

一种有效的电子商务描述模型和多站点检索机制

彭代毅¹, 尹德辉^{1,2}, 李炳法¹

(1. 四川大学 计算机学院, 四川 成都 610064; 2. 宜宾学院 计科系, 四川 宜宾 644007)

(vinvinsent@263.net)

摘 要:目前的电子商务模式中存在着两个问题。一是缺乏多站点间商品信息的统一描述规范, 其二是没有一种有效机制根据统一的描述形式建立高效率的站点关系。给出一种基于 XML 的多元电子商务统一描述和关系模型, 解决上述的第一个问题。提出一个以该模型为基础的分布式智能体框架, 实现一套基于商品营销关系的多站点检索机制。

关键词:XML; 电子商务; 电子商务; 统一描述; 智能体; 多站点检索

中图分类号: TP302.1 **文献标识码:** A

Effective descriptive model of E-commerce merchandise and the multi-website searching mechanism

PENG Dai-yi¹, YIN De-hui^{1,2}, LI Bing-fa¹

(1. School of Computer Science and Engineering, Sichuan University, Chengdu Sichuan 610064, China;

2. Department of Computer Science, Yibin College, Yibin Sichuan 644007, China)

Abstract: Although E-commerce was being popularized and applied all over the world, there were still unsolved problems in current E-commerce modes. Among them, the fact that there was no uniform description for E-merchandise is a critical limit for the development of E-commerce. Besides, there was not an effective mechanism for describing website relationship, enabling effective multi-website search. Aimed to solve above problems, this paper proposed an uniform description and a relationship model for E-merchandise, which were based on XML. Afterwards, a distributed intelligent agent framework was introduced to realize a multi-website searching mechanism for E-commerce based on prior relationship model.

Key words: XML; E-merchandise; E-commerce; uniform description; intelligent agent; multi-website search

0 引言

电子商务是一项复杂的系统工程, 它的实现还依赖于众多的社会问题、技术问题的逐步解决和完善。其中, 电子商务描述的不统一是制约电子商务发展^[1]的重要因素。这使得建立于统一描述规范之上的高效率多站点信息检索变得非常困难。

众多商家对自己商品的描述中, 既有语言的差异, 又有命名规范的差异。如果不将商品的描述形式加以统一, 商品在网络上的表示将永远只是用于人眼浏览的符号。好在 XML 在这网际描述非常混乱的时刻应运而生, 这使电子商品的统一描述成为可能, 也为多站点的高效率相关信息检索打下了基础。下一代的电子商务平台将是建立在 XML 统一规范之上, 由智能体管理复杂的内部运行机制的关系网络平台^[2]。

1 现行电子商务体系中的缺陷

现在的电子商务体系中, 尚无一种公认的规范来统一电子商品信息的描述形式。尽管很多站点已开始使用 XML 来描述自己的商品, 但几乎都是一个站点一个描述方法。例如: 商品这个关键字可能在 A 站点被描述为 <Commodity>, 在 B 站点

却被描述为 <商品>。这使跨系统的信息查询变得十分困难, 也使客户在购买商品时难以和其他供货商的商品指标进行对比。因此, 建立一套完备的电子商品描述规范是实现多站点数据交流的必经途径(目前一些国际机构已开始研发电子商务方面的 XML 规范, 如 cXML^[3], 但并未得到广泛使用)。

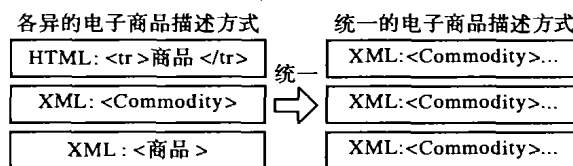


图1 电子商品描述方式的统一

现有的搜索引擎往往都通过网络爬虫的方式采集分布在各个站点的数据。在统一信息描述形式下, 网络爬虫能够高效、准确地搜索出和关键字匹配的相关信息。但是, 电子商务的信息组织不仅需要能够准确匹配商品关键字, 还要求根据商品特有的属性有效地将信息分类。网络爬虫这种数据采集方式很难反映现实生活中商品的营销关系以及商品在流通领域中所经过的途径。

要实现上述的体现营销关系的站点结构, 就需要按照特定的方式将相关联的站点连接起来。由于每个站点中都有很

收稿日期: 2004-07-22; 修订日期: 2004-10-27

作者简介: 彭代毅(1981-), 男, 四川金堂人, 硕士研究生, 主要研究方向: 并行与分布式计算; 尹德辉(1963-), 男, 四川宜宾人, 副教授, 硕士研究生, 主要研究方向: 数字水印版权保护与图像处理、电子商务安全; 李炳法(1947-), 男, 教授, 主要研究方向: 并行处理、分布式操作系统及信息安全。

多不同的产品,并且站点与站点的联系都是建立在特定产品上的。因此,这种站点间关系应该是基于特定产品的逻辑关系。又因为这种逻辑关系在站点间普遍存在,所以在实现这种站点关联体系时应使用松耦合的形式。本文提出一种立体的电子商品统一描述模型,这种模型既能方便站内信息的分类检索,又能有效地进行站点间相关信息的组织。

2 一种统一的电子商品描述模型

用面向对象的方法描述商品,每个商品就是一个具有多重属性的对象。从商品类别的角度看,它具有商标、型号、商品性能描述等属性,我们称这些属性为固有属性,因为一个商品的内在本质就决定了它所在的类别。除了商品的固有属性外,商品在其流通过程中的不同阶段具有不同的指标,不同的流通过程对应着不同的属性块,例如,两个相同的产品从同一个生产商到两个不同的销售商,其路径是不同的,这可能导致最终的产品指标的不同(如价格)。我们定义其流通属性为商品的扩展属性。一般而言,商品的最终指标是由其起源和流通途径决定的。一个商品的流通途径一般会包括出厂、运输、批发1、批发2、……、零售等阶段。为了方便描述,我们将运输到零售之前的所有环节定义为中间环节,那么,每一个商品的扩展属性都由生产商描述、中间环节描述、最终销售商描述三部分组成。

商品的固有属性是在商品的内在本质决定的时候就确定了,即一个商品一旦被生产商生产出来,它的商标、型号、性能指标都已不再改变。我们利用一个基本块来表示商品的固有属性,这个基本块是体现商品本质的描述,记录了商品出厂时最原始的性能指标,如品牌、型号、类别、性能指标描述等。下面给出基本块描述:

```
< Basic >
  < Product_ID > CCN00001 </Product_ID>
  < Brand > ThinkPad </Brand>
  < Manufacturer > IBM </Manufacturer>
  < Series > S </Series>
  < Module > 102 </Module>
  < Appearance >
    < Color > Black </Color>
    < Dimension >
      < X measure = 'cm' > 30 </X>
      < Y measure = 'cm' > 20 </Y>
      < Z measure = 'cm' > 2.5 </Z> </Dimension>
    < Material > Metal </Material>
  </Appearance>
</Basic>
```

基本块可以表示商品的固有特征,但它并不能满足商品表示的所有需求,例如价格属性,在不同的站点可能不同,因此我们定义了一个扩展块来记录各站点关于该产品的扩展信息。在扩展块中,我们定义了 Relationship 块来定义该商品在该站点的供销关系,其中 From 字段表示了该商品的来源,ArrayOfDestination 表示了该商品还将分发到的下一级站点的 URL。需要说明的是,如果一个站点已经是最终销售商,那么它的 ArrayOfDestination 内容为空。通过记录商品来源和商品的去处,商品在本站点的流通关系便被表示了出来。这种关系将在后面的模型中被用来构造站点关系树。下面程序给出了单种商品在不同站点上的统一描述规范:

```
< Commodity >
  < Basic >
```

```
.....
</Basic>
< Extension >
  < Site > http://www.intermediate5.com </Site>
  < Price measure = 'RMB' > 15000 </Price>
  < Description > Intermediate5 </Description>
  < Relationship >
    < From > http://www.intermediate4.com </From>
    < ArrayOfDestination >
      < Dest > http://www.sales1.com </Dest>
      < Dest > http://www.sales2.com </Dest>
    < ArrayOfDestination >
      < Relationship > ...
    < Relationship > ...
  </Commodity>
```

统一了特定商品在不同站点的描述后,我们还需要建立一套本地商品分类机制以高效地在站内检索商品。给出用于本地商品分类检索的 XML 框架,采用 Category 项目来描述特定类型的商品集合,用“点分隔”的命名空间表示法描述类别信息。若查询某一大类商品的信息,如“computer”,搜索引擎将在“commodity.computer.*”中检索商品信息,“commodity.computer”及其子空间内的所有商品将成为被检索的对象。

本地商品的分类信息表示如下:

```
< Category namespace = "commodity.computer.notebook" >
  < Commodity >
    < Basic > ... </Basic>
    < Extension > ... </Extension>
  </Commodity>
  ... <!-- Other Commodities -->
</Category>
```

3 利用统一的描述模型建立站点间关系

从上述的站内商品统一描述模型中,可以看到:站内不同类别的商品对应着唯一的商品描述块,而不同站点内的同类商品都具有不同商品描述块中相同的商品基本块。商品基本块实质上统一了同类型商品在整个互联网上的描述。商品扩展块除了记录商品在本地的描述外,还给出了以本地站点为中心的流通关系局部索引。通过该局部索引,我们可以建立如图2所示的流通层次图。该层次结构体现了特定商品在互联网上商务网站间的逻辑关系。

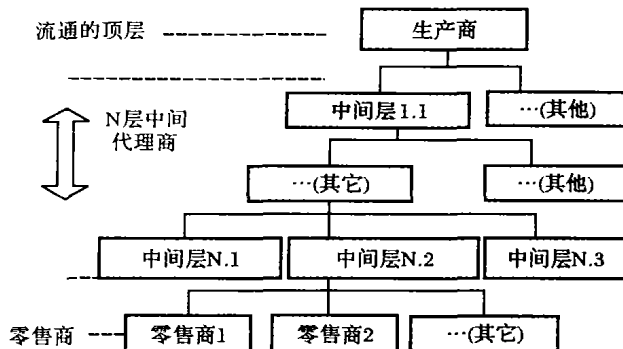


图2 商品分发关系的树状结构

通常情况下,购买者只关心特定商品在不同销售商处的指标,中间供货商对购买者是透明的,那么,在返回站点信息时,只需要返回流通关系树中的叶子节点信息。

4 一种分布式的相关信息查询智能体框架

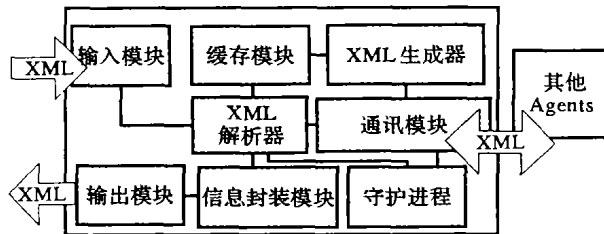


图3 Agent 内部结构

为了使站点间关系的维护对各站点透明,使所有站点间信息交互自动完成。我们将给出一种用于进行相关信息查询的分布式智能体框架。每个商务站点中都种植着一个智能体 (Agent), Agent 有权访问本地描述所有电子商品的 XML 文件。当站点需要得到某商品的相关信息时,只需将商品在 XML 文件中的位置告诉 Agent, 并根据浏览者的需求输入查询范围, Agent 便能返回满足条件的销售商指标。Agent 内部结构如图 3 所示。

Agent 的运行机制为:本地站点将客户查询请求以 XML 方式传入智能体, Agent 通过 XML 解析器读出对应商品的产品号以及扩展块中的 Relationship 内的 From 字段, 通讯模块将 From 字段中的 URL 作为目标地址, 将产品号、请求查询范围等参数封装成 XML 发送到互联网, 完毕后立刻创建一个新的线程等待数据返回, Agent 进程等待新的查询请求。

查询请求的 XML 表示程序如下:

```
<QueryRequest>
  <Query_ID> 1 </Query_ID>
  <Product_ID> CN00001 </Product_ID>
  <Hops> 3 </Hops>
  <From> http://www.sales1.com </From>
  <QueryItems>
    <Property> Price </Property>
    <Property> Description </Property>
    <Property> Site </Property>
  </QueryItems>
</QueryRequest>
```

查询结果的 XML 表示程序如下:

```
<QueryResult>
  <Query_ID> 1 </Query_ID>
  <Product_ID> CCN00001 </Product_ID>
  <From> http://www.sales3.com </From>
  <ResultItem>
    <Price measure = 'RMB'> 15000 </Price>
    <Description> Sales3 </Description>
    <Site> http://www.sales3.com </Site>
  </ResultItem>
</QueryResult>
```

QueryRequest 中的 Query_ID 字段用来标识每一次查询操作, Product_ID 字段记录了请求查询的商品号, Hops 字段的值是自底向上要求查询的层数, From 字段记录了查询的源地址, QueryItems 是一个数组, 记录了请求返回的结果中包含哪些商品信息扩展块中的字段, 其中 Property 的取值必须是商品信息扩展块中的字段名。QueryResult 中的 Query_ID、Product_ID 和 QueryRequest 中的对应字段含义相同, 只是

From 字段记录的是提供结果集的站点地址。ResultItem 是根据 QueryRequest 的 QueryItems 动态生成的结构, 结构中记录了该站点中对对应商品请求返回的值。

需要说明的是, Agent 一开始运行, 就有一个后台进程监视通信接收队列。如果发现队列中的信息是其他 Agent 发送过来的查询请求, 则根据不同条件决定下一步操作: 如果自己是叶子节点, 且请求 From 字段不是自己的地址, 则立刻查询本地信息, 按照图 4 的格式生成结果报文, 将收到的请求包中 From 字段作为目标地址进行发送。如果自己是非叶子节点, 则先将查询请求转发给自己的子节点, 然后判断请求来源, 来源是孩子节点时, 进行 $Hops = Hops - 1$ 操作, 如果操作后 $Hops > 0$, 则继续向父节点转发修改过 Hops 值的请求 (父节点地址记录在本地对应商品的 From 字段中), 否则不再转发。当来源是父节点时, 不再做任何操作。

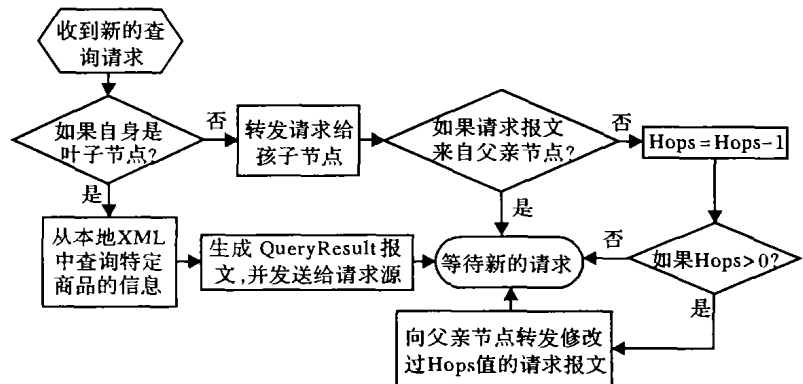


图4 Agent 收到查询请求后的处理流程

当 Agent 收到的是结果报文时, 根据报文的 Query_ID 和 Product_ID 字段查询到对应的已发送请求, 将信息封装后返回给本地的网站系统, 加以适当的用户界面, 便可将结果显示在查询者的面前。

5 结语

本文中提出了一种统一的电子商务描述模型, 并通过商品两个方面的属性有效地建立了多站点的关系体系。这种关系模型反映了现实生活中商品的流通关系, 为高效率组织数据和查询信息提供了保证。

参考文献:

- [1] United Nations Conference on Trade and Development: E-Commerce and Development Report 2002[R], Chapter 1: 3-5.
- [2] ZHAO J, BLUM T. Next-Generation E-Commerce: XML + Mobile Agent + Trust[J], CG topics, 2000, (4).
- [3] Commerce XML Specification[EB/OL]: <http://xml.coverpages.org/cxml.html>, 2004.
- [4] HOLSAAPPLE CW, SINGH M. Toward a unified view of electronic commerce, electronic business and collaborative commerce: a knowledge management approach[J]. Knowledge and Process Management, 2000, (7).
- [5] ABITEBOUL S, BUNEMAN P, SUCIU D. Data on the Web From Relations to Semistructured Data and XML[M]. Morgan Kaufmann Publishers, 2000.
- [6] MAES. Modeling Adaptive Autonomous Agents[J]. Artificial Life Journal, 1994, (1).