迅速搭建全文搜索平台

——开源搜索引擎实战教程

于天恩 编著

清华大学出版社
http://www.tup.com.cn
北京交通大学出版社
http://press.bjtu.edu.cn

迅速搭建全文搜索平台

——开源搜索引擎实战教程

于天恩　编著

清华大学出版社
北京交通大学出版社
·北京·

内 容 简 介

本书作为有心进入搜索引擎业的读者的第二本基础书籍，承接其兄弟篇，讲解了开源搜索引擎的搭建过程中所要解决的基本问题，将搜索引擎这一高起点的技术讲解得清晰透彻，使其变得极为好学，没有任何神秘可言。本书共包括5章，可以分成两个部分。

第一部分（第1章）：建立搜索引擎的方案。这部分用数少的文字总结建立搜索引擎的主要方案，即：常规的数据库搜索、文件搜索，基于数据库全文索引机制的搜索，利用外部非开源Web搜索服务进行的搜索，以及利用开源搜索引擎实现的搜索。

第二部分（第2～5章）：架设网络搜索引擎。从第2章起，陆续介绍数据抓取、数据解析、建立索引和执行搜索这四项内容，它们是创建网络搜索平台所要解决的基本问题；第5章，介绍基于Hyper Estraier搜索引擎框架来搭建桌面搜索引擎和Web搜索引擎的方法，给出了相关的案例。

本书所有源代码都放在出版社的网站上（http：//press.bjtu.edu.cn），读者可以免费下载。

图书在版编目(CIP)数据

迅速搭建全文搜索平台：开源搜索引擎实战教程／于天恩编著. —北京：清华大学出版社；北京交通大学出版社，2007.10

ISBN 978－7－81123－156－4

Ⅰ.迅…　Ⅱ.于…　Ⅲ.互联网络－情报检索－教材　Ⅳ.G354.4

中国版本图书馆CIP数据核字（2007）第132671号

责任编辑：谭文芳
出版发行：清华大学出版社　邮编：100084　电话：010－62776969
　　北京交通大学出版社　邮编：100044　电话：010－51686414
印 刷 者：北京交大印刷厂
经　　销：全国新华书店
开　　本：185×260　印张：18.75　字数：480千字
版　　次：2007年10月第1版　2007年10月第1次印刷
书　　号：ISBN 978－7－81123－156－4/TP·383
印　　数：1～4 000册　定价：32.00元

前　言

说说搜索引擎

搜索引擎这几年热起来了。

作为世界上最大、最出名的搜索引擎，Google 在很多方面都发挥了重要的作用。

但是，当手中没有 Google 的搜索代码时，该如何搭建一个自己的搜索引擎呢？业界的人士说，全新开发一套完备的企业级搜索引擎要五年的时间。诚然，许多“业界”人士的话并不可信，不过，在搜索引擎这一块，真想要做好确实是不容易。

开发搜索引擎要耗费大量的时间和精力，所以有一些人开始研发独立的搜索引擎模块，并将其源代码开放，这样就可以给其他需要建立自己的搜索引擎的人提供一个基础平台。在这些开源搜索引擎模块的基础上做开发，可以节约非常多的时间和精力，大大减少了开发成本，缩短了产品投入市场的周期。而且，由于这些平台是开源的，可以亲自检查每一行代码，修改算法和显示格式等内容，这样的搜索引擎就相当于自己写的，用起来放心。

有时使用某些商业搜索模块，尽管搜索效果也很好，但是很难知道在单击“搜索”按钮的瞬间自己是否做了一些自己并不想做的事情，比如：给某个陌生人发送了一个特洛伊木马。

写这本书的动机

开源搜索引擎对解决企业搜索等问题提供了可靠的二次开发平台（有的甚至不需要二次开发），大大提高了开发搜索引擎的效率，缩减了成本，好处多多。所以，需要有一些书来介绍如何使用开源搜索模块来提供搜索服务，而目前市面上这类书籍并不多。

我编写的这本书——《迅速搭建全文搜索平台——开源搜索引擎实战教程》（以下简称《实战教程》），是《做自己的搜索引擎——搜索引擎精解案例教程》（以下简称《案例教程》）的兄弟篇，用以介绍开源搜索引擎的架构和实现。

《案例教程》和《实战教程》这两本书是非常有意义的，前者介绍搜索引擎的理论和基本应用，后者介绍在开源搜索引擎领域中如何实现搜索引擎的搭建。有了这两本书，一个普通的程序员就可以顺利并且十分容易地掌握与搜索引擎相关的核心知识。看过这两本书之后，就有能力深入地研究主流的开源搜索引擎的代码，之后，就成为优秀的搜索引擎工程师。

按照普通人的观点，从普通的程序员到搜索引擎工程师，这两者之间是有三级台阶的。

第一级：了解搜索引擎的原理和相关术语等基础知识。

第二级：了解现存的搜索引擎是如何运行的，懂得如何应用搜索引擎的原理去搭建搜索引擎。

第三级：认真研究一种或几种开源搜索引擎的源代码，深刻地理解其架构，从而使之成为相当于自己开发的搜索引擎。

前两级台阶，用《案例教程》和《实战教程》可以顺利解决，这正是我编写这两本书的目的。要走第三级台阶，如果看书的话，就得看专门讲解一种搜索引擎框架的书。笔者在这里推荐《Lucene in Action》。但，仅靠这本书还不够，还需要认真去研究其代码才行，尝试去修改代码、增加功能。

由于《Lucene in Action》在版本上的确是有些旧了，所以，我也打算在有时间的时候写一些关于 Lucene 或 Hyper Estraier 的架构原理和代码分析的书。

之后，我还希望写一些书来分析搜索引擎的数学算法：从蜘蛛、索引到排序，这样搜索引擎知识方面的相关书籍就全面了。

这本书的内容

本书作为有心进入搜索引擎业的读者的基础书籍，承接其兄弟篇，讲解了开源搜索引擎的搭建过程中所要解决的基本问题，将搜索引擎这一高起点的技术讲解得清晰透彻，使其变得极为好学，没有任何神秘可言。

本书共包括 5 章，可以分成两个部分。

第一部分（第 1 章）：建立搜索引擎的方案。用尽可能少的文字总结建立搜索引擎的主要方案，即：常规的数据库搜索、文件搜索，基于数据库全文索引机制的搜索，利用外部非开源 Web 搜索服务进行的搜索，以及利用开源搜索引擎实现的搜索。

第二部分（第 2～5 章）：架设网络搜索引擎。从第 2 章起，陆续介绍数据抓取、数据解析、建立索引和执行搜索这四项内容。这四项内容是创建网络搜索平台所要解决的基本问题。第 5 章，介绍基于 Hyper Estraier 搜索引擎框架来搭建桌面搜索引擎和 Web 搜索引擎的方法，给出了相关的案例。

需要留意的是，书中有些程序的实现没有采用最佳结构和最高效的算法，这是为了使程序更加易读、易懂。

这本书的特点

本书的内容以实践为主，并不深挖理论。用石志国博士的话来讲，就是“理论联系实际，并有所发展”。这是他写书的特点，同样适用于我。

本书以实践为主，也包含了必要的理论，但所讲的理论都不是纸上谈兵，而是可以立即付诸实践进行工程应用。书中的代码可以直接拿出来用（只是不要忘了输入信息验证等基本的安全检查）。

本书由浅入深地提供了大量的案例。浅，并不从“什么是程序设计”开始，所以读者需要具有一些编程的基础知识；深，并没有深到“只可意会，无法言传”的地步，所以读者也不需要担心无法看懂。

选择本书的理由

我始终认为翻译别人的书容易，只要照实翻就行，可是自己写书就不是很容易，一定要写有用的书，写好书。以下几点可以成为读者选择本书的理由。

第一，搜索引擎很热，构建搜索引擎也很有用。这本书可以使读者轻松掌握搭建搜索引擎

平台所需的核心知识，并能轻松搭建起自己的搜索引擎平台。

第二，这本书讲解详细，书中涉及的代码将全部提供。使用这些代码，即使不进行修改，也可以建立起一个中型的搜索平台了。

第三，网上有些文档，多半是英文的，还有其他国家语言的。对于外文不太好的读者来说，理解起来难免会有误差和困难。这本书，至少比直接翻译外国文档要强一些，看起来会很轻松。

第四，即便网上所有的文档都是中文写的，我依然认为买一本书来看比在网上浏览要好得多。人要爱护眼睛，软件工程师尤其是这样。

兄弟篇

这本书的兄弟篇《案例教程》介绍了搜索引擎的历史及当前的发展状况，与搜索引擎相关的公司、技术以及搜索引擎对人类生活的影响；讲解了搜索引擎的原理和相关技术；介绍了搜索引擎的基本构造方法，实现索引建立和搜索的基本算法；在基于数据库的全文检索方面，介绍了通常采用数据库的全文索引服务时搜索引擎的基本搭建方式。对于 Windows 索引服务，专门稍微细致地进行了介绍。

《案例教程》适合于作为学习搜索引擎的基础和入门教程，里面的数据库全文搜索部分对于希望利用数据库实现搜索引擎的人大有益处。

在《案例教程》的基础上，《实战教程》继续讲解开源搜索引擎的原理和实现方案，所以对搜索引擎的基本原理和相关概念都略去不提，如果读者在这些方面遇到问题，请参考《案例教程》。

谅解和支持

本书从章节的安排到案例的编写，都经过了仔细揣摩，力图做到最好。然而，没有最好，只有更好。

本书尽力做到精炼，且没有附加光盘以减少读者的购书成本。本书的所有源代码都放在出版社的网站上（http://press.bjtu.edu.cn），读者可以免费下载。

在这本书中发现任何问题，皆望能与笔者联系，以使本书臻于完善。笔者的 E-mail：yutianen@163.com。

衷心感谢

在我的成长过程中，得到过许多人的关心和鼓舞，他们启迪了我的思维，拓宽了我的视野，如果人生是在沙漠中旅行，他们就是眼前的足印和身上的水。

在本书的写作过程中，得到了许多人的支持和鼓励，他们是：哈尔滨工业大学语音处理研究室的李海峰老师，校部机关的蔡德彰、李新美、曲洪勤、黄峰、冯健、孔祥钰等老师，热能动力工程研究所的周逊老师，传统工业基地转型研究所的陈晓东老师，软件学院的田英鑫老师，网络与信息中心的杨庆海、何慧、李亚平、王宇航等老师，研究生院的彭远奎、朱群益、张思琦、王晓磊、雷稚蔷等老师，计时器研究所的王晓溪老师，图书馆的耿小兵老师，外国语学院的王桂芝、常巍（Sabrina）等老师，机器人研究所的蔡鹤皋院士，控制理论与制导技术研究中心的段广

仁、尹航、周彬老师，科学园的宋斌、刘弋滢等老师，计算机学院的刘开昌老师，等等。

他们都是我的良师益友，是我心中的动力，每当想到他们，我总觉得自己应该放弃休息，去做更多有益的事，将真诚与善良传递下去。

在这里，对他们表示衷心的感谢！

同时，对哈工大天萌联合的一切成员表示感谢！那些曾跟我在一起的朋友，我会记得你们为我泡的每一杯咖啡和茶。那些始终保持独立的朋友，我也祝愿你们会有更加辉煌的未来。天萌联合永远是哈工大最强、最自由的社团，你们这些天萌元老的名字，将永远铭刻在哈工大的历史上，铭刻在我的心里。

另外，需要特别感谢：

石志国博士，他的《ASP 精解案例教程》一书是我学习编程的开端。他是个高尚的人。

顾倩萌，让我又怜又爱，时刻挂心。她的爱是股特殊的强大力量，让我找回记忆、重新开始唱歌、安静、宁神。她是我最爱的小月，是我唯一的轻松和仅存的快乐。没有小月，生命不该开始。没有小月，一切都没有意义。

于天恩

2007 年 8 月　哈工大　天人居

目　录

第一部分　建立搜索引擎的方案

第二部分　架设网络搜索引擎

第一部分

建立搜索引擎的方案

第 1 章　建立搜索引擎的方案

本章要点

本章总结了建立搜索引擎的主要方案，对开源搜索引擎的实现原理作了揭示。

1.1　建立搜索引擎的基本方案

如何建立搜索引擎？

基本方法有如下四种。

(1) 常规的数据库搜索

使用“like”、“between”等谓词，或者数据库自带的“instr”等字符串函数。基于这种原理建立的搜索引擎在数据量非常小的情况下是很有效的。

(2) 常规的文件搜索

常规的文件搜索就是对文件下的文件进行遍历，用搜索关键词与每个文件的内容进行对比。这个方法可以用于少量文件的搜索。

(3) 基于数据库全文搜索功能的搜索

利用数据库自带的全文搜索功能，可以解决几百万条记录的数据库搜索问题，这样实现的全文搜索引擎性能是不错的。如果能做好软硬件优化，搜索的效果就会更好。

(4) 基于 Windows 索引服务的全文搜索

使用 Windows 的索引服务，可以对大量文件建立起全文索引，然后执行快速的全文搜索。

考虑到知识的系统性，在这里对这四种建立搜索引擎的基本方式进行简单的回顾。

1.1.1　常规的数据库搜索

常规的数据库搜索，就是基于标准 SQL 语句的句法进行的数据查询。通常要使用“=”进行精确搜索，使用“between”、“in”符号设定范围搜索，使用“like”实现模糊搜索。结合数据库的内置函数(如：instr 等)，还可以实现其他类型的搜索，因数据库而异。

下面举一个例子，使用 JSP 2.4 技术，利用 Oracle 9i 数据库中的 scott 用户的 dept 表实现常规的数据库搜索。

(注：本书程序中加粗部分为关健代码。)

案例名称：常规的数据库搜索

程序名称：fuzzy.jsp

```
<%@ page contentType="text/html;charset=gbk" %>
<%@ page import="java.sql.*" %>
<html>
<body>
```

```
<%
try
{
    Class.forName("oracle.jdbc.driver.OracleDriver");
}
catch(ClassNotFoundException e)
{
    out.print(e.getMessage());
}

Connection conn=null;
Statement stmt=null;
ResultSet rs=null;

try
{
    String url="jdbc:oracle:thin:@localhost:1521:yuanyuan";
    String user="scott";
    String password="tiger";
    conn= DriverManager.getConnection(url,user,password);
    stmt=conn.createStatement();
    String sql="select * from dept where DNAME like '%TI%'";
    rs=stmt.executeQuery(sql);
    while(rs.next())
    {
        String a=rs.getString("deptno");
        String b=rs.getString("dname");
        String c=rs.getString("loc");

out.print(a+"        "+b+"        "+c+"<br
>");
    }
}
catch(SQLException ee)
{
    out.print(ee.getMessage());
}
finally
{
    if(rs!=null)
    {
        try
        {
            rs.close();
        }
        catch(SQLException ex)
        {
            out.print(ex.getMessage());
        }
```

```
        }
        if(stmt! =null)
        {
            try
            {
                stmt.close();
            }
            catch(SQLException ex)
            {
                out.print(ex.getMessage());
            }
        }
        if(conn! =null)
        {
            try
            {
                conn.close();
            }
            catch(SQLException ex)
            {
                out.print(ex.getMessage());
            }
        }
    }
    %>
    </body>
    </html>
```

程序运行结果如图 1-1 所示。

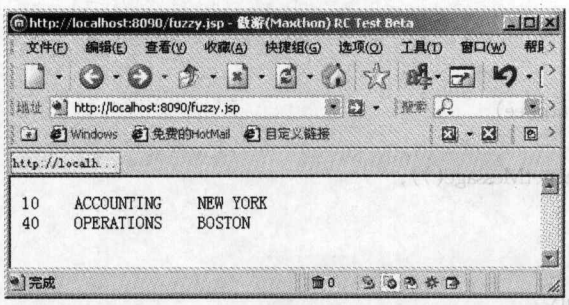

图 1-1 常规的数据库搜索

1.1.2 常规的文件搜索

常规的文件搜索就是对文件下的文件进行遍历,用搜索关键词和每个文件的内容进行对比。这个方法可以用于少量文件的搜索。

下面举一个例子,使用 Java 技术,对一个文件夹下的所有文件进行遍历,读取每个文件的内容与搜索关键词进行对比,从而实现全文搜索。

案例名称:常规的文件搜索

程序名称:FileSearch.java

```
package tianen;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;
import java.io.BufferedReader;

class FileSearch
{
    /* * *
     * 递归遍历文件
     * 因为目录层次复杂都要考虑到
     *
     * @throws IOException
     */
    StringBuffer sb = new StringBuffer("");
    public String getFiles(File f,String s) throws IOException
{
    if(f.isDirectory())
    {
        File[] fs = f.listFiles();
        for(int i = 0;i < fs.length;i ++ )
        {
            getFiles(fs[i],s);
        }
    }
    else
    {
        if(this.getText(f).indexOf(s)! = -1)
        {
            sb.append(f.getPath() + "\n");
        }
    }
    return sb.toString();
}

public String getText(File path) throws IOException
{
    FileReader fr = new FileReader(path);
    BufferedReader br = new BufferedReader(fr);

    String s = br.readLine();
    StringBuffer sb = new StringBuffer("");
    while(s! = null)
    {
        sb.append(s);
        s = br.readLine();
```

```
            }
            br.close();
            return sb.toString();
        }

        public static void main(String[] aa) throws IOException
        {
            String path="D:\\jsp\\docs";
            File f=new File(path);
            String a=new FileSearch().getFiles(f,"大禹治水");
            System.out.println(a);
        }
    }
```

这个程序实现了对“D:\jsp\docs”目录下的所有文件进行遍历,读取每个文件的内容,与“大禹治水”进行对比。从而取出所有含有“大禹治水”的文件。

“D:\jsp\docs”目录的情况如图 1-2 所示。

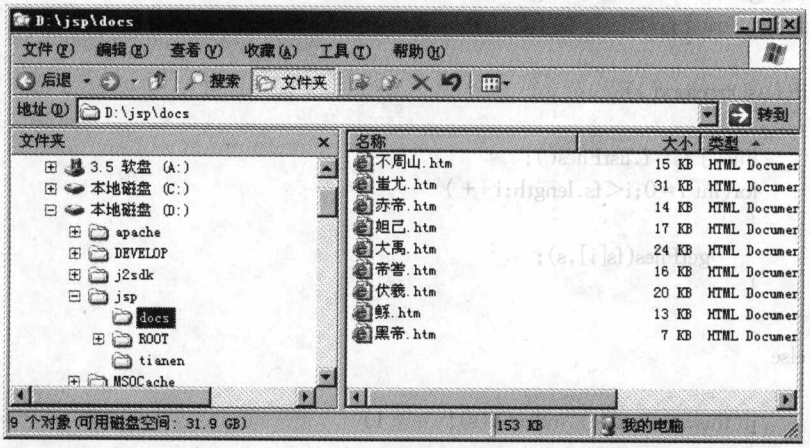

图 1-2 目录的情况

程序运行结果如图 1-3 所示。

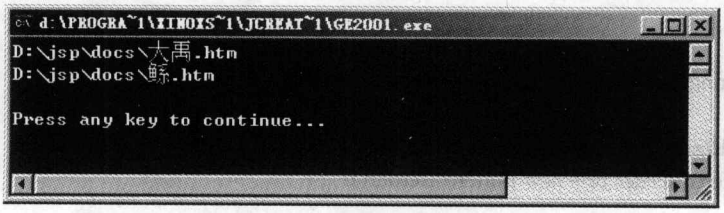

图 1-3 常规的文件搜索

这里要注意:遍历的时候要考虑到子文件,在当前目录下发现文件夹时,就要进入子文件夹取文件,不管目录结构多复杂,都要使程序把当前目录下的所有文件都找出来。

这个程序可以很容易地改写成 Web 的形式,如以下程序所示。

程序名称:FileSearch.java(Servlet)

```
//FileSearch.java
package tianen;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class FileSearch extends HttpServlet
{
public void service(HttpServletRequest request,HttpServletResponse response)
    throws IOException
{
    response.setContentType("text/html;charset=GBK");
    PrintWriter out=response.getWriter();

    String path="D:\jsp\docs";
    File f=new File(path);
    String a=new FileSearch().getFiles(f,"大禹治水");
    out.print("符合条件的结果如下:<p>");
    out.print(a);
}

StringBuffer sb=new StringBuffer("");
public String getFiles(File f,String s) throws IOException
{
    if(f.isDirectory())
    {
        File[] fs=f.listFiles();
        for(int i=0;i<fs.length;i++)
        {
            getFiles(fs[i],s);
        }
    }
    else
    {
        if(this.getText(f).indexOf(s)!=-1)
        {
            sb.append("<a href="+f.getPath()+">"+f.getPath()+"</a><br>");
        }
    }
    return sb.toString();
}

public String getText(File path) throws IOException
{
    FileReader fr=new FileReader(path);
    BufferedReader br=new BufferedReader(fr);
```