一个（数据仓库）身材魁梧：他带来了历史和经验，而且能言会道，所说的大部分话都是真的。但是，在许多方面，它有些自我膨胀，在另一些方面又有些铺张浪费，而且人们厌烦了各种结果的代价。Apache Hadoop 进入了同一栋建筑，声称要接管整个市场。他大肆鼓吹大数据、速度、数据量、种类以及一堆 *v* 开头的词汇，这些词汇在市场营销计划之外没有多大意义。他漫不经心地说着分析、预测等等。而且他要价很低。于是人们开始停下来倾听。

Apache Hive 在屋外徘徊，他没有打算和其他人争论。他希望与 Hadoop 合作，但不同于 Hadoop，他不希望将数据仓库抛在路边。Hive 拥有数据仓库功能，但在商业智能 (BI) 和分析上有一些限制。它具有数据库的潜力，但也具有关系数据库管理系统 (RDBMS) 和结构化查询语言 (SQL) 方面的限制。它更加开放和诚实。它与数据仓库密切相关，与 RDBMS 也密切相关。但它从未站出来声称它并不像初看起来那么简单。Hadoop 打断了谈话，声称它是 Hadoop 领域的数据仓库。Hadoop 似乎让出了最优秀营销公关代表的地位，在一次简单的对话之后，结果变成了是 Hive 和 Hadoop 在拯救世界。这种描述很吸引人，也很有趣。但它是真的吗？ 有几分相似。

数据仓库

构建一个真正的数据仓库可能是一个庞大的工程。有许多不同的设备、方法和理论。最大的共同价值是什么？事实是什么，哪些主题与这些事实相关？以及您如何混合、匹配、合并和集成可能已存在数十年的系统与仅在几个月前实现的系统？这还是在大数据和 Hadoop 之前。将非结构化、数据、NoSQL 和 Hadoop 添加到组合中，您很快就会得到一个庞大的数据集成项目。

描述一个数据仓库的最简单方式是，认识到可以将它归结为星形模式、事实和维度。您如何创建这些元素，决定权在您手上 — 通过暂存数据库；动态提取、转换、加载流程；或者集成辅助索引。当然，您可以构建一个包含星形模式、事实和维度的数据仓库，使用 Hive 作为核心技术，但这并不容易。在 Hadoop 世界外部，这会成为一个更大的挑战。与其说 Hive 是一种合法的数据仓库，倒不如说它是一个集成、转换、快速查找工具。该模式可能像是数据仓库，但适用性表明它不是 RDBMS。那么为什么使用它？

星形模式是什么

想象一颗星星 — 具有一个中心和多个指向不同方向的 “手臂”。中心是动力之源或事实表。所有手臂都指向不同维度。许多数据仓库有一个事实表和多个维度。

事实表包含您可以加权或计算的任何数据。在此示例中，您拥有棒球统计数据，比如跑垒、全垒打、击球率等。您可以计算、增加、减去或乘以这些列。

维度更加以主题为基础。在此示例中，您有运动员信息维度、时间和日期维度，等等。通常没有计算或加权多个维度中的列。

在此示例中，将一个维度表与一个事实表连接的键是 playerID。

简单来讲，有时您需要使用摆在面前的工具。

任何从事过一段时间的 IT 工作的人都可能告诉您，适合一项工作的正确工具并不总是能够用到。或者，正确的工具虽然用得到，但为了削减成本会阻碍使用该工具。有时企业政治学发挥着重大作用。无论什么原因，我们大部分人在很多情形下被迫使用可能并不是最适合其工作的工具来构建、设计和开发。

在我参与的许多项目中，我不得不使用 Hive 作为数据库、作为数据仓库以及作为缓慢变化的系统。这很有挑战性，但偶尔会令人生厌。有时，您不得不摇头并想知道为什么。但在一天结束时，您仍然需要让它工作。如果需要在 Hive 中构建和使用某个数据仓库，而

且需要使用缓慢变化的维度和更新，并协调旧数据，那么您必须这么做。重点并不总是提供最佳的工具，而是创建最适合您工作的工具。

回页首

Hive

由于 Hive 的类 SQL 功能和类数据库功能，它向非编程人员开放了大数据 Hadoop 生态系统。它常被描述为一个构建于 Hadoop 之上的数据仓库基础架构。这是一种部分真实的表述（因为您可将源数据转换为星形模式），但在创建事实表和维度表时，它更关乎设计而不是技术。

尽管如此，Hive 并不真正是一个数据仓库。它甚至并不真正是一个数据库。您可以使用 Hive 构建和设计一个数据仓库，也可以使用 Hive 构建和设计数据库表，但存在的一些限制需要提供许多解决办法，并且将会带来一些挑战。

例如，索引在 Hive 中有一些限制。如何克服这个问题呢？您可以使用 `org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler` 函数在 Hive 中创建索引。Hive 和缓慢变化的维度并不总是可能实现。但是如果构建暂存表和使用一定量的连接（而且计划添加一个新表，转储旧表，并且只保留最新、更新表用于比较），则可能实现它们。

连接到 Hive 的外部报告或分析系统是一个巨大的问题。甚至对于 JDBC 连接，也仅限于连接到默认数据库。人们在寻求更多的经过改进的元数据，而且 Apache HCatalog 等工具正在帮助将各种服务连接到 Hive 元存储。在未来，如果利用得当，这可能是一个重大的增值区域。

所以，尽管 Hive 不是一个可靠的数据仓库或数据库，但仍然可以使用一些方法将 Hive 用作数据仓库或数据库。只是需要做一些工作和利用一些解决办法将 Hive 打造成这样的系统。为什么您要再次经历这一过程？因为您必须使用手头的工具并让它们发挥作用。

回页首

示例：为棒球信息构建一个数据仓库

InfoSphere BigInsights Quick Start Edition

InfoSphere BigInsights Quick Start Edition 是 InfoSphere BigInsights（IBM 的基于 Hadoop 的产品）的一个免费的可下载版本。使用 Quick Start Edition，您可以尝试使用 IBM 开发的特性来提高开源 Hadoop 的价值，比如 Big SQL、文本分析和 BigSheets。引导式学习可让您的体验尽可能地顺畅，包括按部就班、自订进度的教程和视频，可帮助您开始让 Hadoop 为您所用。没有时间或数据限制，您可以自行安排时间，在大量数据上试验。请 观看视频、学习教程 (PDF) 和 下载 BigInsights Quick Start Edition。

下面的棒球数据示例展示了如何在 Hive 中使用来自 Sean Lahman 网站的棒球数据设计和构建一个数据仓库。我很喜欢挑战数据仓库的反规范化 (denormalizing) 并从该数据构建一个数据仓库。在 “使用 Hive 为数据构建一个库” 中，我使用 VMware Fusion 在我的 Macbook 上创建了一个 IBM InfoSphere® BigInsights™ 虚拟机 (VM)。这是一个简单测试，所以我的 VM 只有 1 GB RAM 和 20 GB 固态硬盘存储空间。操作系统是 Linux® 的 CentOS 6.4 64 位发行版。

要开始使用此示例，请下载 IBM InfoSphere BigInsights Basic Edition（参见 参考资料）。您需要有一个 IBM Universal ID 或注册获取一个 ID，然后才能下载 InfoSphere BigInsights Basic Edition。

导入数据

首先下载包含棒球和棒球运动员的统计数据的 CSV 文件（参见 下载）。在 Linux 内创建一个目录，然后运行：

```
$ Sudo mkdir /user/baseball.

sudo wget http://seanlahman.com/files/database/lahman2012-csv.zip
```

示例包含 4 个主要表，每个表有一个惟一列（Master 表、Batting、Pitching 和 Fielding）以及多个次要表。

设计数据仓库

此数据对一个数据库而言是结构化数据，但对于数据仓库，您需要找出事实和维度。数据仓库设计很简单：您对该数据库进行反规范化，基于运动员统计数据创建一个事实表。然后基于与这些统计数据相关的某些主题区域来创建维度。在连接方面，Hive 表现不是很好，而 MapReduce 也好不了多少，所以拥有一个反规范化的星形模式对某些查询会有所帮助。

设计包含一个名为 fact_Player_Stats 的事实表，它包含各种 CSV 文件和表中包含的每个统计列。您需要使用来自核心表（Batting、Pitching 和 Fielding）的数据，以及来自一些补充表（它们也包含统计数据）的数据。因此，必须添加来自以下表的统计列：

- AllStarFull
- hall of Fame
- BattingPost
- PitchingPost
- FieldingOF
- Salaries
- AwardsPlayers
- AwardsSharePlayers
- Appearances
- SchoolsPlayers

一些表仅包含少数统计列。例如，在 FieldingOF 表中，您只需要将列 stint、Glf、Gcf 和 Grf 添加到 fact_Player_Stats 事实表。对于 SchoolsPlayers 表，只需获取 yearMin 和 yearMax 列。对其他表采取类似的步骤。事实表中仅需要统计列。

备注：您不会使用来自 Managers、Teams、TeamsHalf、SeriesPost 等表的任何数据。

fact_Player_Stats 事实表仅包含键 playerID、FranchID、yearID 和 SchoolID。对于维度表，您必须去掉统计数据（如果存在），仅保留与主题相关的列：

- dim_Players 维度表从 Master 表中获取数据（运动员姓名、生日、传记信息）。主键为 playerID。
- dim_TeamFranchise 维度表从 TeamFranchise 表中获取所有数据。主键为 FranchID。
- dim_Schools 维度表从 Schools 表中获取所有数据。
- dim_Year 是一个基于月份和年份的时间维度表 (1871-2012)。

将数据库用于数据仓库

如果尚未创建棒球数据库，推荐您立即这么做，然后根据这些基础表来构建数据仓库。可通过编写复杂的脚本，从一个平面文件构建数据仓库，然后将同一个平面文件重用于另一个表，但对于本文，我选择使用之前在 “使用 Hive 为数据构建一个库” 中创建的数据库。

使用 Hive 构建数据仓库

完成数据分析和设计之后，是时候基于您的星形模式设计来构建数据仓库了。在 Hive shell 中，创建 baseball_stats 数据库，创建表，加载表，验证这些表是否正确。（此过程已在 “使用 Hive 为数据构建一个库” 中提供。）接下来，创建数据仓库事实表。清单 1 给出了相关代码。

清单 1. 创建数据仓库事实表

```
$ Hive

Create Database baseball_stats;

Create table baseball_stats.fact_player_stats as
        ( SELECT a.playerID, FranchID, yearID, SchoolID, stint int, g int,
g_batting int, ab int, r int, h int, 2b int, 3b int, hr int, rbi int, sb int,
cs int, bb int, so int, ibb int, hbp int, sh int, sf int, gidp int, w int,
l int, g int, gs int, cg int, sho int, sv int, ipouts int, ph int, er int,
phr int, pbb int, pso int, baopp int, era int,pibb int, wp int, phbp int,
bk int, bfp int,gf int, pr int, p sh int, psf int, p gidp int, fg int,
fgs int, innouts int, po int, a int, e int, dp int, pb int, wp int, fsb int,
fcs int, zr int, gamenum int, allstargp int, ballots int, needed int,votes int,
playoff_g int,playoff_ab int, playoff_r int, playoff_h int,
```

```
playoff_2b int, playoff_3b int, playoff_hr int, playoff_rbi int, playoff_sb int,
playoff_cs int, playoff_bb int, playoff_so int, playoff_ibb int, playoff_hbp int,
playoff_sh int, playoff_sh, playoff_sf int, playoff_gidp int, pitchplayoff_w int,
pitchplayoff_l int, pitchplayoff_g int, pitchplayoff_gs int, pitchplayoff_ cg int,
pitchplayoff_sho int, pitchplayoff_sv int, pitchplayoff_ipouts int,
pitchplayoff_h int,pitchplayoff_er int, pitchplayoff_hr int, pitchplayoff_bb int,
pitchplayoff_so int, pitchplayoff_baopp int, pitchplayoff_era int, pitchplayoff_ibb int,
pitchplayoff_wp int,pitchplayoff_hbp int, pitchplayoff_bk int, pitchplayoff_BFP int,
pitchplayoff_gf int, pitchplayoff_r int, pitchplayoff_sh int, pitchplayoff_sf int,
pitchplayoff_gidp int, glf int, grf int, gcf int, salary double, award int,
fieldplayoffs_g int, fieldplayoffs_gs int, fieldplayoffs_innouts int,
fieldplayoffs_po int, fieldplayoffs_a int, fieldplayoffs_e int, fieldplayoffs_dp int,
fieldplayoffs_dp int,fieldplayoffs_tp int, fieldplayoffs_pb int, fieldplayoffs_sb int,
fieldplayoffs_cs int,  appearances_g_all int, appearances_gs int,
appearances_g_batting int, appearances_defense int,
appearances_g_p int, appearances_g_c int, appearances_g_1b int,
appearances_g_2b int, appearances_g_3b int, appearances_g_ss int, appearances_g_ss int,
appearances_g_lf int, appearances_g_cf int, appearances_g_rf int, appearances_dh int,
appearances_ph int, appearances_pr int, yearMin double, yearMax double
from baseball.Batting B JOIN Pitching P ON B.playerid = P.playerID
JOIN fielding F ON B.playerID = F.playerID
JOIN Team T ON b.teamid = t.teamid JOIN TeamFranchises TF ON
t.franchid = tf.franchid ...);
```

现在，创建数据仓库维度表。清单 2 给出了相关代码。

清单 2. 创建数据仓库维度表

```
$ Hive

Create table baseball_stats.dim_Players AS
        ( SELECT lahmanID int, playerID int, managerID int, hofID int, birthyear INT,
          birthMonth INT, birthDay INT, birthCountry STRING, birthState STRING,
          birthCity STRING, deathYear INT, deathMonth INT,deathDay INT,
          deathCountry STRING, deathState STRING, deathCity STRING,
          nameFirst STRING, nameLast STRING, nameNote STRING, nameGive STRING,
          nameNick STRING, weight decimal, height decimal, bats STRING,
          throws STRING, debut  INT, finalGame INT,
          college STRING, lahman40ID INT, lahman45ID INT, retroID INT,
          holtzID INT, hbrefID INT
          FROM baseball.master .... );
```

运行一个查询

我们运行一些查询来确保数据看起来很正常。首先，选择事实表中的所有数据（将其限制到前 10 行）。可运行其他两个查询来确保维度表看起来正常，确保它们已连接到事实表，等等。也可以在事实表上运行一次统计，确保总行数一致。当然，您需要将该数据关联到原始基础表并进行累加。清单 3 显示了测试数据是否存在和正确，以及维度是否连接到事实表的代码。

清单 3. 通过测试来了解俄数据是否存在和正确，以及维度是否连接到事实表

```
$ HIVE
        Use baseball_stats;
        Select * from fact_player_stats limit 10;
```

```
Select A.PlayerID, A.name, B.teamID, B.AB, B.R, B.H, B.2B, B.3B, B.HR, B.RBI
FROM dim_players A JOIN fact_player_stats B ON a.playerid = b.playerid;

Select count(*) from fact_player_stats;

Select count(*) from dim_players.

Select max(r) from fact_player_stats where playerid=1234;
```

如果希望对 Hive 数据仓库中的数据进行更彻底的验证，可以对某些列或所有列计算最小值、最大值或平均值，并将结果与原始基础表进行比较。然后寻找准确的匹配值。

回页首

结束语

显然，创建一个简单的星形模式需要做许多设计工作。举例而言，您可以回过头来以团队的形式创建事实表。这种数据仓库模式的优势在于，您无需连接许多表。而且对于此示例，很少进行每年统计数据以外的更新，在这种情况下，改写数据仓库表或者添加另一年数据并重新计算应该不是问题。

Hive 无疑有自己的局限性，但如果想尽力节省预算，或者某些工具受到高层管制，那么借助一些工作，Hive 可为您提供所需的数据仓库。

回页首

下载

描述	名字	大小	下载方法
样例 CSV 文件	lahman2012-csv.zip	11MB	HTTP

关于下载方法的信息

参考资料

学习

- 查阅 The Data Warehouse Institute (TDWI)，这是一个宝贵的 BI 和数据仓库资源。
- 从 TDWI 了解 大数据分析 以及对大型的多样性数据集应用高级分析技术的更多信息。
- 访问 Hadoop.org，了解 Hadoop 的方方面面的信息。
- 查阅 Hive 和 Sqoop 项目站点，了解有关的更多信息。
- 在 IBM SmartCloud Enterprise 上使用这个来自 Big Data University 的 免费课程（需要登录）创建您自己的 Hadoop 集群。
- 查找帮助您开始使用 InfoSphere BigInsights 的资源，这是 IBM 基于 Hadoop 的产品，通过 Big SQL、文本分析和 BigSheets 等特性提高了开源 Hadoop 的价值。
- 在 developerWorks 大数据内容专区 了解大数据的更多信息。查找技术文档、指引文章、教育、下载、产品信息等。

- 在 developerWorks Information Management 专区，了解关于信息管理的更多信息，获取技术文档、how-to 文章、培训、下载、产品信息以及其他资源。
- 随时关注 developerWorks 技术活动和网络广播。

获得产品和技术

- 下载 InfoSphere BigInsights Quick Start Edition，可以原生软件安装或 VMware 映像形式使用。
- 使用 IBM 试用软件 构建您的下一个开发项目，可直接从 developerWorks 下载获得。

讨论

- 加入 developerWorks 中文社区，developerWorks 社区是一个面向全球 IT 专业人员，可以提供博客、书签、wiki、群组、联系、共享和协作等社区功能的专业社交网络社区。
- 加入 IBM 软件下载与技术交流群组，参与在线交流。

使用 Hive 构建数据库

当您需要处理大量数据时，存储它们是一个不错的选择。令人难以置信的发现或未来预测不会来自未使用的数据。大数据是一个复杂的怪兽。用 Java™ 编程语言编写复杂的 MapReduce 程序要耗费很多时间、良好的资源和专业知识，这正是大部分企业所不具备的。这也是在 Hadoop 上使用诸如 Hive 之类的工具构建数据库会成为一个功能强大的解决方案的原因。

Peter J. Jamack, 大数据分析顾问, Peter J Jamack

2013 年 9 月 06 日

如果一家公司没有资源构建一个复杂的大数据分析平台，该怎么办？当业务智能 (BI)、数据仓库和分析工具无法连接到 Apache Hadoop 系统，或者它们比需求更复杂时，又该怎样办？大多数企业都有一些拥有关系数据库管理系统 (RDBMSes) 和结构化查询语言 (SQL) 经验的员工。Apache Hive 允许这些数据库开发人员或者数据分析人员使用 Hadoop，无需了解 Java 编程语言或者 MapReduce。现在，您可以设计星型模型的数据仓库，或者常态化的数据库，而不需要挑战 MapReduce 代码。忽然之间，BI 和分析工具，比如 IBM Cognos® 或者 SPSS® Statistics，就可以连接到 Hadoop 系统。

数据库

构建数据库，并且能够使用这些数据，这不是 Hadoop 或者数据库问题。多年以来，人们一直习惯将数据组织到库中。有许多由来已久的问题：如何将数据分门别类？如何将所有数据连接到集成的平台、机箱或者库？多年来，各种方案层出不穷。

人们发明了很多方法，比如 Dewey Decimal 系统。他们将通讯录中的人名或企业名按照字母顺序排列。还有金属文件柜、带货架的仓库、地址卡文件系统，等等。雇主尝试用时间卡，打卡器以及时间表追踪员工。人们需要结构化和组织化数据，还需要反映和检查这些数据。如果您无法访问、结构化或理解这些数据，那么存储这么多的数据有什么实际意义呢？

RDBMSes 使用了过集合论和第三范式。数据仓库有 Kimball、Inmon、星型模型、Corporate Information Factory，以及专用数据集市。他们有主数据管理、企业资源规划、客户关系管理、电子医疗记录和其他许多系统，人们使用这些系统将事务组织到某种结构和主题中。现在，我们有大量来自各个行业的非机构化或半结构化数据，例如，社交媒体、邮件、通话记录、机械指令、远程信息，等等。这些新数据需要集成到存储结构化的新旧数据的非常复杂、非常庞大的系统中。如何分类才能使得销售经理能够改进报告？如何构建库才能使得执行主管能够访问图表和图形？

您需要找到一种将数据结构化到数据库的方法。否则，只是拥有大量只有数据科学家才能访问数据。有时，人们只是需要简单的报告。有时，他们只是想要拖拽或者编写 SQL 查询。

回页首

大数据、Hadoop 和 InfoSphere BigInsights

本小节将向您介绍 InfoSphere® BigInsights™，以及它与 Hadoop、大数据、Hive、数据库等有何联系。InfoSphere BigInsights 是 Hadoop 的 IBM 分区。您可能对 Apache 和 Cloudera 比较了解，但是业内许多人都曾涉足 Hadoop。它开始于开源的使用 MapReduce 的 Hadoop 和 Hadoop 分布式文件系统 (HDFS)，通常还包括其他工具，比如 ZooKeeper、Oozie、Sqoop、Hive、Pig 和 HBase。这些发布版与普通 Hadoop 的区别在于它们被添加在 Hadoop 顶层。InfoSphere BigInsights 就属于这一类版本。

您可以在 Hadoop 的 Cloudera 版本之上使用 InfoSphere BigInsights。此外，InfoSphere BigInsights 提供一个快速的非结构化的分析引擎，您可以将它和 InfoSphere Streams 结合在一起使用。InfoSphere Streams 是一个实时的分析引擎，它开创了联合实时分析和面向批次的分析的可能。

InfoSphere BigInsights 还拥有内置的、基于浏览器的电子表格 BigSheets。这个电子表格允许分析人员每天以电子表格样式使用大数据和 Hadoop。其他功能包括基于角色的安全和管理的 LDAP 集成；与 InfoSphere DataStage® 的集成，用于提取、转换、加载 (ETL)；常用的使用案例的加速器，比如日志和机器数据分析；包含常用目录和可重复使用工作的应用目录；Eclipse 插件；以及 BigIndex，它实际上是一个基于 Lucene 的索引工具，构建于 Hadoop 之上。

您还可以使用 Adaptive MapReduce、压缩文本文件、自适应调度增强来提高性能。此外，您还可以集成其他应用，例如，内容分析和 Cognos Consumer Insights。

回页首

Hive

Hive 是一个强大的工具。它使用了 HDFS，元数据存储（默认情况下是一个 Apache Derby 数据库）、shell 命令、驱动器、编译器和执行引擎。它还支持 Java 数据库连接性 (JDBC) 连接。 由于其类似 SQL 的能力和类似数据库的功能，Hive 能够为非编程人员打开大数据 Hadoop 生态系统。它还提供了外部 BI 软件，例如，通过 JDBC 驱动器和 Web 客户端和 Cognos 连接。

您可以依靠现有的数据库开发人员，不用费时费力地寻找 Java MapReduce 编程人员。这样做的好处在于：您可以让一个数据库开发人员编写 10-15 行 SQL 代码，然后将它优化和翻译为 MapReduce 代码，而不是强迫一个非编程人员或者编程人员写 200 行代码，甚至更多的复杂 MapReduce 代码。

Hive 常被描述为构建于 Hadoop 之上的数据仓库基础架构。事实是，Hive 与数据仓库没有什么关系。如果您想构建一个真实的数据仓库，可以借助一些工具，比如 IBM Netezza。但是如果您想使用 Hadoop 构建一个数据库，但又没有掌握 Java 或者 MapReduce 方面的知识，那么 Hive 会是一个非常不错的选择（如果您了解 SQL）。Hive 允许您使用 Hadoop 和 HBase 的 HiveQL 编写类似 SQL 的查询，还允许您在 HDFS 之上构建星型模型。

Hive 与 RDBMSes

Hive 是一个读模式 系统，而 RDBMSes 是一个典型的写模式 系统。传统的 RDMBSes 在编写数据时验证模型。如果数据与结构不符，则会遭到拒绝。Hive 并不关心数据的结构，至少不会在第一时间关心数据结构，它不会在您加载数据时验证模型。更确切地说，只在您运行查询之后，它才会关心该模型。

Hive 的限制

在使用 Hive 时可能会有一些挑战。首先，它与 SQL-92 不兼容。某些标准的 SQL 函数，例如 `NOT IN`、`NOT LIKE` 和 `NOT EQUAL` 并不存在，或者需要某种工作区。类似地，部分数学函数有严格限制，或者不存在。时间戳或者 `date` 是最近添加的值，与 SQL 日期兼容性相比，更具有 Java 日期兼容性。一些简单功能，例如数据差别，不能正常工作。

此外，Hive 不是为了获得低延时的、实时或者近乎实时的查询而开发的。SQL 查询被转化成 MapReduce，这意味着与传统 RDBMS 相比，对于某种查询，性能可能较低。

另一个限制是，元数据存储默认情况下是一个 Derby 数据库，并不是为企业或者生产而准备。部分 Hadoop 用户转而使用外部数据库作为元数据存储，但是这些外部元数据存储也有其自身的难题和配置问题。这也意味着需要有人在 Hadoop 外部维护和管理 RDBMS 系统。

回页首

安装 InfoSphere BigInsights

这个棒球运动数据示例向您展示了在 Hive 中如何从平面文件构建常用的数据库。虽然这个示例比较小，但它显示了使用 Hive 构建数据库有多么轻松，您可以使用该数据运行统计数据，确保它符合预期。将来尝试组织非结构数据时就无需检查那些信息。

完成数据库构建之后，只要连接到 Hive JDBC，就可以使用任何语言构建 Web 或者 GUI 前端。（配置和设置一个 thrift 服务器，Hive JDBC 是另一个话题）。我使用 VMware Fusion 在我的 Apple Macbook 上创建了一个 InfoSphere BigInsights 虚拟机 (VM)。这是一个简单的测试，这样我的 VM 就有 1 GB 的 RAM 和 20 GB 的固态磁盘存储空间。操作系统是 CentOS 6.4 64-bit distro 的 Linux®。您还可以使用某些工具，例如 Oracle VM VirtualBox，如果您是 Windows® 用户，那么您还可以使用 VMware Player 创建 InfoSphere BigInsights VM。（在 Fusion 上设置 VM、VMware Player 或者 VirtualBox 不在本文的讨论范围之内。）

从下载 IBM InfoSphere BigInsights 基础版开始（参阅 参考资料）。您需要有一个 IBM ID，或者您可以注册一个 ID，然后下载 InfoSphere BigInsights 基础版。

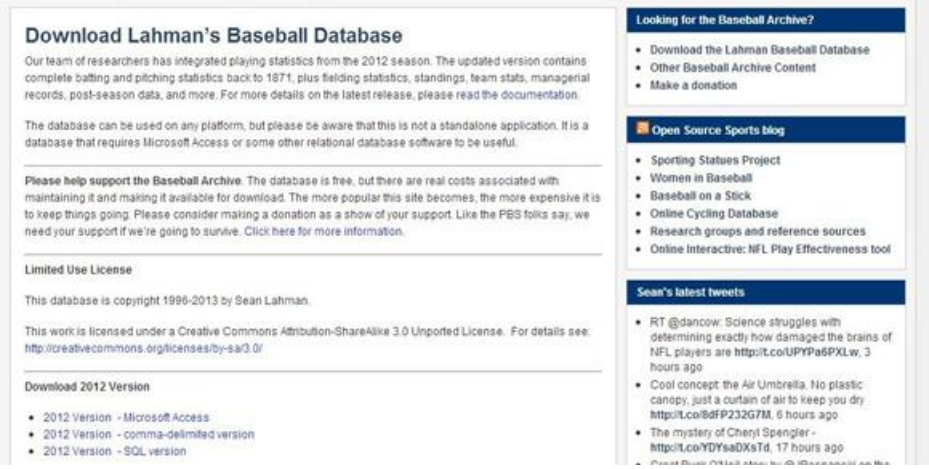
回页首

输入和分析数据

现在，您可以在任何地方获取数据。绝大多数网站都提供了逗号分隔值 (CSV) 格式的数据：天气、能源、运动、金融和博客数据。例如，我使用来自 Sean Lahman 网站的结构化数据。使用非结构化数据会费力一些。

首先 下载 CSV 文件（参见 图 1）。

图 1. 下载示例数据库



如果您宁愿在一个更手动的环境中，那么可以从 Linux® 完成它，您需要创建一个目录，然后运行 `wget`：

```
$ Sudo mkdir /user/baseball.
```

```
sudo wget http://seanlahman.com/files/database/lahman2012-csv.zip
```

该数据使用了 Creative Commons Attribution-ShareAlike 3.0 Unported 许可。

压缩文件在 CSV 文件中，包含了棒球和棒球运动员的统计数据。示例中包含四个主表，每个表都只有一个列（Player_ID）：

- **Master table.csv—** 运动员姓名、出生日期和生平信息
- **Batting.csv—** 击球统计
- **Pitching.csv—** 投球统计
- **Fielding.csv—** 接球统计

辅表：

- **AllStarFull.csv—** 全明星阵容
- **Hall of Fame.csv—** 名人堂投票数据
- **Managers.csv—** 管理统计
- **Teams.csv—** 年度统计和排名
- **BattingPost.csv—** 赛季后的击球统计
- **PitchingPost.csv—** 赛季后的投球统计
- **TeamFranchises.csv—** 加盟信息
- **FieldingOF.csv—** 场外位置数据
- **FieldingPost.csv—** 赛季后的现场数据
- **ManagersHalf.csv—** 经纪人的分季数据
- **TeamsHalf.csv—** 团队的分季数据
- **Salaries.csv—** 球员薪资数据
- **SeriesPost.csv—** 赛季后系列信息
- **AwardsManagers.csv—** 经纪人奖项
- **AwardsPlayers.csv—** 球员奖项
- **AwardsShareManagers.csv—** 经纪人奖项投票
- **AwardsSharePlayers.csv—** 球员奖项投票

- **Appearances.csv**
- **Schools.csv**
- **SchoolsPlayers.csv**

回页首

设计数据库

设计数据库的大部分内容已经完成。Player_ID 是四个主表（Master、Batting、Pitching 和 Fielding）的主键。（为了更好地理解表格结构和依赖性，请阅读 Readme2012.txt。）

设计非常简单：主表是通过 Player_ID 连接的。Hive 并没有真的使用主键或者引用完整性的概念。*Schema on Read* 意味着 Hive 会摒弃您输入到表格中的所有内容。如果文件是混乱无序的，那么可能需要寻求连接它们的最佳方法 。此外，在将数据加载到 HDFS 或 Hive 之前，需要进行一些转化。根据 Schema on Rea 原理，不良数据在 Hive 中将彻底变成不良数据。这就是数据分析（无论是源级别的或者 HDFS 级别的）是一个重要步骤的原因。没有数据分析，最终获得的原始数据没有人可以使用。幸运的是，这个棒球的示例包含一些数据，这些数据在您输入 Hadoop 之前，已经被清理和组织到一起。

回页首

将数据加载到 HDFS 或者 Hive

将数据加载到 Hadoop 使用了很多不同的理论和实践。有时，您可以将原始文件直接输入到 HDFS。您可能会创建一个目录和子目录来组织文件，但是将文件从一个地方复制或移动到另一个位置是一个简单的过程。

就这个示例来说，只需发出 put 命令，然后创建一个名为 **baseball** 的目录即可：

```
Hdfs dfs  -mkdir  /user/hadoop/baseball

hdfs dfs  -put  /LOCALFILE  /user/hadoop/baseball
```

回页首

使用 Hive 构建数据库

随着数据分析和设计的完成，下一步就是构建数据库了。

虽然我没有介绍所有的示例，但是，如果跟随我构建了第一个示例，那么您就能够了解如何完成剩下的步骤。我通常会构建一些 SQL 文本脚本，然后将它们输入或者粘贴到 Hive。其他人可以使用 Hue 或其他工具来构建数据库和表格。

为了简便起见，我们使用了 Hive Shell。高级步骤是：

1. 创建棒球数据库
2. 创建表格
3. 加载表格
4. 验证表格是正确的

您会看到一些选项，例如，创建外部或者内部数据库和表格，但是在这个示例中，需要遵守内部默认设置。实际上，内部的 就意味着 Hive 处理了内部存储的数据库。清单 1 说明了 Hive shell 的流程。

清单 1. 创建数据库

```
$ Hive

        Create Database baseball;
```

```
Create table baseball.Master
 ( lahmanID int, playerID int, managerID int, hofID int, birthyear INT,
   birthMonth INT, birthDay INT, birthCountry STRING, birthState STRING,
   birthCity STRING, deathYear INT, deathMonth INT, deathDay INT,
   deathCountry STRING, deathState STRING, deathCity STRING,
   nameFirst STRING, nameLast STRING, nameNote STRING, nameGive STRING,
   nameNick STRING, weight decimal, height decimal, bats STRING,
   throws STRING, debut  INT, finalGame INT,
   college STRING, lahman40ID INT, lahman45ID INT, retroID INT,
   holtzID INT, hbrefID INT  )
 ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' ;
```

其他所有表也都遵守这个程序。为了将数据加载到 Hive 表，将会再次打开 Hive shell，然后运行以下代码：

```
$hive
    LOAD DATA LOCAL INPATH Master.csv OVERWRITE INTO TABLE baseball.Master;
```

回页首

使用 Hive 构建标准化数据库

这个棒球的数据库或多或少是标准化的：有四个主表和几个辅表。再次重申，Hive 是一个 Schema on Read，因此您必须完成数据分析和 ETL 阶段的大部分工作，因为没有传统 RDBMSes 中的索引或者引用完整性。如果您想要使用索引功能，那么下一步应该使用类似 HBase 的工具。请查看 清单 2 中的代码。

清单 2. 运行一个查询

```
$ HIVE
        Use baseball;
        Select * from Master;
        Select PlayerID from Master;
        Select A.PlayerID, B.teamID, B.AB, B.R, B.H, B.2B, B.3B, B.HR, B.RBI
        FROM Master A JOIN BATTING B ON A.playerID = B.playerID;
```

回页首

结束语

这就是 Hive 的优势以及构建数据库的好处：它为混沌的世界创建了结构。和我们喜欢讨论的非结构化或半结构化数据一样，它最终还是要了解谁可以分析数据，谁能基于它运行报告，以及您如何能够让它快速投入到工作中。大多数用户将 Hive 视为某种黑盒：他们不在意数据来自何处，也不在乎需要做什么才能以正确格式获取数据。也不会在意集成或者验证这些数据有多么困难，只要这些数据是精确的。这通常意味着您必须有组织和结构。否则，您的数据库会成为一个永久存储无限制数据的死区，没人能够或者想要使用这些数据。

结构复杂的数据仓库已经风光不再。虽然近年情况有所好转，但是概念还是一样：这是一个业务，业务用户想要结果，而不是编程逻辑。这就是在 Hive 中构建数据库会成为正确开端的原因。

回页首

下载

描述	名字	大小
样例 CSV 文件	lahman2012-csv.zip	11MB

参考资料

学习

- 请阅读 Mckinsey & Company 撰写的 *Big data: The next frontier for innovation, competition, and productivity*。
- 了解关于 The Data Warehouse Institute（TDWI）的更多信息，这是一个有关 BI 和数据仓库的宝贵资源。
- 了解关于 大数据分析 的更多信息，大型多样数据集的高级分析技术应用，来自 TDWI。
- 在 Big Data University（需要注册）中使用免费的 Hadoop Fundamentals 教程了解 Hadoop 的基础。了解 Hadoop 架构、HDFS、MapReduce、Pig、Hive、JAQL、Flume 和其他许多与 Hadoop 相关的技术。使用以下方法在 Hadoop 集群上进行实机操作：在云上，使用提供的 VMWare 图像或者本地安装。
- 在 Big Data University 上探索免费资源，了解以下主题：从 Hadoop 原理和文本分析本质，到 Hadoop 和实现流计算的 SQL 访问。
- 使用 Big Data University（需要注册）的免费教程，在 IBM SmartCloud Enterprise 上创建您自己的 Hadoop 集群。
- 了解关于 IBM InfoSphere Streams 的更多信息。
- 在 developerWorks 大数据内容专区 了解有关大数据的更多信息。查找技术文档、how-to 文章、教育、下载、产品信息，等等。
- 在 developerWorks Information Management 专区，了解关于信息管理的更多信息，获取技术文档、how-to 文章、培训、下载、产品信息以及其他资源。
- 随时关注 developerWorks 技术活动和网络广播。

获得产品和技术

- 访问 Hadoop.org，获取有关 Hadoop 的所有信息。
- 访问 Hive 项目网站，获取关于 Apache Hive 的更多信息。
- 访问 Sqoop 项目网站，获取关于 Apache 的批量数据转换工具的更多信息。
- 访问 HBase 项目网站，获取关于 Apache 的分布式数据商店的更多信息。
- 了解关于 IBM Netezza 的更多信息。
- 了解关于 InfoSphere BigInsights 的更多信息。
- 下载 InfoSphere BigInsights Basic Edition for Linux 2.0.0.0。
- 了解关于 SPSS Statistics 的更多信息。
- 了解关于 Cognos 的更多信息。
- 获取 IBM InfoSphere Streams 的试用版。
- 获取 IBM InfoSphere BigInsights 的试用版，管理和分析大量静止的结构化和非结构化数据。
- 使用 IBM 试用软件 构建您的下一个开发项目，可直接从 developerWorks 下载。

讨论

- 加入 developerWorks 中文社区，developerWorks 社区是一个面向全球 IT 专业人员，可以提供博客、书签、wiki、群组、联系、共享和协作等社区功能的专业社交网络社区。
- 加入 IBM 软件下载与技术交流群组，参与在线交流。