基于Lucene的海量数据库全文检索的设计与实现

徐叶强，朱艳辉，栗春亮，王文华

（湖南工业大学 计算机与通信学院，湖南 株洲 412008）

摘　要：基于Lucene实现了一个海量数据库全文检索的原型。把关系数据库引入了本系统，可针对不同类型的源数据库灵活配置，比采用配置文件更加灵活；采用多线程，通过动态机制来实现不同类型源数据库中记录的抽取、转换、建立索引；提供定时自动更新索引的功能；提供多种检索方式。

关键词：Lucene；关系数据库；全文检索

中图分类号：TP391　　**文献标志码**：A　　**文章编号**：1673-9833(2011)02-0081-04

The Design and Implementation of Massive Database Full-Text Retrieval Based on Lucene

Xu Yeqiang，Zhu Yanhui，Li Chunliang，Wang Wenhua

（School of Computer and Communication，Hunan University of Technology，Zhuzhou Hunan 412008，China）

Abstract：Proposes a database full-text retrieval model based on Lucene. Aiming at different source databases configuration, the databases customized via relation database is more flexible than customized via configuration file. The support to kinds of database for extracting, exchanging and indexing based on threads programming and polymorphism are implemented. The function of periodic indexing update and kinds of query requirements are provided.

Keywords: Lucene; relational database; full-text retrieval

0　引言

随着互联网的飞速发展，数据量与日俱增，越来越多的大型企业或集团的核心业务数据都存储在关系数据库管理系统（RDBMS）中。但传统的关系数据库缺乏对存储在库中字段的内容进行检索和分析的核心功能，解决这个问题的关键是建立一条有效的包含数据整合、高速查询、信息分析的、将数据转化为信息的途径。从目前信息科学技术的发展来看，海量信息的全文检索技术是最先进、最适合的解决途径。

国内外相继出现了一些全文检索产品，国内比较有代表性的如易宝北信信息技术有限公司设计和开发的全文信息检索和管理系统TRS等，而国外比较著名的有IBM公司研发的关系型数据库DB2其中的Text Extender，Oracle公司的Oracle Text，Microsoft公司开发的SQL Server 和开源的Lucene[1]全文检索工具包。利用大型关系数据库本身提供的检索服务还有较多不足，所以不适合作为开发平台。而Lucene是Apache软件基金会Jakarta项目组的一个子项目，是一个纯Java编写的开放源代码的全文检索工具包。作为一个开放源代码项目，Lucene自问世之后，

收稿日期：2010-12-17

基金项目：湖南省自然科学基金资助项目（10JJ3002），教育部人文社会科学研究青年基金资助项目（09YJCZH019），中国包装总公司科研基金资助项目（2008-XK13）

作者简介：徐叶强（1982-），男，安徽芜湖人，湖南工业大学硕士生，主要研究方向为文本分类，信息检索，E-mail：x.y.q19820116@163.com

引发了开放源代码社群的巨大反响，程序员们不仅使用它构建具体的全文检索应用，而且将之集成到各种系统软件中去，以及构建Web应用，甚至某些商业软件也采用了Lucene作为其内部全文检索子系统的核心。近几年，学者对基于Lucene全文检索的应用研究层出不穷，如Web页面检索、数据库全文检索、图像检索[2]等。

本文主要研究基于Lucene的海量数据库全文检索。文献[3-4]中结合Hibernate Search技术及Lucene工具包，实现了索引库与数据库的同步，但这种技术也有一定的局限性，因为需要重新开发业务系统，这需要较大的人力、物力、财力，而对海量数据库中数据的分析是没有必要进行实时更新的。文献[5]提出了利用分布式数据库搜索引擎架构来实现智能化的搜索和定位，将数据库中的表格读成xml或doc文件，通过建立和优化索引，将所有字段内容都转化为文本形式再索引，速度较慢。文献[6]中通过实验对数据库SQL查询和基于Lucene全文检索作了比较，证明基于Lucene的数据库全文检索，适用于海量数据的检索和查询。文献[7]给出了一个针对Oracle数据库全文检索的实现方法，但该文中非主键字段只进行索引不存储到索引文件中,这样非主键字段不在内存中，就索引不到那些字段。

综上所述，学者基于Lucene的海量数据库全文检索的研究成果目前还比较少，本文提出了在全文检索系统中使用专门的关系数据库，方便数据源配置。由于待索引的源数据库是经常更新的，所以提出了定时更新索引的方法。同时系统能实现多种检索方式，如智能检索、精确检索、关联表检索、字段组合检索等。

1 数据库全文检索系统的架构和工作原理

数据库全文检索系统的总体架构如图1所示。

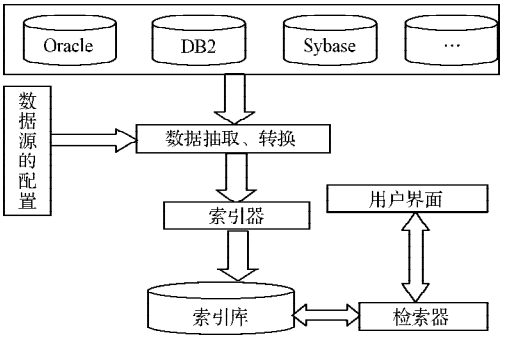

图1 数据库全文检索模型的总体架构

Fig. 1 The structure of database full-text retrieval model

首先在后台配置需要抽取索引的源数据库（Oracle, DB2等），相关的配置信息会保存到全文检索系统的数据库中；然后启动索引器，索引器通过解析配置文件，获取全文检索系统的数据库连接，之后从该库中读取需要抽取索引的源数据库信息，进行数据抽取、转换，建立索引保存到索引库；对于检索过程，通过解析检索条件，检索器可以工作在不同的模式，实现多种检索方式。

1.1 数据源配置模块

对于不同的海量源数据库，创建索引的输入参数以及访问的方式是不同的。文中对不同类型的源数据库实现了统一的接口，还引入了关系数据库来简化建立索引的工作，也就是说，通过算法获取要索引的源数据库信息，通过前台可视化配置索引的输入参数（索引字段名称、是否摘要、是否是时间字段等），剩下的工作由索引任务读取数据库中的配置信息，得到建立索引的相关参数，使用关系数据保存配置信息的这种方式。对于多源数据库，多表，多字段的情况，可以大大减少工作量。

1.2 建立索引模块

该模块主要功能是：创建索引和定时启动索引任务。索引任务读取数据库中的配置信息，通过配置信息读取源数据库的连接，构造SQL语句，读取源数据库中的记录，记录字段经过转换（字典代码转换为汉字，时间字段的转换、大字段的读取等），生成Lucene定义的索引存储单元Document。

考虑到那些经常处于动态更新且海量的源数据库，为保证索引文件与源数据库的一致性，需要不断更新索引的情况，本系统设计了定时全量或增量创建索引的功能。它根据读取配置信息得到索引更新时间和更新方式，创建定时自动运行索引任务。

1.3 检索模块

该模块的创新点在于创建抽象类封装检索条件，不同的检索条件继承并实现这个抽象类，从而实现智能检索、精确检索、关联表检索、字段组合检索。

1）智能检索：在检索框中输入一个或多个关键字进行检索。关键字之间的关系可以是与、或、非的方式。

2）精确检索：即单表查询，表中每个字段都有一个相对应的检索框，各检索框的关系是与的方式。

3）关联表检索：对查询结果中字段再查询，如“身份证号码”字段，查出各库中有这个身份证号码的信息，此过程是对查询结果的二次查询。

4）字段组合检索：输入一个或多个检索条件，每页输出结果数。输出结果将在设定的条件、范围内。

2　主要技术及实现细节

2.1 数据源配置模块及实现

本模块是通过接口实现的，可扩展性强，本文以 Oracle 数据库的实现说明。如图 2 所示 IdbUtil 是一个获取源数据库信息的一个抽象接口，可实现不同类型的数据库（Oracle, DB2, Sybase 等），OracleUtil 是其中一个实现 IdbUtil 接口的类。通过 OracleUtil 可以获取 Oracle 数据库的连接、数据库中的表，以及表中的字段，以 web 页面可视化展示，配置索引的输入参数，保存到全文检索系统的数据库中。

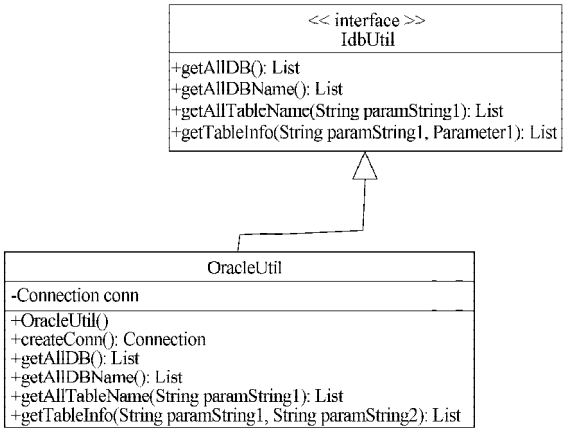

图 2　IdbUtil 接口实现

Fig. 2　Implementation of IdbUtil interface

2.2　多线程建立索引

本系统是针对海量数据库创建索引的，因此采用多线程设计，并发访问性能高，系统资源占用率低，且对服务器硬件资源要求低。使用 Oswego 大学 Doug Lea 编写并发实用程序开放源码库 util.concurrent 包，编写一个类 DbUrlHandlePool 封装 util.concurrent 的线程池，从本系统的关系数据表中取出多条与数据库对应的 url，然后创建线程，一条 url 创建一个线程，然后根据 url 对数据库中记录抽取、转换、建立索引。

目前索引的建立是针对文本进行的，但本文是针对数据库进行的。因此，在建立索引前要把源数据库中的记录读取出来，每条记录中的每个字段进行判断是否需要特殊处理，处理过程如下：

1）时间字段。进行规范化处理成 "18500101" 格式，以便可以进行时间段查询。

2）大字段。主要是文本字段，而不是图片字段，转化成 xml 文件。

3）身份证号码字段。判断是否是合法的身份证号码。

4）是否是字典项。如果是，进行代码与汉字的转换。

……

可根据查询的不同应用，添加不同的处理函数。源数据库中一条记录对应索引文件中的一个 Document，每个字段对应 Document 中的一个 Field，对每个 Field 进行存储，索引。

本检索系统有专门的关系数据保存配置信息，对资源进行分类。同时一个数据表产生一个索引文件，命名规则是：数据库名 + 表名，可以把这个表划分到一个资源类别中，可提高检索效率。

2.3　索引任务定时运行

定时运行自动更新索引程序，读取任务配置文件，文件如下：

#==== 任务调度参数 ====

抓取线程数，同时不超过两线程

thread.number=2

每隔 4 个小时执行一次

time=1 1 /4 * * ?

任务调度和索引更新流程分别如图 3 和 4 所示。

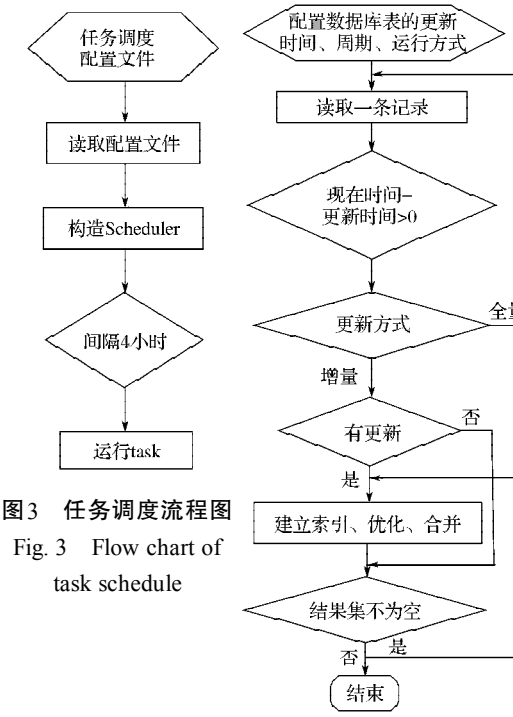

图 3　任务调度流程图

Fig. 3　Flow chart of task schedule

图 4　索引更新流程图

Fig. 4　Flow chart of indexing update

任务调度运行程序使用 Quartz，它是个开源的作业调度框架，允许开发人员根据时间间隔（或天）来调度作业。Quartz 调度包的 2 个基本单元是作业和触发器。本系统使用 Cron 触发器实现对任务执行的调度，根据配置文件设定的时间，每 4 个小时运行一次 task，从数据库里读取每个表更新的时间以及更新方

式（全量、增量）。如果某个表的更新时间早于当前时间，增量运行方式采用时间戳判断是否有新数据入库，如果没有新数据入库，根据设定的间隔周期，计算下一次更新的时间，而在全量运行方式下，则每次全量从源数据库中创建索引。

2.4 数据检索方式与返回结果

系统检索的对象是Lucene索引，当用户选择相应的查询条件并输入关键字后，系统会自动产生相应的查询条件对象，不同检索方式的查询条件类都继承于抽象类searchValue，通过判断searchValue.getMode()的值，决定是智能检索、精确检索、关联表检索还是字段组合检索。检索界面如图5所示。

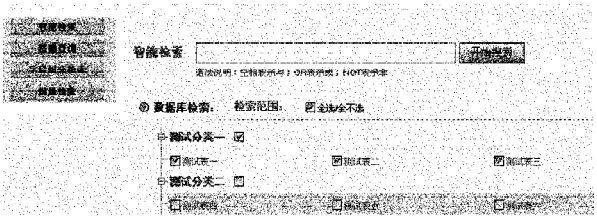

图5 用户查询界面

Fig. 5 Interface for users searching

代码实现如下所示：

```
public void Search() {
    if(this.searchValue != null)
      if(this.searchValue.getMode() == 4)
        relatedTableSearch();
      else if(this.searchValue.getMode() == 3)
        multiResultTableSearch();
      else if(this.searchValue.getMode() == 2)
        ClassifyTableSearch();
      else if(this.searchValue.getMode() == 1)
        multiTableSearch();
      else if(this.searchValue.getMode() == 0)
        singleTableSearch();
```

关于查询结果，对于含有图片的字段，是通过查出主键字段，再连接到源数据库进行读取，如果图片字段直接存储到索引文件中，会占用大量的磁盘空间。

3 实验分析

实验的硬件设备：CPU是Intel P6000双核，主频1.87 GHz，内存2 G。实验采用Oracle数据库。更新索引的时间：从源数据库中抽取需更新的数据、转换、索引，再把索引合并到旧的索引中。更新索引的时间经测试后其详细时间如表1所示。

表1 更新索引时间表

Table 1 Time table of indexing update

记录数/个	字段数/个	需转换的字段数/个	时间/s
5 211	67	10	179
5 733	46	11	191
117	51	9	90
7 010	66	20	61
3 480 583	10	1	2 433

实验结果表明，由于采用多线程创建索引程序，更新记录数较少时，建索引用的时间并无规律，更新的记录数较多时建立索引的消耗时间明显增多。

对于数据库创建索引时的效率，主要是由数据库的存取速度决定的。对海量关系数据库更新索引，效率可能会低，但并不影响对数据库建立索引的意义，通过对数据库建立Lucene索引能有效提高数据转化为信息的效率。

数据库表中的记录在亿数量级的时候，检索响应时间在0.1 s以内，完全可以满足用户的需求。

4 结语

本文结合Lucene开源工具包，把互联网全文检索技术应用到对数据库的检索上，实现了一个能对关系数据库中的数据进行全文检索的原型。它支持多种类型的关系数据库，并且具有定时更新索引的功能。提供了多种检索方式，提高了检索的精度和效率。

参考文献：

[1] Lucene工作组. Welcome to Apache Lucene[EB/OL].[2010-04-01]. http://lucene.apache.org/.
Lucene workgroup. Welcome to Apache Lucene[EB/OL]. [2010-04-01]. http://lucene.apache.org/.

[2] Lux Mathias，Savvas A. Chatzichristofis：Lire： Lucene Image Retrieval：An Extensible Java CBIR Library[C]//In Proceedings of the 16th ACM International Conference on Multimedia. New York：ACM，2008：1085-1088.

[3] 阳 奇，林镇灿，黄 帆，等. 基于Hibernate搜索的数据库全文检索系统[J]. 计算机工程，2010，36(4)：74-76.
Yang Qi，Lin Zhencan，Huang Fan，et al. Database Full-Text Retrieval System Based on Hibernate Search[J]. Computer Engineering，2010，36(4)：74-76.

[4] 王 彬，张计龙，徐迎晓. 整合数据持久化与全文检索的新方法[J]. 计算机工程，2009，35(3)：42-44.
Wang Bin，Zhang Jilong，Xu Yingxiao. New Method of

（下转第103页）