# 基于 XML 文档检索的搜索引擎设计

谭新良　蔡代纯

（湖南城市学院计算机科学系　益阳 413049）（湖南城市学院图书馆　益阳 413049）

**摘　要**　设计了面向 XML 文档检索的搜索引擎模型，该模型包括机器人模块、转换模块、解析模块、索引模块和查询模块这五个部分。转换模块和解析模块是专门设计的。介绍了模型的设计思想及框架，详细描述了各模块的结构和实现思想。

**关键词**　XML，检索，搜索引擎

## Design of Search Engine Based on XML Document Retrieval

TAN Xin-Liang　CAI Dai-Chun

(Department of Computer and Science, Hunan City University, Yiyang 413049)　(Library of Hunan City University, Yiyang 413049)

**Abstract**　In this paper, a new search engine model is studied and designed based on XML document retrieval. It comprises robot module, switch module, parse module, index module and query module. The switch module and parse module are specially designed. The paper introduces the design thought and frame. A detailed description of structure and implementation thinking is given.

**Keywords**　XML, Retrieval, Search engine

## 1　引言

随着信息高速公路的建立，Internet 的普及，多媒体、网络技术的迅猛发展，人类社会已经进入了一个信息化社会。当今，互联网已成为人类有史以来资源最多、品种最全、规模最大的信息库。作为网上最主要的信息检索工具，搜索引擎发挥着非常重要的作用。传统的搜索引擎大都是基于 HTML 的搜索引擎，HTML 重显示而非内容的特点大大限制了搜索引擎的查准率，传统搜索引擎的查准率亟待提高。现在，一种可扩展标记语言 XML 开始慢慢发展起来，越来越多的文档开始用 XML 语言来描述、存储和交换。XML 的标记含义丰富、意义明确，能明确地提示所标记的内容，搜索引擎可以依靠标记和内容之间的依存关系，准确定位、找到目标，从而大大减小搜索范围，提高检索精度。为此，本文对面向 XML 文档进行检索的搜索引擎进行了研究，并设计了一个搜索引擎模型。

## 2　设计思想及模型

面向 XML 的搜索引擎仍然包括普通搜索引擎的几大模块，即采集、索引和查询这几大模块。另外，针对 XML 文档的特点，还专门设计了转换模块和解析模块。本搜索引擎模型中的采集机制即机器人模块，它从 Internet 上搜集 HTML 文档和 XML 文档，将搜集到的文档传递给转换模块，统一转换为 XML 文档后再由索引模块为其建立索引，以方便用户查询。

目前在 Internet 上只有部分网页是由 XML 编写的，因而本搜索引擎并不规定数据源的格式，机器人不仅下载 XML 文档，还下载 HTML 文档。当机器人下载了各类文档后，需要将其转换成 XML 文档，同时为没有文档类型定义（DTD）的 XML 文档生成 DTD，即将无效但格式良好的 XML 文档转换成有效的 XML 文档。完成这两项工作的部分称为转换模块。

XML 文档分为文本信息和结构信息，它是严格的树状结构，各个标签之间有严格的父子、兄弟关系，标签之间是文本信息。标签中的内容用来标明夹在起始标签与结束标签间的数据的性质。建立索引之前需要提取 XML 文档的文本信息和结构信息，这个工作由解析模块来完成，所以在转换模块与索引模块之间应该有个解析模块，转换模块将转换后的文档传给解析模块，解析模块解析 XML 文档的文本信息与结构信息。

搜索引擎为了加快对用户检索要求的响应速度，需要给采集到的数据建立索引，XML 文档被解析后，就由索引模块对 XML 文档的文本信息和结构信息建立索引记录，并将记录存入索引数据库中。索引模块相当于普通搜索引擎的数据组织与索引机制。

此外，还应有个查询模块，对应着普通搜索引擎的用户检索机制。查询模块即用户与搜索引擎的接口部分，它需要引导用户输入其查询请求，并将用户输入的查询请求翻译为搜索引擎能读懂的格式，即针对 XML 的查询语言，然后查询索引数据库，最后将结果排序后显示给用户。

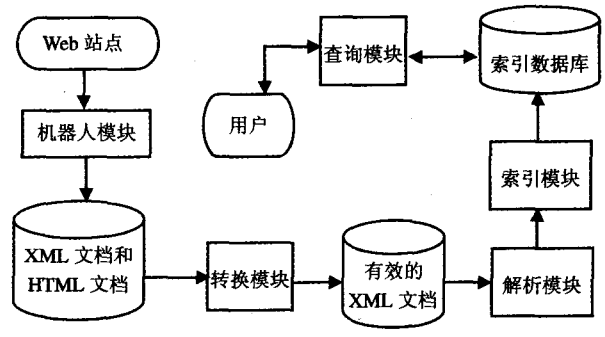


图 1　面向 XML 文档进行检索的搜索引擎模型

根据以上分析，本搜索引擎模型分为五大模块：即机器人模块、转换模块、解析模块、索引模块与查询模块。转换模块、

**谭新良**　硕士，讲师，主要研究方向：数据库应用技术；**蔡代纯**　馆员，主要研究方向：信息检索技术。

解析模块是专门设计的。面向 XML 文档进行检索的搜索引擎模型如图 1 所示。

## 3 各模块的具体实现

### 3.1 机器人模块

机器人模块在 Internet 上一刻不停地漫游，其作用是抓取超文本网页资源，同时定期浏览已存储在自己数据库中的网页，以避免网页过期导致的无效链接[1]。

机器人和 Web 站点的 Web 服务器通过 HTTP 协议进行交互，从 Web 站点下载 XML 文档和 HTML 文档。它应该分为检索和过滤二个部分。

(1)检索部分

显然，机器人的检索部分应该实现其抓取页面的功能和检查更新其已抓取过的页面。因比较复杂，本文只描述搜索引擎进行第一次单一遍历时的过程。信息收集方法一般首先通过搜索判断当前 Web 上有哪些站点正在运行，创建 Web 上完整的网站列表，然后根据站点的列表由机器人获取每个站点上所有可以访问的文档。此过程的大致思想是：先由一个种子网站开始，这个网站一般是比较流行的网站，依次分析该网站中的网页，提取出该网页的链接，分析这些链接，进行抓取和存盘工作。首先设置四个队列，它们分别为：W：待抓取站内页面的 URL 队列；S：待抓取网站的 URL 队列；A：被抛弃页面 URL 队列；N：已抓取页面 URL 队列。机器人抓取文档的步骤大致如下：

①搜集一些网站地址放入 S 中，另三个队列置空。

②从 S 中的网站队列中取出一个 URL 将其放入 W 中。

③从 W 中取出首个待抓取页面的 URL，判断是否满足抓取条件。判断时要用到过滤部分，下文将进行介绍。若该页面符合抓取要求，则将该页面的 URL 放入 N 中，并将该文档传给转换器。然后依次分析该页面的每个链接，若链接为外部链接且符合采集范围，则判断该链接的 URL 地址是否已存在于 S 中，如不存在，将其添加至 S 队列中；若该链接目标为本站点的页面且未被抓取，则判断该链接的 URL 地址是否已存在于 W 中，如不存在，则将其添加至 W 尾部。若该页面不符合抓取要求，将其抛弃并将其对应的 URL 放入 A 中。若 W 不空，则重复进行此步骤。

④若 W 空但 S 不空，转步骤(2)。

⑤若 W 空 S 空，结束操作。

(2)过滤部分

并非所有的文档内容都是可以被抓取的，这些文档往往会涉及到作者的隐私或版权问题，如果不加以限制，就会发生诸如侵权或是服务器过载等问题。通过过滤，可以防止机器人运行带来的大量社会问题。为此，搜索引擎必须遵循 SREX (Standard for Robot Exclusion) 协议或者是类似的防止搜索引擎侵犯网站的协议(如：协议规定每个搜索引擎都应该向站点提供标示、搜索目的、搜索引擎拥有者的联系方式等内容)[2]。

### 3.2 转换模块

机器人模块下载的文档包括 HTML 文档和 XML 文档，而索引器要为 XML 文档建立索引，必须把 HTML 文档转换为 XML 文档，这就需要有个转换器。另外，规范的 XML 文档都必须有良好的格式，但其数据结构可以被规定，也可以不被规定。如果被规定，那么该文档是有效的，否则称为无效的。有效的 XML 文档必须带有 DTD 或 XML Schema，即带有一段关于该文档中数据的组织和存放结构的说明，它不但严格定义了某项数据应该在哪出现，而且规定了各种数据之间的关系，即哪个标记可以包含其它标记、标记的排列顺序以及可包含的数据标记类型。

网页中的许多 XML 文档虽然是规范的，却不是有效的，本搜索引擎模型的解析模块在解析时需要为文档的各个元素编码，这要用到 DTD，所以转换模块还应包含一个 DTD 生成器，其基本结构如图 2 所示。

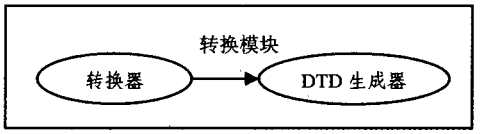


图 2 转换模块结构图

要实现 HTML 数据向 XML 的转换，关键是给出 HTML 的内容数据及其关系的一种组织方式，找出这种方式在 XML 模式中相应的表达规则，建立 HTML 标识到 XML 模式的一种映射，从而实现从 HTML 内容到 XML 结构的转换[3]。

HTML 中既有用来表现特定意义的内容数据，又有用来呈现数据格式的数据，要把 HTML 自动有效地转换成 XML 数据，主要问题是实现内容数据和格式信息的分离，这可以分为对网页中 HTML 标示符的识别和文本的自动分词两步来进行。HTML 中的标示符都是以“〈”开始，后跟控制命令和各种参数，最后以字符“〉”结束。

在实际处理时，需要根据标示符命令的实际语义，把标示符命令分为两类：一类是在语义上不起分隔作用的标示符命令，另一类是起分隔作用的标示符命令。前一类标示符命令包括：〈A〉，〈B〉，〈I〉，〈EM〉，〈T2〉，〈BIG〉，〈SUB〉，〈SUP〉，〈FWT〉，〈SMALL〉，〈STRONG〉，〈STRIKE〉等以及它们对应的结束符标示符命令。这类标示符命令在语义上不起分隔作用，两字符之间若出现这样的标示符命令，仍应认为是两个连续的字符。起分隔作用的标示符命令包括除第一类标示符命令以外的绝大多数。被这类标示符命令隔开的两个字符，应该被视为不连续的。根据标示符命令的以上特点，在提取 HTML 文档中的信息时，遇到第一类标示符命令时，只要把标示符命令去掉即可，遇到第二类标示符命令时则要用空格代替它。

HTML 文件中放置内容数据的标示符主要有：页面的标头(head)、段落(p)、图像(img)、表单(form)、表格(table)以及多页面(frame)等。HTML 文件中的内容数据主要表现为文本、图形链接以及文本链接等。表单、表格以及多窗口页面给出了这些原子内容数据的组织形式。表格的语法形式是〈table〉…〈/table〉，〈tr〉用来定义表行，〈th〉用来定义表头，〈td〉用来定义表元。表头中原子数据信息都在〈th〉和〈/th〉之间，表格的具体数据在〈td〉和〈/td〉之间，根据这些规则可以界定表格中的内容数据。

### 3.3 解析模块

解析模块是面向 XML 文档进行检索的搜索引擎与普通搜索引擎不同的结构之一，是针对 XML 文档的特点所设计的模块，也是非常重要的一个模块。XML 文档相比于 HTML 的特点是重数据的内容和结构，而非数据的显示，它使面向 XML 文档的搜索引擎的搜索有了针对性，故而大大提高了其查准率。解析模块就是体现这一特点的一个重要部分。正如前面所分析的，解析模块的主要任务是提取 XML 文档的内容信息和结构信息，它由模块中的解析器来完成。解析器负责对输入的 XML 文档进行解析，生成 DOM(文档对象模型)树。

DOM 树生成后，应该对树的每个节点做个标记，这个标记在它所处的文档中是唯一的，即一个标记唯一标识一个节

点，这可以方便建立索引。本文把用来完成这个任务的部分命名为节点编码器。

由上所述，解析模块包括解析器和节点编码器，结构如图3所示。

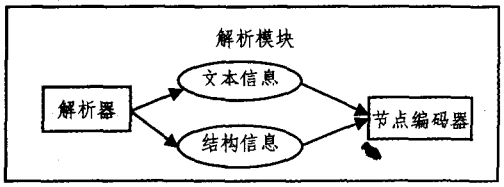


图3 解析模块结构图

### 3.4 索引模块

索引的组织方式对于搜索引擎的检索效率起着关键作用，基于XML文档进行检索的搜索引擎也不例外，索引器就是用来为文档建立索引的。对XML文档的索引既要对标签标注的内容建立索引，又要对标签本身建立索引，所以XML文档的索引文件比HTML文档的索引文件要大。

基于XML文档检索的搜索引擎属于基于内容的搜索引擎，而基于HTML文档检索的搜索引擎是基于文本的。前者不但要考虑关键词的匹配，而且还要兼顾语义上是否一致，后者则简单得多，只要考虑字符的匹配与否就可以了。所以，为XML文档建立索引前，需要将每个节点的结构弄清楚，即该节点的标签名、编码、父亲编码、属性值等，所以本系统设计的索引模块中有一个节点结构构造器，用来将已解析好的文档树中各节点结构列清楚，以方便索引器建立索引。

索引模块的具体结构如图4所示。

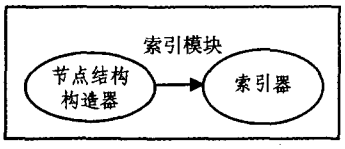


图4 索引模块结构图

节点结构构造器是为索引器做准备工作的。XML文档分结构信息和内容信息，这些都要编入索引，把XML文档看成一棵树，树中的节点作为一个基本的存储单元，每个节点有一个唯一的标识符，这个标识符是由节点编码器分配的，本文将其简记为Id，其标识符为一个编码，形式为(start，end)。

把每个节点看成一个记录存储在索引数据库中，叶子节点和中间节点结构稍有不同，中间节点的节点结构包括如下部分：

①Id：元素节点标识符；②Tag：元素标记；③Attrs：元素属性及属性值；④Parent：父元素节点Id，即父元素节点标识符；⑤Previous：下一个兄弟元素节点Id，即右兄弟元素节点标识符。

叶子节点的节点结构包括如下部分：

①Id—元素节点标识符；②Parent：父元素节点标识符；③Text：文本内容。

XML文档经过节点编码器后生成DOM树，再经过节点结构构造器后产生的节点结构仍是一颗树，树的最下一层是叶子节点，即文本部分，其余都是中间节点，即结构部分。

### 3.5 查询模块

查询模块是搜索引擎与用户的接口部分，它为用户提供一个界面，即用户界面，它应包含两个功能。首先是引导用户输入查询信息，功能大致与其它搜索引擎的用户界面相似，其次，由于XML文档是树状结构，本身比较复杂，用户界面还应该起到一个导航的作用，逐级地提示用户。此外，查询模块还应包括一个解释器，主要负责把用户的输入翻译成查询执行器能够读懂的语法格式(包括逻辑表达式)，即针对XML的查询语言，然后由查询执行器搜索索引数据库，最后将结果排序后返回给用户。

由此可见，查询模块包括用户界面、解释器和查询执行器，其结构如图5所示。

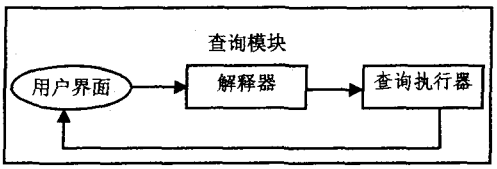


图5 查询模块结构图

在进行检索时，首先用户从用户界面输入查询请求，用户输入查询请求后，将查询请求先传给解释器，解释器主要功能是对用户以各种形式(取决于查询界面)输入的查询条件进行预处理，然后送交给查询执行器。如果输入的是一句话，则由解释器将其中对查询不起作用或作用很小的词去除，比如“Of”，“is”之类的词。所以解释器还要起到过滤的作用，解释器中存储着所有这类词汇，查询请求里面如果有和这类词汇中相同的词，就将其过滤掉。

查询执行器接收到用户输入的关键词后，将关键词与关键词索引表中的词进行对比，这需要将该查询转化为一定的查询语句，在关键词索引表中进行查询，若找不到，则返回查询失败信息；若找到该词，则从该词所对应的表中的记录信息中提取其文档编码与父节点编码。这又分为两种情况：一种是用户没有输入关键词类别，则查询文档结构表，依次查找与提取出来的文档编码所对应的文档，根据父节点编码在查找到的文档结构记录中找到该关键词的父节点。一般父节点的内容是描述该关键词的，提取出其信息。这样提取出来的父节点可能会有成千上万条记录，但有很多标签实际上表现同一个实质，因而能分类，可以将意义相同或相近的词归到一起，即将其归类返回给用户，在用户界面上以选择的形式让用户表明自己所想要的类别，再将用户筛选后的结果返回给查询执行器。

另一种是用户输入了关键词的类别，查询关键词索引表，找到包含有该关键词的文档及该文档中关键词的父节点，将用户输入的关键词类别与查找到的文档中的关键词的父节点相比较，如果相同，返回该文档的地址；如果不同，则查询文档结构表，继续向上比较父节点的父节点，如此反复，直到根节点。用户输入的关键词有时与数据库中存储的关键词也许并不相同，但意思相近或本质相同，可返回，询问用户是否查询，从而根据用户的要求返回文档或抛弃。

**结束语** 本文对基于XML文档检索的搜索引擎进行了研究，并设计了一个搜索引擎模型。介绍了模型的思想及框架。该模型包括机器人模块、转换模块、解析模块、索引模块和查询模块五个部分。转换模块和解析模块是面向XML文档检索的搜索引擎与传统搜索引擎区别最大的地方。对各模块进行了详细介绍，包括它的基本结构和设计思想。

## 参考文献

1 王海波，姜吉发. XML搜索引擎研究. 计算机应用研究，2001，18(4)：68～71

2 Guidelines for Robot Writers[EB/OL]. http://info.webcrawler.com/mak/projects/robots/robots.html(Accessed Jul. 25，2006)

3 张文斌，陈恩红，王进. 一种基于多叉树的HTML到XML的转换方法. 小型微型机算机系统，2003，24(4)：713～715