

# 云平台下的装备保障管理系统结构化数据查询<sup>①</sup>

张惠民, 胡海荣, 向阳霞

(装甲兵工程学院 信息工程系, 北京 100072)

**摘要:** 首先针对我军装备保障管理系统关系数据库在海量数据查询以及分析处理方面的不足, 在对系统数据资源分析的基础上提出并实现了基于 Hadoop 云平台结构化数据查询策略, 通过 Sqoop 工具将数据库中的数据导入到 HDFS 中, 并利用 Hive 进行数据分析. 然后通过实验证明此方法克服了海量数据在单机环境中查询效率低下的缺点, 具有较高的实用价值.

**关键词:** 数据资源; Hadoop; 云平台; Sqoop; Hive

## Structured Data Query in the Equipment Support Management System Based on Cloud Platform

ZHANG Hui-Min, HU Hai-Rong, XIANG Yang-Xia

(Department of Information Engineering, Academy of Armored Force Engineering, Beijing 100072, China)

**Abstract:** First, this paper talked about the deficiencies of current Equipment Support Management System on massive data query and analysis. Then it proposed and realized a method of Structured Data Query used by the Cloud Platform based on Hadoop, after analyzing the data source. In the article, we used the tool of Sqoop to import data from a relational database into HDFS, and then use Hive to analyze the data. Finally, the experiment proved that the method makes up the shortage of the data query in the environment of single computer, and has high practical value.

**Key words:** structured data; Hadoop; cloud platform; Sqoop; Hive

我军装备保障信息化建设中开发了一系列的装备保障管理系统, 经研究发现这些系统大都建立在关系数据库基础之上. 由于装备种类繁多, 数量巨大, 如果继续采用传统方式对海量数据进行集中式查询, 将超过关系数据库的支撑范围, 或者说超过关系数据库的高效处理范围. 针对上述问题可以考虑利用 Hadoop 搭建的云平台来解决, 但是 Hadoop 中的 HDFS 一般情况下只支持非结构化数据的存取; 而对于关系数据库占主导地位的装备保障管理系统来说, 如果将结构化数据转换成非结构化数据不但浪费大量时间和财力还可能会导致很多重要信息的丢失. 虽然 Hadoop 中也有 Hbase 数据库, 但是它只适合存储非结构化数据. 因此, 如何用 Hadoop 搭建的云平台进行关系数据库的海量数据查询是本文研究的重点. 本文将在分析全军装备保障管理系统数据资源的前提下, 设计并实现基于云平台的结构化数据查询系统架构.

## 1 相关技术介绍

### 1.1 Hadoop

Hadoop<sup>[1]</sup>是并行技术、分布式技术和网格计算技术发展的产物, 是一种为适应大规模数据计算和存储而发展起来的模型架构<sup>[2]</sup>. Hadoop 的两个核心技术为 HDFS 和 MapReduce.

HDFS<sup>[3]</sup>是一个运行在大量廉价硬件之上的分布式文件系统, 它是 Hadoop 平台的底层文件存储系统, 主要负责数据的管理和存储. 一个 HDFS 集群中有一个 NameNode 和多个 DataNode. NameNode 是中心服务器, 主要用来管理元数据和文件块、简化元数据的更新操作; DataNode 通常在集群中一个节点一个, 用来存储、检索数据块.

MapReduce<sup>[4]</sup>是 Hadoop 用来处理云计算中海量数据的编程框架, 简单易用, 程序员在不了解底层实现细节的基础上便可写出程序来处理海量数据.

<sup>①</sup> 收稿时间:2014-05-22;收到修改稿时间:2014-06-26

MapReduce 首先通过 Map 程序将海量数据分割成多个小区块, 将其分配给大量服务器进行处理; 然后将处理结果交给 Reduce, 最后 Reduce 将处理结果汇总后输出到客户端.

### 1.2 Hive

Hive<sup>[5]</sup>是基于 Hadoop 的数据仓库基础架构. 它可以存储、查询、分析 HDFS 中的大规模数据. Hive 最初设计目的是让熟悉 SQL 语言的人员可以方便操作 HDFS 文件系统, 技术人员可以通过 HiveQL 语言与 Hive 交互, HiveQL 是类 SQL 语言, 他们之间的语法非常相近, 它的设计受 MySQL 的影响很大; HiveQL 语言通过对语句解析和转换, 最终生成多个 MapReduce 任务进行处理.

### 1.3 Sqoop

Sqoop<sup>[6]</sup>全称“SQL-to-Hadoop”, 它是 Clouder 公司开发的用来将关系型数据库与 Hadoop 中的数据进行相互转移的工具. 用它可以很方便地将 HDFS 中的数据导入到关系型数据库(如 MySQL, SQLServer, Oracle 等)中; 也可以将关系型数据库中的数据导入到 HDFS 中.

## 2 装备保障管理系统数据资源分析

### 2.1 关系数据库结构设计

在装备保障管理系统中, 为达到业务数据可以纵向贯通、横向共享的目的, 经过分析系统中的业务数据和业务组成, 对各系统中使用的数据进行了分类、规整. 从业务处理角度将系统分为装备、人员、弹药、器材、设施、设备、资料、经费八类管理对象.

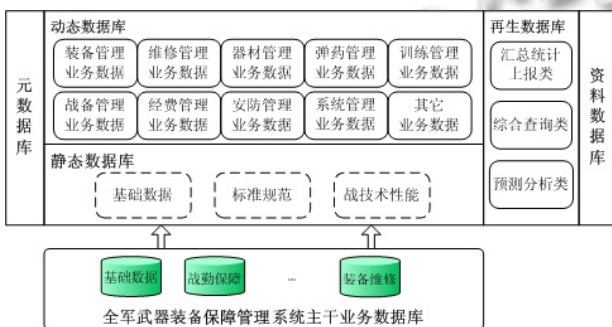


图 1 数据库逻辑关系

系统中有 3 类数据库: 元数据库、资料数据库和业务数据库. 元数据库作用是管理和组织资源信息、快速定位资源位置、创建资源目录、实现资源交换、

提供资源转换信息等. 通过分析通用装备保障业务信息系统中的业务数据, 对各业务系统中的数据进行了规整和分类, 将系统中的业务数据分为: 装备基础数据、装备综合数据、保障业务数据、交换数据和其他数据等. 数据库的逻辑关系如图 1 所示, 业务数据库按照数据性质分为静态、动态和再生数据库; 元数据库以及资料数据库用来提供对数据库访问的支持, 业务数据信息也通过元数据库向外提供信息服务.

### 2.2 基于 RDF 的元数据描述方法

装备保障管理系统信息资源庞大, 利用元数据提取数据信息本质, 抽象出统一的描述模型, 能够在很大程度上提升数据整合的效率; 通过记录用户权限以及数据资源位置来提高数据查询的准确性、高效性、安全性以及可靠性. 通过业务元数据记录数据之间的转换规则和关联关系来提高系统的可维护性和可集成性. 利用元数据不但方便用户和开发人员理解数据, 还能够确保数据信息的一致性、准确性、完整性. 所以, 对于元数据标准格式的设计是很有必要的. 对于系统中的元数据设计, 本文主要从数据描述性元素进行考虑.

RDF 是 W3C 发布的基于 XML 来描述 Web 资源的一种标准框架, 具有抽象性、领域无关性、易扩展、易交换、开放性、简单等特点, 因此本文选择 RDF 作为元数据的描述框架.

RDF 通常由资源、属性值、属性类型三部分组成. 其资源描述方式一般有两种: 三元组表示法, {主题, 谓词, 客体}; 图形表示法, 椭圆表示资源, 直线表示属性, 属性值用矩形表示<sup>[7]</sup>. 如图 2 所示, 本系统基于文献[8]研究设计了 RDF 资源描述的方法过程, 对系统中的任何资源都可以通过该过程生成资源描述文档, 达到异构系统中的数据交互和资源共



图 2 RDF 资源描述方法步骤

建模: 首先, 对原始数据进行建模, 确定要描述的数据资源以及各种属性, 再根据各属性元素以及它们之间的关系生成 RDF 模型, 最后利用该模型生成资源模型文档. 建模的过程与原始数据的组织形式没有关系, 这是解决异构数据共享交互的关键.

描述：用上一过程生成的资源模型文档，加上 RDF 自带的词汇表及在建模过程中自己定义的词汇表来进行资源的统一描述，生成资源描述文档，该文档是基于 XML 的。

应用：通过交换资源描述文档，可以在异构平台之间有效发现、访问其他平台的资源，实现异构数据

源的共享与交互。

利用上述方法，结合 RDF 图形表示法进行数据资源描述，如图 3 所示。用椭圆形表示元数据模型；用直线表示 TZBZ\_DSPort、TZBZ\_DSIP、...、TZBZ\_DSKeyref 等各种属性；用矩形表示 port、ip、...、key 等属性值。

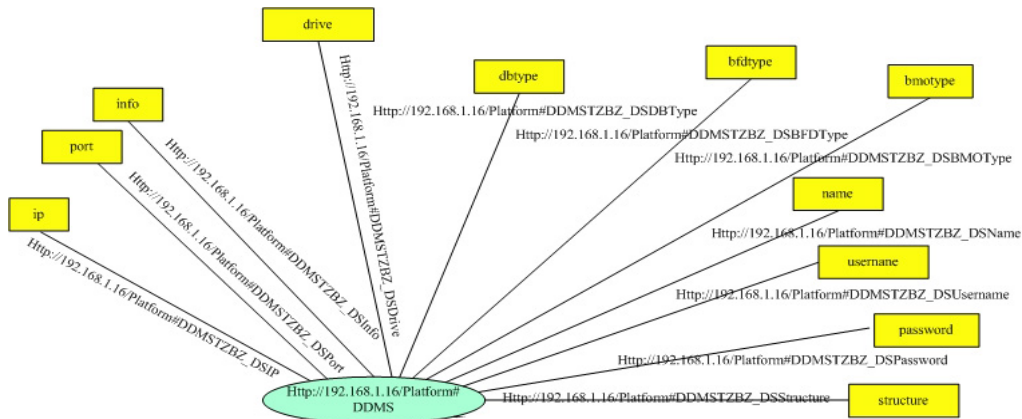


图 3 基于 RDF 的元数据描述模型

### 3 基于云的结构化数据查询系统设计

#### 3.1 系统架构设计

用传统的关系型数据库在海量数据中进行集中查询时效率低下，用户等待时间较长，导致交互性较差。所以，依靠传统单一节点的计算能力已经无法满足需求，利用云计算将消耗大量计算资源的复杂计算通过网络分布到多节点上计算成为了新的有效解决方案<sup>[9]</sup>。本文就是利用云计算的思想将系统中的数据查询部分迁移到 Hadoop 搭建的云平台上，如图 4 所示，系统架

构自上而下分为四层，分别为：视图层、业务逻辑层、Hive 层和 Hadoop 层。

视图层：本层包括装备管理、器材管理、弹药管理、人员管理、权限管理、日志管理、元数据查询等多个部分。用户登录系统后，可以根据权限输入查询条件或者关键字信息来查询自己所需要的数据。

业务逻辑层：本层包括装备管理、器材管理、权限认证、数据处理、查询条件解析等模块。系统接受并解析用户输入的查询条件，然后将不同的请求分给不同的业务逻辑进行处理，再调用底层的接口进行查询，最后把查询结果返回给用户。

Hive 层：该层接收业务逻辑层传递过来的查询请求，并通过自身解释器和语义分析器对用户提交的 HiveQL 语句进行解析和编译，然后通过逻辑计划生成器和查询计划生成器生成基于 Hadoop 的 MapReduce 任务。系统运行之前的数据准备阶段需要借助 Sqoop 将数据从关系型数据库中导入到 Hive 中。

Hadoop 层：也可称为计算层，本层利用 MapReduce 执行 Hive 层的计算任务来读写导入到 HDFS 中的不同数据。该层将查询任务分为 Map 和 Reduce 两个阶段，动态分配给集群中不同节点执行，最后将查询结果存储在 HDFS 文件系统中。

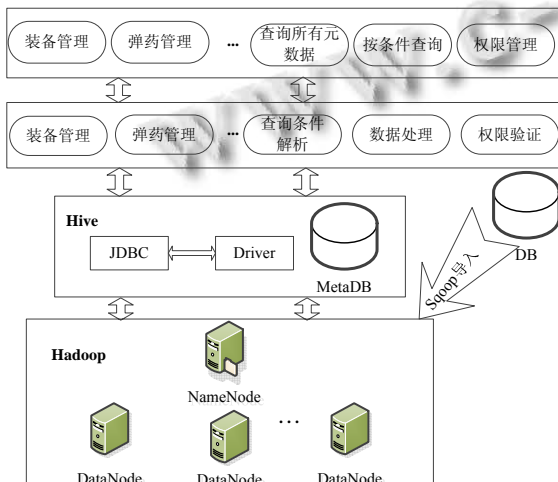


图 4 基于 Hadoop 的数据查询架构图

### 3.2 系统工作流程

首先采用 Sqoop 工具将数据导入到 HDFS 中建立数据仓库模型,在此过程中将生成元数据并全部存储在 Hive 自带的 Derby 元数据库中(若处于多用户环境中,需要将其存储在自己安装的 MySQL 中). 建立完成后的数据仓库按照如下方式运行: 客户端发起数据查询请求, 根据请求内容查询元数据库中相对应的表定义, 如果满足请求则进入文件目录查询相对应的表; 通过 HiveQL 查找要查询目录的属性, 获取到属性值后进行数据的查询、分析、生成报表等操作. 然后将数据查询计划存储在 HDFS 中, 并将结果返回并显示给客户端. 其工作过程如图 5 所示.

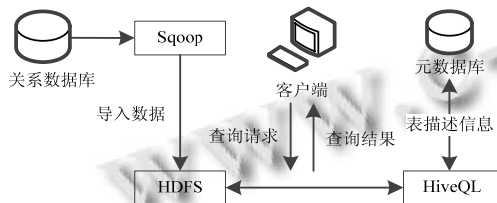


图 5 查询系统工作过程

## 4 性能测试

### 4.1 环境搭建与数据准备

为验证提出的方案是否具有高效性, 本文搭建了由 6 台服务器组成的 Hadoop 集群, 其硬件配置为处理器: Intel Core i5, CPU: 2.4GHz, 内存: 2G, 硬盘: 500G; 软件配置为 OS: Red Hat Linux OS5.0, Hadoop: hadoop-1.0.3, Hive: hive-0.9.0, Sqoop: sqoop-1.4.4. 其中 1 台 NameNode 节点、5 台 DataNode 节点, Hive 和 Sqoop 运行在 NameNode 节点上. 由于装备保障管理系统中数据库表的数量众多、关联性复杂, 而且装备数据由于其保密性而不易获取, 因此本文的测试数据通过编写程序获得, 将其存放在关系数据库中, 方便起见选择 MySQL 作为测试数据库来存放数据. 数据获取示例程序如下所示.

```

InsertIntoData insert = null;
int i=0;
while(i<500000){
    int a = (int) (Math.random()*10000);
    int b = (int) (Math.random()*10000);
    int c = (int) (Math.random()*10000);
    int d = (int) (Math.random()*10000);
    int e = (int) (Math.random()*10000);
    String sql = "insert into project" +
        "(id,name,description,type,type,status)" +
        " values(?,?,?,?,?,?)";
    String[] paras = {"id"+a,"name"+b,
        "description"+c,"type"+d,"status"+e};
    insert = new InsertIntoData();
    insert.updateData(sql, paras);
    i++;
}
    
```

若用云平台处理关系数据库中的数据, 必须将数据导入到 HDFS 中. 这里我们使用 Sqoop 工具, 文章前面已经介绍过此工具, 它具有在 Hive 和关系型数据库之间迁移数据的功能, 可以用它把数据从关系型数据库导入到 HDFS 中, 也可以将 HDFS 中的数据导入到关系型数据库中, 该实验用的是 Sqoop 的第一个功能. 实验数据大小、数据条目如表 1 所示.

表 1 数据大小(MB)和元组数(条)

数据大小	61	123	247	614	985	1220
元组数	$5 \times 10^5$	$10^6$	$2 \times 10^6$	$5 \times 10^6$	$8 \times 10^6$	$10^7$

### 4.2 查询性能对比

本实验主要是为了得出不同数据量在单机环境以及集群环境不同节点中的查询时间, 实验内容如下:

#### ① 聚集查询

查询表 project 中的总元组数, sql 语句为: select count(\*) from project. 每组实验进行三次, 记录执行时间并计算平均值. 聚集查询测试结果如图 6 所示.

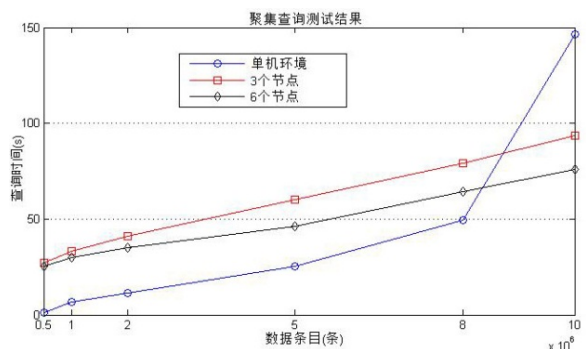


图 6 聚集查询测试结果对比图

#### ② 连接查询

查询两张表中的某些字段相同的数据并按照 id 降序排列, sql 语句为: select p.id, p.create\_at, t.update\_at, t.status from project p join test t on (p.name=t.name and p.description=t.description and p.status=t.status) order by p.id desc. 每组实验进行三次, 记录执行时间并计算平均值. 连接查询测试结果如图 7 所示.

从图中可以看出当数据量较小时, 关系数据库的查询速度要高于 Hadoop 集群, 而当数据量达到一定程度时, 例如聚集查询达到一千万行、连接查询达到五百万行时, 集群环境下的查询优势开始显现出来. 而且随着数据量的不断加大, 这种优势更加明显. 正如上文提到的, Hadoop 是处理大数据的, 对于小数据量



并没有优势,因为它的任务处理机制Map-Reduce本身就非常耗时;再加上Hive在解析Sql语句以及调度Map-Reduce任务上都需要耗费一些时间。

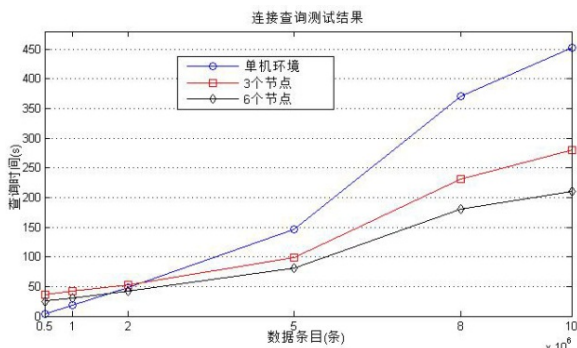


图7 连接查询测试结果对比图

## 5 结论

目前关系数据库对于少量数据的操作已经很成熟,它一般通过建立索引来提高查询效率。但是海量数据的查询对于关系数据库来说已经成为一个瓶颈,它不像云存储平台那样可以无限扩充;而且不可能为每个属性都建立索引,因为这会额外占用很多空间。因此当查询大规模数据时,云存储的优势就显而易见了,由于条件所限,在实验数据还不是很大的情况下,其查询的效率已经赶上甚至超过关系数据库了。由此可见,利用云计算的思想解决目前我军装备管理系统关系数据库存在的海量数据查询分析能力不足的问题,

具有广阔的应用前景。下一步的工作将继续增加集群中节点的数量,并探究节点数与查询效率之间的关系,寻找最优配置,达到节约资源的目的。

## 参考文献

- 1 Shvachko K, Kuang H, Radia S, Chansler R. The hadoop distributed file system. Sunnyvale. California USA. IEEE, 2010: 1-10.
- 2 高蓟超.Hadoop 平台存储策略的研究与优化.北京:北京交通大学,2012.
- 3 黄晓云.基于 HDFS 的云存储服务系统研究[学位论文].大连:大连海事大学,2010.
- 4 杨文志.云计算技术指南:应用、平台与架构.北京:化学工业出版社,2010.
- 5 Ashish T, Joydeep S, Namit J, et al. Hive-A petabyte scale data warehouse using hadoop. IEEE 26th International Conference on Data Engineering(ICDE). 2010. 996-1005.
- 6 Apache Sqoop. Sqoop. <http://sqoop.apache.org>. [2012-03]
- 7 程相等.基于 RDF 的军事网格资源描述方法研究.计算机技术与发展,2011,21(10):129-132.
- 8 苏天醒等.基于 RDF 的资源描述与共享方法的研究.应用与安全,2010,(9):36-38.
- 9 刘永增,张晓景,李先毅.基于 Hadoop/Hive 的 web 日志分析系统的设计.广西大学学报(自然科学版),2011,36(1):314-317.