FTP 搜索引擎的设计与实现（优化版）

By falcon

摘要：FTP 是因特网最主要的服务之一，FTP 搜索引擎为资源共享提供了极大的方便。本文分析和设计了一个基于 WEB 的 FTP 搜索引擎，在 ASP+ACCESS+VB 环境下给出了编程实现，并体现了具体实践中总结出的一些经验。

关键词：FTP；搜索引擎；ASP；ACCESS;SQL;VB

引言

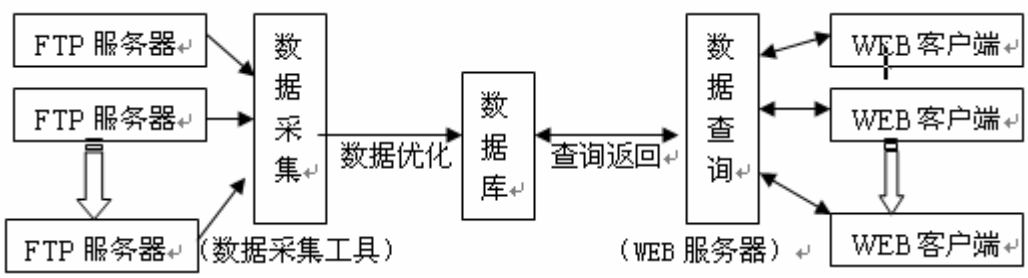
FTP 是因特网最主要的服务之一，在 FTP 服务器上保存有大量的各种各样的共享软件、技术资料和多媒体数据等文件。由于 FTP 服务器上的文件组织结构比较复杂，而且文件分布不一定有规律，因此想很快的找到自己需要的资源比较困难。基于 WEB 的 FTP 搜索引擎可以很好的解决上述问题。目前，国内外有很多 FTP 搜索引擎，国内高校中做得比较出色的有天网资源搜索（北大）、歪酷搜索（复旦）、星空搜索（清华）。如果能够打造一个我校自己的 FTP 搜索引擎，将极大的方便校园网用户共享 FTP 资源，进而对校园网的发展作出巨大的贡献。

正文

1．FTP 搜索引擎的结构

FTP 搜索引擎由数据采集、数据查询、数据管理和维护三大核心模块组成。实现一个 FTP 搜索引擎，首先要获得 FTP 服务器上的文件信息，并保存到数据库中；然后为用户提供一个查询界面，通过把用户提交的关键字转化成数据库查询语言，从库中获取匹配的文件信息，并以友好的界面返回给用户。另外，为了保持数据库和服务器信息的同步更新、保证库中数

据量等，引入数据管理和维护的功能模块。下面是 FTP 搜索引擎的结构图示：



使用的相关软件和语言说明：

操作系统:Windows XP

WEB 服务器:IIS

数据库查询语言:SQL

数据库系统:ACCESS

WEB 页编程语言:html+asp+javascript

数据采集工具编程语言:VB。

2. 数据库结构的分析和设计

从 FTP 搜索引擎的结构图示中不难看出，数据库是整个系统的核心部分。它是三大功能模块设计的基础。下面先对三个模块和数据库之间的关系进行一定的分析。

数据采集模块需要通过访问 FTP 服务器获取文件信息，那么不难想到把 FTP 站点信息和文件信息保存到数据库中。另外数据查询模块的主要操作对象也正好是文件信息，因此需要访问数据库。另外，为了方便用户更好的查询，引入关键字统计等模块并把统计信息保存到数据库中。而数据管理和维护的对象涉及对站点信息的更新以及文件信息与服务器的同步，比如添加删除站点，重新获取 FTP 服务器的文件信息等，也涉及到数据库的操作。

下面就文件信息、站点信息、关键字统计信息等几个方面考虑数据库的设计。

2.1 文件信息分析

在 FTP 服务器上，文件系统都是树形结构的，其中包括目录和文件，而每个文件包括文件名（隐含类型）、文件地址、文件大小、日期等信息，其中最为重要的信息莫过于文件地址，用户通过它可以进行下载等操作。据此，我们可以设计出文件信息表，但是出于对数据

库冗余度，以及必要的优化等方面考虑，我们先从文件系统结构、文件信息的自身性质等进行进一步分析。

1）文件系统的结构是树形的，我们必须设法把它转换成现在流行的关系数据库中的表结构。

2）如果每个文件的 URL 地址都保存到一个字段中，那么将由于他们有相同的目录而产生极大的数据冗余。因此应该设法把冗余度降低。

3）通常要获得的是文件而非目录。因此可以设想把文件和目录保存到不同的表中。

4）由于文件一般都由文件后缀来指定文件类型，因此可以据此设置一个类型字段来对文件进行分类，甚至可以据此把文件表划分成更细的表。

通过分析后不难想到引入类似静态链表中的游标来建立树形结构和表结构的联系，于是，我们设计出这样的表结构：（目前只划分为【目录表】和【文件表】来存放文件信息）

【目录表】：cat_tab

字段名	字段类型
目录编号(id)	Integer
目录名(cat)	Text(50)
父目录编号(pid)	Integer
所属 IP 站点编号（ipid）	Integer
访问次数(acctime)	long

这里的 id 和 pid 实现了类似静态链表中的游标，极大的优化了数据库，而且使得查询操作更加方便。而 ipid 是该表和站点表之间的关系字段，指定目录所属的站点，至于访问次数，是为了使得查询返回结果更人性化，下同。

【文件表】：file_tab

字段名	字段类型
文件名(file)	Text(50)
后缀(postfix)	Text(4)
父目录编号(pid)	Integer
所属 IP 站点编号（ipid）	Integer
访问次数(acctime)	long

这里的 pid 指向目录表中的 id，据此建立文件表目录表之间的联系。

2.2 FTP 站点信息分析

一个 FTP 站点通常包括服务器名（域名或者 IP）、开放的端口号、用户名和密码、站点说明信息等。保存 FTP 站点信息表的结构具体设计如下：

【站点表】：site_tab

字段名	字段类型
IP 编号(id)	Integer
站点地址(site)	Text(15)
端口（port）,默认为 21 端口	Integer
登陆用户名(user),默认为 anonymous	Text(10)
登陆密码（pw）,默认为 falcon@	Text(10)
当前是否可以访问(acc)	Boolean
该站数据是否已经入库(indb)	Boolean
站点说明信息（info）	备注型

“当前是否可以访问”字段，是为数据管理和维护模块以及数据查询模块设计的，前者通过“站点连接测试”获取该信息，而数据查询模块据此判断是否返回该站点的信息给用户，从而保证返回结果的有效性。而该站点数据是否入库与是否可访问信息结合起来作为查询的 ipid（站点 id 编号）的域。另外，它可以为数据采集时是否还需要采集该站点数据作依据。

2.3 关键字统计信息分析

经过实践发现，对关键字的统计非常重要，它可以帮助用户更好的查询。该信息可以单独提供给用户访问，也可以在用户每次访问时把类似的关键字信息显示给用户，让用户通过对比观察其他用户查询的关键字而找到更好的查询办法。目前该表只设置关键字信息和被成功搜索次数信息。

【关键字统计表】key_tab

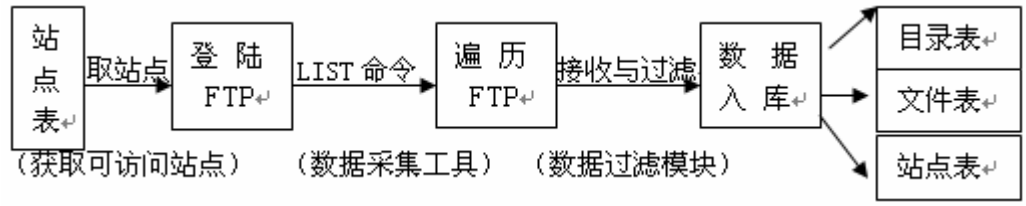
字段名	字段类型
关键字(key)	Text(25)
访问次数(acctime)	Long

2.4 数据库结构设计

经过分析和初步的设计，我们的数据库基本成型。它包括四个表，分别是【目录表】、【文件表】、【站点表】、【关键字表】，表名分别为 cat_tab,file_tab,site_tab,key_tab。出于系统整体风格的考虑，所有的命名全部采用小写以及特殊文件名命名的见名知义原则。另外该数据库的名字根据站点的名字而来,取为 falcon_search.mdb,为了在数据库安全方面的考虑，实际命名为#falcon_search.asp。

3. 数据采集

要构建 FTP 搜索引擎，必须通过自动方式采集 FTP 服务器上的文件信息，并以相对完善的格式保存到上面设计好的数据库中。这个过程大体如下：



该模块先从站点表中获取站点的必须信息进行站点登陆、遍历等操作，与此同时接收 FTP 站点返回的文件信息，并进行一定的优化后把目录和文件分别保存到目录表和文件表中，另外在数据入库后把站点表中是否入库标记置为真。这里涉及到两个个子核心模块：站点遍历以及数据过滤。

前者可以通过站点遍历过程实现（可以是递归的，也可以用堆栈实现的非递归过程）；而后者的数据过滤涉及到对大量站点数据的统计分析后才能做出妥善的安排，从而为提高整个站点的性能做出非常大的贡献，比如，过滤掉一些“黄色”字眼可以净化网络环境，过滤掉垃圾文件可以精简数据库的到小并且使得用户能够获得尽可能满足要求的信息，而对某些特殊的文件名进行过滤不但可以压缩数据库，而且可以提高访问效率。比如，没有意义的文件名（如纯数字的），用户永远都不会去查询即使查询到也不知道该文件是否满足需要，因此把这类文件信息过滤掉也相当重要，不过，它们所处的目录命名往往是有意义的，因此保留目录信息不但可以方便用户查询到相关信息，而且上面过滤掉的文件信息实质上依然可以为用户所利用，即可以通过它们所在的目录访问到。

下面先把数据采集的算法用自然语言描述如下：

```
数据采集过程{
    从库中读取站点信息;
    登陆站点并遍历;
    接收返回信息并过滤;
    数据入库;
}
```

然后我们具体分析每个细节：

3.1 从库中读取站点信息

这里我们需要设计一个数据结构来保存站点数据，设计时完全参照表中的相关字段设置。下面主要介绍 VB 连接数据库的操作，以及记录型数据的设计。

A，VB 连接数据库：

先引用 Microsoft DAO 3.6 Object Library,然后就可以通过下面的代码实现对 Access 的连接。

```
Set mydb = DBEngine.OpenDatabase(“数据库的绝对路径”)
```

至于执行 sql 语句，可以这样实现：

```
Set myrs = mydb.OpenRecordset(sql 语句)
```

B，记录型数据的设计:

我们来设计一个存放每个站点信息的记录型数据结构：

```
Const siteLen = 15
Const userLen = 10
Const pwLen = 10
Const infoLen = 1000
Public Type siteStructure
    Id As Integer
    Site As String * siteLen
    Port As Integer
    User As String * userLen
    Pw As String * pwLen
    Acc As Boolean
    Indb As Boolean
    Info As String * infoLen
End Type
```

注：上面声明了一些常量，目的是为了方便日后的系统的维护。

3.2 登陆站点和遍历以及数据过滤

实际操作过程中，我们把“站点遍历”和“数据接收“放到一起考虑。为什么把它们放

在一起呢？因为遍历的过程也是接收数据的过程，它们并不是纯粹的顺序关系，而是并行关系。该部分是数据采集过程的核心。因此需要详细的分析。

为了实现站点的登陆和遍历，我们有两种实现方案。

方案一，可以使用 vb6.0 提供的 Winsock 控件，它提供了基于 TCP 协议的套接字连接，可以通过它实现 ftp 协议，而我们要做的仅仅是实现 ftp 协议客户端，并且只需实现登陆、发送 list 命令和接收返回信息即可。

方案二，vb6.0 也提供了一个更高级的 internet transfer control6.0,该控件使得底层的 ftp 协议透明化，我们无需知道 ftp 协议的底层实现过程就可以通过它进行类似 dos 下 ftp 命令操作。用 dir 命令就可以实现对对方目录的文件列表操作，而且可以在该控件的状态改变事件中判断状态并在合适状态下接收数据。

通过分析，不难发现，两者都可以实现我们的需求。前者在操作控制方面比后者麻烦，些，不过比后者灵活，比如超时控制更自由。而且前者有个优点，容易通过 load（装载）多个控件来实现多线程。

为了设计的方便，我们这里采取第二个方案。我们把这“站点遍历”和“数据接收”分成两个过程来考虑。站点登陆和遍历过程实际上是一个命令发送过程，而后面的接收数据和数据过程可以放到控件的状态改变过程中实现。两个过程的细化如下：

命令发送过程{

（登陆并进入服务器的根目录）

通过递归实现进入子目录，直到遍历完所有目录

（通过引入堆栈来实现非递归算法，后序遍历树）

}

控件的状态改变过程{

判断服务器是否响应完成，如果是执行下面操作：

接收数据

数据过滤

数据存入临时变量中

}

同样，为了保存每个文件的信息，我们得设计一个存放文件的记录型数据。每个文件包括文件编号、父目录编号、文件名（或目录名）、文件后缀（目录后缀为”/”，文件后缀从文件名拆分出来）、文件深度（指所处文件系统中目录的层次，这是为了便于遍历而设置，并不入库）另外，为了实现非递归，我们设计一个堆栈来存放遍历过程中的目录，为了保存

最终的结果，我们设计一个数组（即上面提到的临时变量）来存放。由于目录个数和文件信息总量的预先不可估量，我们采取动态数组来实现它们。

这里记录型数据的设计同 3.1，这里介绍一下 VB 中动态声明数据的方法：

先通过 Dim 定义一个空的数组，然后通过 Redim 重新定义，如：

```
Dim TestArray() as string   ‘以前的声明
Redim preserve TestArray(数组大小) as string  ‘重新声明
```

现在，我们来设计数据过滤方案：

数据过滤在介绍数据采集过程时有了初步的介绍，总结一下，我们分几个级别来进行过滤：

第一级别，不区分文件和目录，我们过滤掉所有包含有特定字眼的信息，如黄色、色情等；过滤掉那些长度超过数据库中定义的相关字段最大长度的信息。

第二级别，过滤某些目录，比如 bin/，Program Files/，新建文件夹/等，其中包含的信息，往往并不是人们需要的。

过滤掉某些文件，其中有一个办法就是通过后缀过滤掉好多文件，另外，也可以通过文件名来过滤，比如 readme，新建 Microsoft Word 文档，新建 文本文档等，他们通常也是一无所用的。

第三级别，前两个级别都是在数据过滤阶段进行，但是这个级别是数据入库之前。这里也是为了过滤掉某些文件，但是这些文件有一些特别：只是它们的命名没有任何意义。比如纯数字命名，但是它们的目录往往是非常有用的，所以它们自身被过滤掉，而目录保留下来。

到目前为止，数据过滤就差不多拉，完全可以通过 ini 文件来保存某些要过滤的文件，或者需要保留的文件，从而使得程序具有很好的扩充性。

现在介绍一下 Microsoft internet transfer 控件登录 FTP 站点时涉及的一些操作：

我们先通过添加 component 来导入该组件，并添加到窗体中，命名为 inet1

1）,数据采集过程部分

A，在命令发送之前，我们得进行一些初始化操作。

```
Inet1.URL   ‘ftp 站点的 Url 地址
Inet1.RemotePort   ‘ftp 站点开放的端口号
Inet1.UserName  ‘登录用户名
Inet1.Password   ‘登录密码
```

而在窗体初始化的时候，初始化下面两个属性

```
Inet1.Protocol = icFTP  ‘使用的网络协议
```

Inet1.AccessType = icUseDefault ‘是否直接连接，这里选择默认，即使用当前 IE 的设置。

现在介绍命令发送：

Inet1.execute ，“cd ” & “目录”

Inet1.execute ,“dir ”

我们可以通过该控件的 execute 方执行 cd 和 dir 命令来实现登陆 FTP 以及数据遍历操作。

B，控件状态接收端，我们可以在该控件的状态改变事件触发过程中判断状态，并在适当时候接收数据和处理数据。

```
Private Sub Inet1_StateChanged(ByVal State As Integer)
Dim filelistTmp as string,filelistArray() as string
If State = 12 Then
    filelistTmp = Inet1.GetChunk(0) ‘这里 filelistTmp 用来接收文件列表信息
    fileArray = Split(filelistTmp, vbCrLf) ‘由于接收到的数据是用换行符分割着的，这样我们用分割函数就可以取得每个文件名
     ‘这里进行进一步的处理，包括数据过滤数据以及存入临时变量等操作
end if
end sub
```

2),数据过滤

我们采取的三种方案都是没有经过仔细的统计分析后做出的，所以可能存在这样的问题：即剔除了某些重要的文件信息。但是权衡利弊后，我们发现：它对数据库的信息的压缩和优化来说起到了非常重要的作用。因此，目前需要改进的是在对大量站点信息进行统计分析以及对其他一些相关工作（比如“命名学“方面的知识等）进行学习和研究后得出更科学的过滤方案。但是过滤措施的提出似乎是一个“创举”。我们可以从另外一个角度减少建设搜索引擎站点的成本。

至于具体的实现可以查看源代码，其实只进行了比较简单的比较和判断操作。具体实现时把上面的几个不同级别的过滤方法写到一个过程里拉。目的是为了日后程序的维护。

3.3 数据入库

这里主要是把经过过滤和优化后的数据，即把临时变量中的数据保存到数据库中。这里

同样涉及连接数据库和 sql 语句的执行，具体同 3.1。但是操作是比较方便的，原因是临时变量的结构和数据库中表的结构非常类似。

它主要涉及到对原有站点记录的删除操作，可以通过 myrs.delete 和 myrs.update 来实现；另外还涉及数据的追加操作，可以通过 myrs.addnew 和 myrs.update 来实现。

最后这里有个需要注意的问题：就是在执行记录删除操作后，库的大小并不会因此而变小，而需要你进行数据库压缩后才会真正释放掉空间，因此我们在每次执行删除记录操作后专门通过 DBEngine.CompactDatabase 方法来实现。

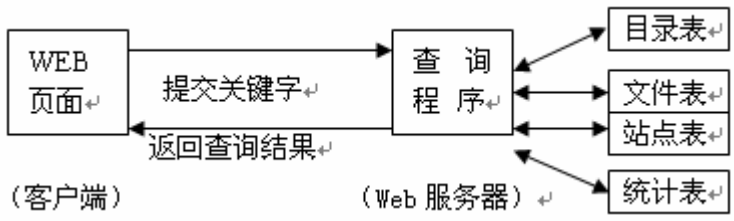
至此数据采集过程分析完毕，具体代码参见源程序。

4. 数据查询

通过数据采集我们获得了查询需要的数据，下面将具体分析和实现数据查询模块。

数据查询主要包括查询页面的设计、查询程序的编写和查询结果的返回以及关键字统计等。查询页面由 Web 服务器提供，用于收集所要查找的文件信息，用户浏览到此 Web 页面，填写并提交表单。为减少 Web 服务器的工作量，提交时由客户端对关键字进行过滤。表单中需包含文件名信息，另外为方便用户查找特定类型的文件，设置一个下拉菜单供用户选择文件类型。表单提交给 Web 服务器之后，由查询程序进行分析，生成查询语句并执行查询操作，查询结果由查询程序进行统计分析，并按搜索次数排序、以分页的方式返回给用户浏览。查询结果主要包含文件在第三方服务器上的符号链接地址，以及相关关键字等信息。另外，查询结果的统计信息保存到关键字统计表中。与此同时，还可以据此更新目录表和文件表中的搜索次数字段。

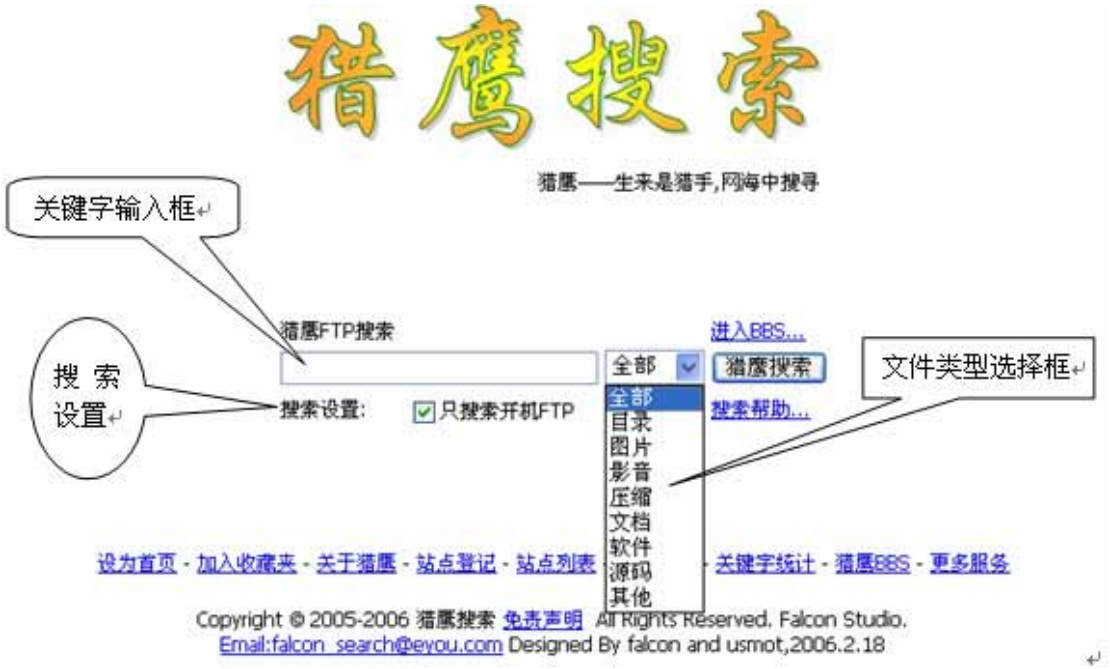
数据查询的具体过程图示如下：



下面详细分析和设计 WEB 提交页面和查询程序。

4.1 WEB 提交页面的设计

经过对常用搜索引擎的分析，发现提交页面都非常简单实用，我们的做法也是这样。下面是我们的提交页面：



通过上面的图片，我们看到提交内容的地方只有三个：关键字输入文本框、文件类型选择下拉框、以及搜索设置复选框。我们就这三个表单元素的作用进行分析。

A,关键字输入框：这里是用户输入搜索关键字的地方，用户可以输入单个或者多个关键字，另外可以进行模式匹配操作，例如，可以输入“mp3:童年”来搜索文件名为童年的 mp3。

B,文件类型选择框： 通过这里用户可以缩小搜索范围，从而找到更满足自己要求的文件。这里需要详细解释一下具体的分类情况。

全部：包括所有的目录和文件，需搜索目录表和文件表

目录：只包括目录，仅搜索目录表

一下只搜索文件表，根据文件后缀分类(这里的分类是可以扩展)

图片：jpg,gif,bmp,ico

影音：mp3,rm,wav,wma,mid,wmv,rmvb,mpg,swf,avi,asf

压缩：rar,zip,iso,tar,gz,tgz,tbz,bz2

文档：txt,doc,htm,html,pdf,ppt,chm,pdg

软件：exe,rpm,bin

源码：c,cpp,java,asm

其他：dll,w3x,conf 等

C,搜索设置：为了保证搜索结果是有效的，所以设置该项非常有用。它可以尽量保证结果的有效性。

表单的内容设置好以后，就涉及到提交过程拉，提交过程是这样的：

用户单击提交按钮或者按下{Enter}键，触发提交事件，并使网页转向表单的 action 属性所指向的查询程序所在页面。

为了减轻 WEB 服务器的一些负担，我们在客户端通过 javascript 或者 vbscript 来对用户输入的关键字进行一定的过滤。比如，过滤掉空串、不雅、以及常见字或者词语。

4.2 查询程序

通过数据查询过程图示可以看出，“查询程序”是整个“数据查询”模块的核心。而查询程序本身包括几个大的方面：接收提交页面提交的信息；将接收的信息进行区分和处理并转换成查询语句；连接数据库并执行查询语句进行查询操作；对查询结果进行分页处理和统计分析；将结果返回给用户。

为了能够比较透明的描述查询程序，先看看查询结果页面。

可以看出，这里具体包括查询提交模块、用户搜索的设置以及搜索结果总数、页数，查询过程耗费的时间以及返回给用户的文件信息，另外还有相关搜索的显示等。

由于涉及到的内容比较多，我们这里只简单介绍几个最关键的地方，详细情况参看源代码：

第一部分，时间统计部分，该部分其实只需要在连接数据库之前先设置一个时间变量，记为 SearchStartTime 用系统时钟 Timer 赋值，然后在执行完查询语句后用当前的时钟减去 SearchStartTime 就可以拉。

第二部分，从关键字的接收、查询语句的生成、连接数据库并执行查询语句、通过分页返回结果、关键字的统计几个方面来介绍。

1）关键字的接收有两种方法：

如果查询提交页面中的表单方法属性设成 get 即 method=get。假如我们要接收关键字信息，并设提交页面的相应表单元素名为 key,那么可以在结果返回页面（我们命名为 search.asp）中用 request.querystring(“key”).value 来取得。

如果查询提交页面中的表单方法属性设成 post 即 method=post，那么可以在结果返回页面中用 request.form(“key”).value 来取得。

2）关键字的处理与查询语句的生成

A,多关键字的处理

我们通过 asp 中的一个分割函数把包含空格的关键字拆分成多个,然后生成查询语句中条件。

B,根据不同的分类来查找不同的表或者不同的字段

之前我们得接收“文件类型”参数，并且定义一个数组，分别存放不同“文件类型”参数对应的文件后缀。然后根据“文件类型”参数的不同取值并结合上面得到的查询条件来生成不同的查询语句。

当文件类型为“全部”时，需要联合查询目录表和文件表。用到 SQL 查询中的联合查询技术。

另外，在查询文件表中不同的类时，还需要用 in 来限定查询的文件后缀的范围。

C,根据接收的“搜索设置”的值来决定是否访问站点表，来获取当前开机的 ftp 站点编号来限定访问目录表和文件表中对应编号，从而加强上面得到的查询语句的条件。

到这里查询语句就可以生成拉。

另外，上面提到了模式匹配。这里可以通过分析关键字中是否包含“：”解析一个“文件类型”值，从而和上面的根据“文件类型”生成查询语句一样进行处理。

3）连接数据库和执行查询语句

在 asp 中一个比较好的连接数据库的方法是 DAO 连接：

Set conn = Server.CreateObject("ADODB.Connection")

conn.Open "driver={Microsoft Access Driver (*.mdb)};dbq="+Server.MapPath("库名")

driver 是驱动器参数，而 dbq 是数据库所在的绝对路径，通过 Server 对象的 mappath 以及当前路径就可以获得。

至于执行查询语句，就这么简单

set rs = Server.CreateObject("ADODB.Recordset")

rs.open 查询语句（这里是我们在上面生成的查询语句）

4)数据返回和分页显示

这里涉及到查询的效率问题，通过总结分析 Asp 优化的方法以及具体实现，发现先用 set 方法把记录放到内存变量中能有效提高搜索的效率，如:set id=rs(id 字段的编号)

然后直接可以在后面通过使用 id 的值来使用该记录信息。

而分页的设置也有好多措施，但是原理大都一样，我们只介绍具体使用到的分页技术中的一些主要参数：

rs.PageSize 设置每页的记录数

rs.PageCount 总的页数

rs.CacheSize 设置页缓冲区

5）相关搜索显示以及统计分析

相关搜索是通过访问统计表，查找出与当前查询的关键字近似的关键字信息。

而统计分析是通过当前查询的关键字来更新关键字统计表。

6）实践过程中引入的一个比较重要的技术

在返回的结果中，提供给用户的连接并不是某个文件或者目录的 url 地址，而是它在数据库中的索引信息，即文件的父目录编号和所属的站点编号。而这些信息全部存放在目录表中。这样做有两个非常大的好处：能够很好的提高搜索效率，而且减少服务器的工作量。

如果返回的结果中就是文件的 url 地址，那么获取 url 地址的过程将耗费很多时间，因为对每个文件都要执行该操作。而且取得每个文件的 url 地址的过程需要访问数据库的次数都非常大。

而做了上面的处理后，我们可以节省获取 url 地址的时间，从而可以做大更快把文件信息返回给用户。与此同时，用户如果想下载某个文件，同样可以通过点击连接按钮打开一个获取 url 地址的页面，而这个页面包含地址重定向功能，在访问数据库后找到 url 地址并引导用户打开该地址。

可想而知，用户的最大操作可能是打开所有的文件。这样的情况数据库的访问量才和一次性给出所有文件地址的操作量相当，但大多数情况下，用户只选择其中的某些结果。因此，该改进对可以明显降低对数据库的访问次数，减少 WEB 服务器的负荷。

5. 管理与维护

通过数据采集和数据查询模块的分析和设计，我们基本实现了一个 FTP 搜索引擎，但是还需要相关的数据更新和维护。

这里的数据更新包括“站点可连接性的测试”,校园网内新开通的 FTP 站点的登记等，我们这里是通过 winsock 控件来实现的。只需要通过它来登录 ftp 站点就可以拉。

另外，站点采集最好是做成自动化的，从而较少人为的操作负担。

这样一个相对完善的 FTP 搜索系统就已经完成拉。

6. 其他功能

通过对其他 ftp 搜索引擎的分析发现,大都有站点快照功能,经过仔细的分析和设计后，终于完成基于原有的数据库就很快实现拉。

站点快照功能有个好处,可以引导用户快速的浏览某个 ftp 站点下的文件并进行下载等操作。

这里有两种设计方案：

第一种，“天网”的树形单页结构，这里把文件系统的结构完美的体现出来拉。而且浏览起来很方便。

第二种，“星空”的表形多页结构，这里其实只是对表结构的一个简单“翻译”，而且它同我们平时访问 ftp 服务器见到的界面差不多，只是浏览的数据很能有很大的提高，而且页面排布可以更美观些。

由于时间等方面因素，暂时本搜索引擎只实现第二种方案。在时间允许的情况将实现第二种方案。

结束语

我们设计的 FTP 搜索引擎已经在网站上运行了两个多月,为校园网用户提供了很好的服务。访问量到现在已经超过 1 万次。最近和 usmot 共同推出了最新版本——“猎鹰搜索”。对数据采集和数据查询、数据的更新维护、以及 WEB 页面的设计方面都做了优化，效果比较好，当然也存在一些不足。随着搜索引擎技术的不断发展，我们将继续秉着为校园网添彩、为校园网用户服务的热情继续努力，推出更好的 FTP 搜索服务。另外，我们打算把该项目转到 linux 下，作为开源项目来开发，具体打算采用 php+c+mysql 模式。我们期待着会有新的收获。

参考文献：

1，张运凯 刘宏忠 郭宏刚 《FTP 搜索引擎的设计与实现》

2，（美国）MICROSOFT 公司 《Visual Basic 6.0 控件参考手册》

3，《用asp制作强大的搜索引擎》 http://www.hoogl.net/20051126204241/

4，《SQL参考手册》http://www.poptool.net/quickcheck/sql/

5，《ASP基础教程》
http://www.ahtvu.ah.cn/jxc1/wshdsh/webstyle/computer/asp%BB%F9%B4%A1%BD%CC%B3%CC%B5%E7%D7%D3%CA%E9/homepage.htm

6，《ASP 网页的优化》http://www.linkwww.com/article/list.asp?id=16

7，歪酷搜索页面风格 http://www.ycul.org/

其他说明：

1， 该 FTP 搜索引擎目前的测试地址：falcon.96.cn

2， 本人联系方式：qq:253087664 Email:falcon_search@eyou.com

3， 关于详细源代码的下载问题，暂时不提供下载，以后可以到“兰大 ftp 联盟 QQ 群:18301157”查看详细源代码的发布地址”。申请加入时请输入：“兰大 ftp 联盟”字样，以表明自己身份，谢谢。

4， 由于打算到 Linux 下做 ftp 搜索引擎的开源项目，大家如果有兴趣，可以联系本人，或者加入上面的群，也可以到 falcon.96.cn 下的 BBS 留言。

5， 由于 falcn.96.cn 为申请的二级域名，可能不稳定，以后也许访问不了，那么请原谅。

2006.3.9

falcon 于兰州大学