

## 基于内容的多媒体数据递进目标搜索

贾长云<sup>1</sup>,程永上<sup>2</sup>,朱敏<sup>1</sup>

(1. 淮海工学院 计算机工程学院, 江苏 连云港 222005; 2. 南京财经大学 信息工程学院, 南京 210046)

(lyghhitjey@vip.sina.com)

**摘要:** 为了有效提高移动终端多媒体信息的能力, 讨论了一种新的多媒体信息查询方法——基于内容的递进目标搜索, 提出了“查询流”的技术, 通过多媒体查询方法(MQF)与XML片断请求单元及片断更新单元的结合, 使得用户对多媒体信息的查询逐步进行, 先查询相关的元数据描述, 然后查询最终结果。这样有效降低了查询的数据通信量, 非常适合于配置较低、信道有限的移动终端实现多媒体信息的查询。

**关键词:** 多媒体; 目标搜索; 多媒体内容描述接口; 逆波兰表示法

**中图分类号:** TP311 **文献标志码:** A

## Progressive goal searching of multimedia data based on content

JIA Chang-yun<sup>1</sup>, CHENG Yong-shang<sup>2</sup>, ZHU Min<sup>1</sup>

(1. School of Computer Engineering, HuaiHai Institute of Technology, Lianyungang Jiangsu 222005, China;

2. School of Information Engineering, Nanjing University of Finance and Economics, Nanjing Jiangsu 210046, China)

**Abstract:** In order to improve the querying ability of the multimedia on the mobile terminal, a new method about querying the multimedia, namely, the progressive goal searching based on contents was presented, and the technique of the querying stream was provided. Combining Multimedia Query Format (MQF) of media querying method with the unit of XML requesting snippet and the unit of the updating snippet can realize the querying of the media information by users step by step. First, the description of the correlative meta data was queried, then, the result was queried. Therefore, this method can reduce the querying data transfer and it is suitable for the low configuration and the limited channel mobile terminal to achieve the media information query.

**Key words:** multimedia; goal searching; MPEG-7; Reverse Polish Notation (RPN)

## 0 引言

随着网络上多媒体信息的日益增长, 针对多媒体的信息查询越来越受到关注。这一点在由普通用户可以产生或共享的多媒体网站(如YouTube或Flickr等)表现得尤为突出。但是要共享这些多媒体数据并非轻而易举, 因为需要在多媒体数据上附加元数据以对多媒体数据进行描述。目前主要有两种描述多媒体数据的标准: 级别较低的多媒体内容描述接口(MPEG-7)<sup>[1]</sup>和级别较高的Dublin Core<sup>[2]</sup>(都柏林核心元数据, 主要用于描述电子资源)。

近年来, 对多媒体数据库的探索在各种文献中受到了广泛的关注, 对多媒体信息查询的表达能力如内容信息表达能力、时空信息表达能力等也进行了研究<sup>[3]</sup>。本文的焦点在于探讨多媒体数据库(如IBM的MARVEL<sup>[4]</sup>、Google的Google Image<sup>[5]</sup>等)与客户端之间进行数据通信的高效途径。一般而言, 多媒体数据容量非常庞大, 如果使用普通的基于内容的多媒体查询方法, 返回到客户端的数据量显然很大, 这一方面影响了数据的传输速率, 另一方面更重要的是在无法容纳如此量数据的客户端(如屏幕、内存都很小的移动设备)上根本就无法显示相关的多媒体信息。本文以递进目标搜索作为一种可能的解决方案来实现多媒体数据的递进查询, 以适应小容量客户端的查询需要。

## 1 递进目标搜索

递进目标搜索是一种特别适合于小容量客户端实现的基

于内容查询多媒体信息的方法, 它不是返回用户所希望的完整的结果集, 而是先返回一系列与结果集相关的元数据描述信息, 然后引导用户在这些元数据描述中不断地选择最终的结果集。它仍然属于基于内容的多媒体信息查询<sup>[6-7]</sup>, 但这种方法可以有效地减小返回到客户端最终结果集的大小。

实现多媒体数据查询是建立在具有存放多媒体信息的多媒体数据库和能够在数据库与客户端进行通信的协议这两个基础之上的, 而要实现递进目标搜索这两个基础还必须满足一些附加条件。

从数据库来说:

1) 无论是采用人工的或是自动的手段, 数据库中的多媒体数据都必须带有相应的元数据描述标签;

2) 描述标签必须与被描述的项目密切相关。

从协议来说:

1) 用于客户端与服务器之间的数据交换协议必须能够允许任何格式的查询(如文本的或多媒体的);

2) 必须可以对查询命令的局部进行更新, 以便能在先前的结果集中进行递进检索;

3) 查询命令的格式也必须是足够弹性的, 以容纳不同类型的内容, 同时还能将元数据完整地传回到客户端以便在客户端, 以最合适的方式展现这些元数据。

已经成为国际标准的简单对象访问协议(Simple Object Access Protocol, SOAP)就是能够实现数据传输的可伸缩基本协议, 而参考文献[8]中的基于XML的多媒体查询方法

收稿日期: 2009-09-09; 修回日期: 2009-10-29。

**作者简介:** 贾长云(1960-), 男, 江苏涟水人, 副教授, 硕士, 主要研究方向: 数据库、软件工程; 程永上(1971-), 男, 江苏连云港人, 副教授, 博士, 主要研究方向: 分布式计算、Web服务组合; 朱敏(1970-), 男, 江苏东海人, 讲师, 硕士研究生, 主要研究方向: 数据库、软件工程。

(Multimedia Query Format, MQF)可以认为是 SOAP 协议的堆叠,本文即是以此为基础研究实现递进目标搜索的方法。递进目标搜索的基本原理如图 1 所示。

1) 用户向服务器发送一个带有初始关键字的查询;

2) 服务器在数据库中检索满足关键字的数据形成结果集,初始查询仍保留在服务器中;

3) 服务器向客户端返回带有元数据描述的结果集;

4) 客户端随后再向服务器发送所需要结果集的元数据,到现在为止客户端并收到任何实际的结果集;

5) 服务器更新初始查询,重新执行步骤 2),但只是在已形成的结果集中处理,并不需要重新访问数据库;

6) 客户端根据需要重复步骤 4),直到找到满意的结果集;

7) 服务器将最终结果集发送到客户端。

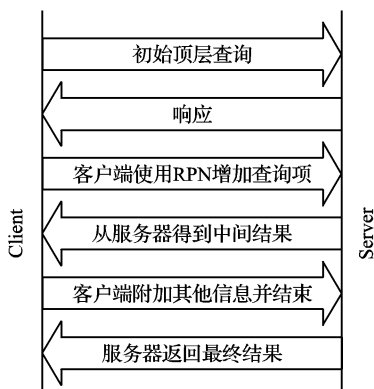


图1 递进目标搜索原理

## 2 关键技术

### 2.1 查询流

实现递进目标搜索的关键技术之一是查询命令能形成查询流。查询流是一种能够根据所发送的附加信息不断更新现有查询的一种方法。这与日常使用的网上搜索类似:先通过某关键字检索出一个结果集,然后再通过另一个关键字在此结果集查询。在此过程中并没有产生新的查询,只不过在原有的查询中增加新的过滤项。这样后面的查询不需要在涉及相关数据的所有数据库中再执行一次,既节省时间又节省带宽。

实现查询流与递进目标搜索的关键技术有两个:一个是多媒体查询方法 MQF;另一个是 XML 片断请求单元 (Fragment Request Units, FRU) 与 XML 片断更新单元 (Fragment Update Units, FUU)。本文将讨论这种技术在查询流生成上的应用以实现递进目标搜索。

### 2.2 多媒体查询方法 MQF

MQF 的具体实现原理请参见参考文献[1],它有三个主要技术特征:逆波兰表示法<sup>[9]</sup> (Reverse Polish Notation, RPN)、查询级别控制和元搜索能力。

#### 2.2.1 逆波兰表示法

在 MQF 中,对多媒体的查询会形成非常复杂的查询表达式,为了便于查询的不断更新,需要服务器要按照查询输入的序列来解析查询。显然要实现这一点查询表达式就不能使用一般的代数表示法,而要采用 RPN。

在通常的代数表示法中,二元运算符总是置于与之相关的两个运算对象之间,所以,这种表示法也称为中缀表示。对中缀表达式的计值,并非按运算符出现的自然顺序来执行其中的各个运算,而是根据算符间的优先关系来确定运算的次序,此外,还应顾及括号规则。因此,要从中缀表达式直接产生目标代码一般比较麻烦。而在 RPN 中,每一运算符都置于其运算对象之后,故称为后缀表示。这种表示法的一个特点是,表达式中各个运算是按运算符出现的顺序进行的,故无须使用括号来指示运算顺序,因而又称为无括号式。

例如要查询所有这样要求的图片:图片要包含山脉或海滩,而且还必须有汽车。以代数表示法表示如下:  $(A \text{ OR } B) \text{ AND } C$ ,而如果用逆波兰表示法就变成了  $A \text{ B OR } C \text{ AND}$ 。使用逆波兰表示法后,服务器就可以按照输入的查询流来解析该表达式,而不再需要考虑运算的优先级。这样极大地减少了查询所需要占用的内存,同时也保证了查询结果的准确性。

#### 2.2.2 查询级别

在 MQF 中,每一个多媒体查询都可以是以下 4 种查询级别之一。

1) 精确匹配。客户端提供的查询要么是精确描述,要么是使用 MPEG-7 或 Dublin Core 描述。

2) 样例。客户端给予服务器查询要求的是与其数据库对应的样例。

3) 语义。客户端给出的是语义的描述,要求服务器在其数据库中查询与该语义对应的多媒体信息。

4) 自由文本。客户端使用自然语言来描述一个查询,需要服务器来解释。

以上 4 级别的查询并不互相排斥,在一个查询中可以包含多种级别,例如:客户端可以在一个查询中包括一个精确匹配(如歌唱家是 U2)、一个样例查询(如一部分 U2 歌曲)、一个语义描述(如歌曲的和弦级数)和一个自由文本(如 1980 年在都柏林举办的音乐会)。

#### 2.2.3 元搜索能力

元搜索能力能够使得查询专注于结果本身,而不是数据的结构。这样就可以使用户只描述所需要的结果集而不去关心数据实际是如何存储的。当某个查询必须从多个服务器上得到结果,而这些服务器上的数据存放模式并不相同的时候,这一点就非常有用了。这种以结果驱动的查询方式最近已经被 W3C (WWW 联盟) 所采用,经 SPARQL (Simple Protocol and RDF Query Language) 批准为 W3C 推荐标准。但 MQF 在这方面以前的研究进行了修改,没有使用有限状态机的方式,从而在现行系统中简化了以执行 MQF 所必须完成的细节。

#### 2.2.4 MQF 举例

MQF 中查询输入的一般格式与输出格式如下所示。由于 MQF 最主要的目的是为了实现在多媒体数据按内容查询,因此,本文的实例均面向使用 MPEG-7 描述的多媒体数据查询。

MQF 查询的输入:

```
<?xml version="1.0" encoding="gb2312"?>
<mqf>
  <queryID>QID_123</queryID>
  <from>http://server.uri</from>
  <replies>
    <reply>
      <item>
        <!-- 使用 MPEG-7 描述的媒体定位 -->
      </item>
      <description>
        <!-- 任何 MPEG-7 描述 -->
      </description>
      <index>
        <!-- 响应的索引 -->
      </index>
    </reply>
    <!-- 其他的响应项 -->
  </replies>
</mqf>
```

MQF 查询的输出:

```
<?xml version="1.0" encoding="gb2312"?>
<mqf>
```

```

<queryID> QID_123 </queryID>
<from> http://client.uri </from>
<query>
  <item queryLevel="example">
    <!--使用 MPEG-7 或 Dublin Core 描述的元素-->
  </item>
  <operator>
    <!-- 操作符 -->
  </operator>
  <!-- 以 RPN 表示的项目与操作符 -->
</query>
</mqf>

```

例如:客户端需要查询所有这样的图片文件:图片应包含山或海滩,同时图片上应有汽车。下面的程序表达了这样一个相对复杂的以逆波兰表示法表示的查询,它代表前述的实例。

```

<?xml version="1.0" encoding="gb2312"?>
<mqf xmlns="urn:mpeg:mpeg7:mqf:schema:2007"
  xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:mpeg:mpeg7:mqf:schema:2007
    ../../schema/mqf.xsd">
  <queryId> ID_2-3 </queryId>
  <from> UOW </from>
  <query>
    <replyType> image/* </replyType>
    <item queryLevel="example">
      <freeText> Mountain </freeText>
    </item>
    <item queryLevel="example">
      <freeText> Beach </freeText>
    </item>
    <operator> OR </operator>
    <item queryLevel="example">
      <freeText> Car </freeText>
    </item>
    <operator> AND </operator>
  </query>
</mqf>

```

### 2.3 查询片断请示单元 FRU 与片断更新单元 FUU

为了从客户端得到“查询流”,需要具备内置形成 RPN 表示的查询命令的能力,高效的查询流必须能够处理未知结构的数据。由于现有 SQL 命令主要用于文本数据库,数据结构又是已知的,而且每条 SQL 命令必须是完整的查询,所以 SQL 不能实现查询流。MQF 恰好就是可实现这样的查询流要求,一是它支持 RPN,二是它的命令格式是开放的。然而,MQF 当初并不是为查询流而设计的,因此,要使 MQF 能使用查询流必须对其进行某些修改。

对 MQF 必须进行的修改是在初始查询中包含一个查询流标记(在 MQF 查询中增加一个属性 stream=true),所有来自客户端的后续处理信息也必须具有是查询流一部分的标记信息。

由于 MQF 中的客户端与服务器端间的查询/响应都是基于 XML 的,同时也遵循了 MPEG-7 的标准。所以查询流的生成与部分修改及服务器的响应都要涉及 XML 文档中的部分片断。在 MPEG-7 中已经内置了一个用于传送 MPEG-7 XML 片断的标准,这个标准提供了一个用于传送含有内部片断更新单元 XML 片断的方法,这些 FUU 指示接受器要插入或更新哪些 XML 片断,哪些节点需要删除。这个技术满足了流传输环境的需要,然而,在查询结果集的上下文中,FRU 与 FUU 还必须具备获取部分结果集元数据的能力,在实际结果集传送之前,这些元数据必须传送到客户端以便用户进行进一步的查询。

与 FUU 对应生成的是片断请示单元,它提供了一种能够获取远程 XML 文档片断的标准方法。用户通过这个技术可

以只传送相关的 XML 文档片断,而不需要传送整个 XML 文档。至于需要传送哪个片断则由事先确定的信息(如 XML Schema)或基于已经接收的 FUU 信息决定。

FRU 操作的结果通过 FUU 传回到原始客户端,FUU 中包含了 XML 片断在原始 XML 文档中定位的上下文路径信息以保持 XML 文档的原始结构。FRU 与 FUU 的原理如图 2 所示。

FRU 实例如下:

```

<FRU>
  <Src> MPEG7Description.xml </Src>
  <Query> //Classification <Query>
</FRU>

```

FUU 实例如下:

```

<FragmentUpdateUnit>
  <FUCommand> addNode </FUCommand>
  <FUContext> /Mpeg7/Description/Creationinformation/
    Classification </FUContext>
  <FUPayload>
    <Description>
      <Statement mimeType="text/plain">
        <CaptionLanguage> en-US </CaptionLanguage>
        <CaptionLanguage> de </CaptionLanguage>
        <CaptionLanguage> ja </CaptionLanguage>
      </Statement>
    </Description>
  </FUPayload>
</FragmentUpdateUnit>

```

借助于在 MQF 中使用 RPN,就可以将 MQF 与 FRU/FUU 相结合以形成查询流。这样做的好处是一方面用户不需要去检查每一次查询的结果,另一方面通过对结果集的描述用户可以排除某些含糊不清的查询关键字。这种递进目标搜索的方法非常适合于带宽很低、传输成本很高和传输速率很慢的场合。

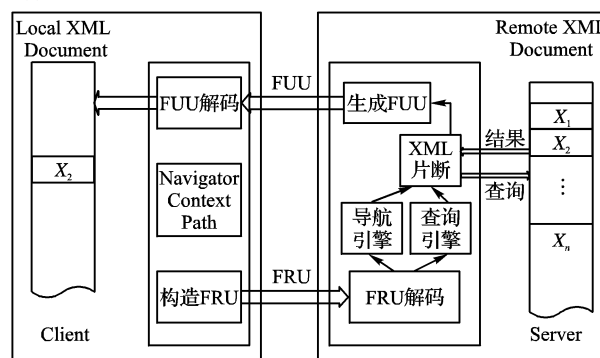


图2 FRU 与 FUU 原理

### 3 应用举例

下面举例说明本文所讨论的递进目标搜索。某用户想搜索山的图片,但是哪一类的山并没有明确的目标,这有待于在搜索过程中临时决定。递进目标搜索的步骤如下。

1) 将以“山”为关键字的初始查询发送到服务器。

2) 服务器在数据库中查询含有“山”关键字的所有图片,并将结果保存在服务器上。同时将包含在结果集中相关的关键字元数据发送回客户端,如断背山、雪山、山脉、山地自行车和冷山等。

3) 在返回的相关的元数据关键字列表中,用户发现被雪覆盖的山正是他/她想要找的,这样就再次选择“雪山”进行递进查询,这样就可以得到“雪山”的查询结果。

4) 根据需要重复第 2) 步,可以对查询进行进一步的细化,这种递进搜索的过程可以很快找到所需要的目标。

(下转第 1102 页)

时间。

设计了 XML 索引表结构,故需要多增  $m+1$  个单元的存储空间。另外,在数据服务商处设置一个阈值,用于规定是否存储查询的初始结果,原理是当某用户查询的频率超过该值时,就存储其初始查询结果,那么它是利用一部分的空间代价来取代时间代价。基于现代的硬件发展,达到更好的时间复杂性比达到更好的空间复杂性要求更强烈。

## 5 结语

ILISA 充分考虑了 XML 数据库文档的结构:它是一棵非常“宽”而高度较低的树型结构的文档,而它具有极好的对称性。基于此类性质,本文设计的数据结构可以将树型结构数据的检索变换为顺序链表检索,这不但带来了更优的检索时间复杂性,而且也大大减小了数据的检索空间,即数据规模从  $n$  个用户子树的检索减少到  $m$  个索引项的检索,通常  $n$  比  $m$  大得多。另外,从 Dewey 编码的构造原理,添加用户子树到 XML 数据库文档时,只需要获取当前 XML 索引表中的 Final 数据域的值,对新增用户子树进行 Final 数据域值加 1 的 Dewey 编码,再将其添加到文档的最后,每次的 Dewey 编码空间将会很小。ILISA-1 与 ILISA-2 有相同的时间复杂性,而它们比起  $O(n+e)$  的时间复杂度要优异得多。

### 参考文献:

- [1] HACIGUMUS H, IYER B, MEHROTRA S. Providing database as a service [C]// Proceedings of the 18th International Conference on Data Engineering. Washington, DC: IEEE Computer Society, 2002: 29.
- [2] 孔令波,唐世渭,杨冬青,等. XML 数据索引技术[J]. 软件学报, 2005, 16(12): 2063-2079.

- [3] JAMMALAMADAKA R C, MEHROTRA S. Querying encrypted XML documents [C]// Proceedings of the 10th International Database Engineering and Applications Symposium. Washington, DC: IEEE Computer Society, 2006: 129-136.
- [4] 孔令波,唐世渭,杨冬青,等. XML 信息检索中最小子树根节点问题的分层算法[J]. 软件学报, 2007, 18(4): 919-932.
- [5] 王静,孟小峰,王宇,等. 以目标节点为导向的 XML 路径查询处理[J]. 软件学报, 2005, 16(5): 827-837.
- [6] 姚全珠,丁晓剑,任雪利. 一种新的基于 XML 的索引机制[J]. 计算机工程, 2006, 32(15): 15-17.
- [7] GRUST T. Accelerating XPath location steps [C]// Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2006: 4-6.
- [8] GRUST T, van KEULEN M, TEUBNER J. Accelerating XPath evaluation in any RDBMS [C]// ACM Transactions on Database Systems. New York: ACM, 2006: 91-131.
- [9] DeHAAN D, TOMAN D, CONSENS M P, et al. A comprehensive XQuery to SQL translation using dynamic interval encoding [C]// SIGMOD: Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data. San Diego: ACM Press, 2003: 623-634.
- [10] AMAGASA T, YOSHIKAWA M, UEMURA S. QRS: A robust numbering scheme for XML documents [C]// ICDE'03: Proceedings of 19th International Conference on Data Engineering. Washington, DC: IEEE Computer Society, 2003: 705-707.
- [11] RUNAPONGSA K. Methods for efficient storage and indexing in XML databases [D]. Ann Arbor: University of Michigan, 2004: 18-23.
- [12] 刘大有,唐海鹰,孙舒杨,等. 数据结构 [M]. 北京: 高等教育出版社, 2001.

(上接第 1098 页)

通过实验,本实例使用直接搜索与使用递进目标搜索传输到客户端的总的数据量比较如表 1 所示。

表 1 不同方式下的多媒体查询数据传输大小比较

搜索方式	说明	传输的 XML 文件大小/B		
		总尺寸	附加的二进制尺寸	MQF 尺寸
直接搜索	直接得到结果	114 743	97 979	16 764
递进目标搜索	结果集描述	6 805	—	6 805
	结果	19 186	12 163	7 023
	合计	25 991	12 163	13 828

从表 1 可以看出,在不使用递进目标搜索的情况下,向客户端传输文件的总尺寸为查询结果的二进制文件与 MQF XML 文件大小之和;而在使用递进目标搜索的情况下,传输的总尺寸为两块:一块为先返回的结果集描述数据;另一块则为最终的查询结果数据。两者比较可以看出,使用递进目标搜索的传输总量只有直接搜索的传输总量的 22.7%,传输的数据量下降了 78%,显然这将大大降低数据传输所需的带宽与速度,所以特别适合于各种移动终端使用。

## 4 结语

本文探讨的是一种基于内容的多媒体信息搜索的方法,通过将 MQF 与 FRU/FUU 相结合,通过从服务器返回结果集描述的元数据来实现递进目标搜索,而不需要返回结果集本身。这种方法特别适合于使用移动终端来进行多媒体数据库的搜索,从而大大降低了客户端与服务器的数据通信量。另外,只要采用一些标准的压缩算法(如 MPEG-7 中的 TeM 标

准)还可以进一步对传输的数据进行压缩,使得传输的数据量明显减少。

### 参考文献:

- [1] MARTÍNEZ J M, KOENEN R, PEREIRA F. MPEG-7: The generic multimedia content description standard, Part 1 [EB/OL]. [2009-06-10]. [http://www.chiariglione.org/mpeg/tutorials/papers/IEEEEMM\\_mp7overview\\_withcopyright.pdf](http://www.chiariglione.org/mpeg/tutorials/papers/IEEEEMM_mp7overview_withcopyright.pdf).
- [2] ZHAO L. DC 元数据介绍 [EB/OL]. [2009-05-10]. <http://www.dlresearch.cn/download/metadata/xiamen2.pdf>.
- [3] 曹忠升,吴宗大,王元珍. 多媒体查询语言及其评价准则 [J]. 计算机科学, 2009, 36(3): 9-13.
- [4] IBM, Watson Research Center. MARVEL: Multimedia analysis and retrieval system [EB/OL]. [2009-05-20]. <http://www.research.ibm.com/marvel/>.
- [5] Google. Google images [EB/OL]. [2009-04-10]. <http://www.google.com/imghp?hl=en&tab=wi&q>.
- [6] 李国辉,王辰,柳伟. 基于内容的多媒体数据库系统引擎 CDB [J]. 小型微型计算机系统, 2004, 25(7): 1113-1118.
- [7] 张正兰,李冬梅,赵春燕. 多媒体数据库查询技术研究 [J]. 微机发展, 2004, 14(3): 70-73.
- [8] ADISTAMBHA K, RITZ C, BURNETT I S. MQF: An XML based multimedia query format [C]// ICME 2007: Proceedings of the 2007 IEEE International Conference on Multimedia and Expo. Washington, DC: IEEE Computer Society, 2007: 264-267.
- [9] MCHROY M. Reverse polish notation [EB/OL]. [2009-05-18]. <http://mathworld.wolfram.com/ReversePolishNotation.html>.