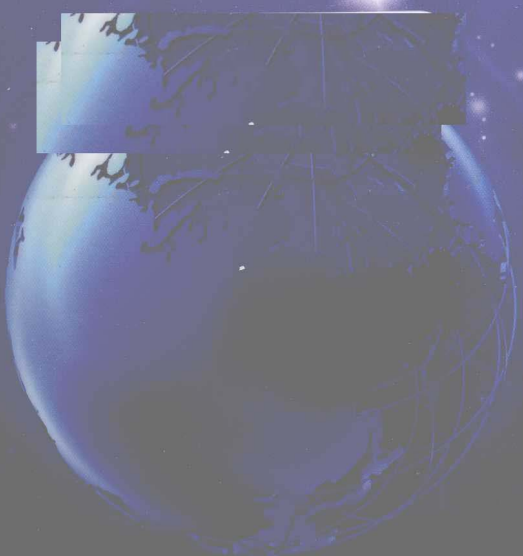
XML数据查询与信息检索系统

韩忠明 著



中国水利水电出版社
www.waterpub.com.cn

XML 数据查询与信息检索系统

韩忠明　著

中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书主要研究改进 XML 数据查询和信息检索的相关理论与技术，以便于它们更好地集成在一起，从而可以更加优化地执行用户的查询需求，针对这个研究目标，本书做了大量的研究工作。本书提出了一个新颖有效的节点编号模式，详细讨论了节点编号模式的定义和性质，还提出了一种新颖有效的对基于 XML 信息检索查询进行相关度打分的算法，这个打分机制结合了检索查询关键词的频度、文档的结构化特性、文档的语义特性等。基于对结构化查询和信息检索的相关研究成果，本书提出了处理 XML 结构化查询和信息检索的有效算法与机制。本书还讨论了一个原型系统的设计目标、分析与设计过程。

图书在版编目（CIP）数据

XML数据查询与信息检索系统 / 韩忠明著. -- 北京 : 中国水利水电出版社, 2010.3
ISBN 978-7-5084-7151-8

Ⅰ. ①X… Ⅱ. ①韩… Ⅲ. ①可扩充语言, XML－程序设计②计算机应用－情报检索－检索系统 Ⅳ. ①TP312②G354.4

中国版本图书馆CIP数据核字(2010)第008406号

策划编辑：雷顺加　责任编辑：张玉玲　加工编辑：胡海家

书　名	**XML 数据查询与信息检索系统**
作　者	韩忠明　著
出版发行	中国水利水电出版社 （北京市海淀区玉渊潭南路 1 号 D 座　100038） 网址：www.waterpub.com.cn E-mail：mchannel@263.net（万水） sales@waterpub.com.cn 电话：（010）68367658（营销中心）、82562819（万水）
经　售	全国各地新华书店和相关出版物销售网点
排　版	北京万水电子信息有限公司
印　刷	北京市天竺颖华印刷厂
规　格	170mm×227mm　16 开本　8.25 印张　168 千字
版　次	2010 年 3 月第 1 版　2010 年 3 月第 1 次印刷
印　数	0001—1000 册
定　价	30.00 元

凡购买我社图书，如有缺页、倒页、脱页的，本社营销中心负责调换

前　言

XML 已经成为互联网上数据表示和数据交换的标准。随着 XML 文档数据量和文档数量的快速增长，产生了很多问题，其中很重要的一个问题就是如何有效地查询这些文档，也就是结构化查询，亦称为数据查询。而有效的数据查询又涉及文档的存储机制和索引结构等问题，这些问题已经引起了学术界和工业界广泛的研究热情，学者们在这些问题上作了大量的研究工作。另外一个问题是基于 XML 文档的信息检索，这也是一个新产生的研究问题。面对巨大的网络信息，如何才能为用户检索出真正有效的信息是一个非常具有挑战性的研究问题。现存的大部分搜索引擎是基于关键字搜索的，页面排序算法采用页面的超链接或页面内容的文本特性。如果页面采用 XML 来编写，那么就需要合理地利用 XML 的结构特性、语义特性以及其他的相关性质来提高检索的效果和效率。这就促使对 XML 文档进行信息检索成为了一个非常有意义的问题。XML 上的信息检索正开始受到学术界的高度重视。

对 XML 进行结构化查询和信息检索是两个既具有相关性又具有不同特性的研究问题，如何集成这两个研究问题就更加具有研究意义。本书的主要研究对象就是如何改进 XML 的结构化查询以及信息检索的相关理论与技术，以便于它们更好地集成在一起，从而可以更加优化地执行用户的查询需求。

针对这个研究目标，本书做了大量的研究工作。首先，本书在 XQuery 语言的基础上扩充了全文本检索功能，为了与原来的 XQuery 区分，扩充后的语言称为 XQuery+（XQuery Plus）。XQuery+语言有如下特点：在 XQuery+语言里，扩充了 XQuery 的检索功能，增加了一个为检索服务的谓词；在 XQuery+中还支持检索词的布尔操作。

本书的主要研究任务之一是如何有效地处理 XML 的结构化查询。作为处理 XML 结构化查询的基础，XML 文档的节点编码模式和索引结构是研究的核心问题。本书提出了一个新颖有效的节点编号模式，详细地讨论了节点编号模式的定义和性质。节点编号模式为 XML 文档索引和查询提供了基础，一个有效的节点编号模式应该可以包含结构信息，易于支持索引和查询。从本书给出的节点编号定义和性质分析，我们知道编号模式可以满足这些基本的要求。本书在这个节点编号模式的基础上建立了一个 HiD 索引结构，HiD 索引结构有效地集成了结构索引和值索引两个部分。通过大量有竞争性的实验分析表明，采用 HiD 索引机制方法可以在索引的构建时间和空间消耗上得到很好的平衡和性能表现。

本书研究的第三个主要任务是基于 XML 的信息检索。XML 信息检索的核心问题是如何进行相关度打分。本书提出了一种新颖有效的对基于 XML 信息检索查

询进行相关度打分的算法，该算法同时考虑了结构相关度和语义相关度。结构相关度主要利用了检索词的距离概念；语义相关度的计算则采用了节点相关度语义权重系数的方法。为了合理地评价和比较本书提出的方法与其他研究者的方法之间的效果差异，本书还做了大量的实验。从所做的实验结果中可以看出，在合理应用本书的方法后，检索的查全率和查准率都得到了显著提高，检索结果非常合乎用户的需求。

基于对结构化查询和信息检索的相关研究成果，本书提出了处理 XML 结构化查询和信息检索的有效算法与机制。这些算法分别处理了 XQuery 和 XQuery+查询。虽然这些算法都基于 HiD 索引结构之上，但是这些算法的特点不同，处理对象不同。对于 XQuery 查询来说，本书给出的两种算法分别是处理单路径查询的算法和具有两个分支的树模式查询的算法。基于这两种算法，可以方便地构造出处理复杂查询的算法。而对于 XQuery+查询的处理，本书也给出两种不同的处理算法。算法 XQuery+G-1 采用了 on-the-fly 的查询和打分机制，而算法 XQuery+G-2 则简单地采用了查询后计算相关度的技术。通过实验，本书还详细地分析了各种算法的性能和效果，为了合理地评估相关算法的性能，实验中对不同的算法还选用了不同的、合理的比较算法。实验结果表明无论是处理结构化查询还是信息检索，本书提出的对应算法都表现出较高的执行效率，有效地提高了查询和检索的速度。

课题的最后一个研究任务是在相关研究成果的基础上设计开发一个原型系统。本书详细地讨论了原型系统的设计目标、分析与设计过程，确定了原型系统的架构。经过分析原型系统的系统流程，得出各个模块的功能与实现过程。最后，我们采用 Java 语言并在 Qizx/open 的基础上实现了原型系统。从原型系统的体系架构和模块功能可以看出，原型系统基本可以满足 XML 文档查询和检索的需求。原型系统的特色在于：①开放和层次化的结构，这样可以方便地支持和扩充新的功能和算法；②原型系统实现了两种过滤机制和两种结果表示方法，这些都扩展了原型系统的性能和表现力，为将来做成熟的系统打下了良好的基础。

全书组织结构如下：

第 1 章，介绍课题研究问题的背景以及相关研究，并分析课题研究的主要内容及研究意义。

第 2 章，介绍如何在 XQuery 语言的基础上扩展检索功能。为了使 XQuery 语言满足信息检索的要求，本书引入了一个新的检索谓词，并且在检索谓词中支持检索条件的布尔组合。

第 3 章，主要论述节点编号模式与索引结构。首先定义了基本概念，然后详细地给出了节点编号模式定义、性质以及一些应用算法等，在节点编号模式的基础上提出 HiD 索引结构，HiD 索引结构包含结构索引和值索引结构等。

第 4 章，解决了 XML 信息检索的一个核心问题，即节点相关度打分机制问题，这个研究为处理 XML 检索查询的算法提供了基础。主要的研究内容包括 XML 检索的表达以及节点打分算法和排序机制。

第 5 章，在第 3 章和第 4 章的基础上详细介绍了处理 XML 结构化查询和信息检索的查询处理算法以及合成两种查询算法的机制。

第 6 章，给出了原型系统的分析与设计过程，详细描述了原型系统核心模块的处理过程和功能，并介绍了原型系统的几个特色与简单使用方法。

第 7 章，进行了全书的总结，分析了本书研究内容的主要结果以及可能存在的一些问题，最后讨论了下一步可能的几个研究方向。

作　者

2010 年元月

目　　录

第 1 章　绪论

1.1　研究背景

XML（可扩展标记语言）是 Web 数据使用的通用语言，具有结构化、规范性、可扩展性、简洁等特点。它可使开发人员将来自各种应用程序的结构化数据传送给桌面以便在本地计算和表示；允许为特定应用程序创建独特的数据格式，是结构化数据从服务器到服务器传输的理想格式；它是在超级分布式系统之间实现多数据集传输的一种手段；它可同时使开发人员以更具价值的新型方式聚集和组合各种来源的数据。正是因为它有众多优点，所以 XML 已经成为互联网上数据表示和数据交换的标准。

在互联网上，很多大型的网站系统已经开始采用 XML 文档作为页面。在学术界、工业界也都涌现出大量的 XML 文档。XML 继承了 SGML（通用标记语言标准）的强大功能，又充分采取了 HTML（超文本标记语言）的“易用”原则。结构化资源（XML）和资源的描述框架（RDF）互相配合，将大大提高信息查找效率。XML 简化元数据的提取工作，从而协助人们寻找信息，并协助信息生产者和信息消费者的相互发现。使用了 XML，人们可利用设备的智能去访问不同的网站，并对信息进行集中，逐渐实现将控制信息的权力交给那些需要信息的人们。由于所有文件都以 XML 格式存在，所有的用户都可以方便地查找和使用其中的信息。内容供应者、合作伙伴和用户可高效地沟通和共享信息，这就创造出了一种全新的协同工作模式。这一切都说明，XML 为智能代理、人工智能、数据挖掘等技术在信息检索领域的应用开辟了广阔的天地，XML 将使信息检索系统更加智能和准确。

大量的 XML 文档以不同的存储方法存储在不同的系统上。随着文档数据量和文档数量的快速增长，产生了两个问题。第一个问题是如何有效地查询这些文档。有效的文档查询涉及文档的存储机制和索引结构研究问题。这些问题已经引起人们广泛的研究热情，学者们在这些问题上做了大量的研究工作。第二个问题是基于 XML 文档的信息检索问题。面对巨大的网络信息，如何才能为用户检索出真正有效的信息是一个非常具有挑战性的问题。现在大部分搜索引擎都是基于关键字搜索，在页面排序上是基于页面的链接或页面内容来检索结果的。如果页面采用 XML 来编写，这就需要合理地利用 XML 的结构特性，那么如何对 XML 文档进行信息检索（Information Retrieval）就是一个非常有意义的问题。这个问题已经引起学术界的高度重视。信息检索，其核心为文本信息的索引和检索，包括信息的存

储、组织、查询、存取等。信息检索是一个一直以来都在研究的热点问题，有大量研究者从事信息检索的研究。

XML 的结构化查询主要是基于 XML 的结构上精确的数据查询，当用户给出一个 XML 的查询时，要求查询系统返回的是精确的查询结果节点集；而 XML 的信息检索主要是基于 XML 文档的内容，即以文本信息为主的。用户给出的一般是文本信息的要求，而结构信息等可能是模糊的、不精确的，要求返回的节点可能在查询时也不会被清晰地定义好。信息检索的结果是给出和用户要求的文本信息最相关的节点集或者最相关的前 K 个结果，也就是所谓的 TOP-K 查询。在信息检索领域，涌现了大量新型智能检索技术，如职能检索、知识挖掘、全息检索等。下面简单介绍一下这些技术。

（1）智能检索。智能检索利用分词词典、同义词典、同音词典改善检索的效果。比如用户查询“计算机”，那么与“电脑”相关的信息也能检索出来；进一步还可在知识层面上辅助查询，通过主题词典、上下文词典、相关同级词典形成一个知识体系，给予用户智能知识提示，最终帮助用户获得最佳的检索效果。智能检索还包括歧义信息检索处理，如“苹果”究竟是指水果还是指电脑品牌，将通过歧义知识描述库、全文索引、用户检索上下文分析以及用户相关性反馈等技术结合处理，高效、准确地反馈给用户，使其得到最需要的信息。

（2）知识挖掘。知识挖掘主要指文本挖掘技术，目的是帮助人们更好地发现、组织、表示信息，提取知识，满足信息检索的高层次需要。包括摘要、分类（聚类）和相似性检索等。自动摘要就是利用计算机自动地从原始文献中提取文摘。在信息检索和服务中，自动摘要有助于用户快速评价检索结果的相关程度和多种形式的内容分发。自动分类可基于统计或规则，经过机器学习形成预定义分类树，再根据文档内容特征将其归类。自动聚类则是根据文档内容的相关程度进行分组归并。相似性检索技术基于文档内容特征检索与其相似或相关的文档，是实现用户个性化相关反馈的基础，也可用于去重分析。

（3）异构信息整合检索和全息检索。在信息检索分布化和网络化的趋势下，对于信息检索系统的开放性和集成性要求越来越高，需要它能够检索和整合不同来源和结构的信息，这是异构信息检索技术发展的基点，包括支持各种格式化文件的处理和检索；支持多语种信息检索；支持结构化、半结构化及非结构化数据的统一处理等。全息检索即支持一切格式和方式的检索，从目前实践来看，已经发展到异构信息整合检索的层面，而基于自然语言理解的人机交互以及多媒体信息检索整合等方面尚有待进一步突破。

从实际的应用角度分析，信息的充分利用和交流是实现真正的信息化的基础。美国财富 500 强企业网站信息检索工具拥有率几乎达到了 100%，而中国企业的信息利用状况则形成了极大的反差。企业只是追逐时尚，花费大量的人力物力构建自己的网站和丰富的内容体系，然而信息检索工具却不尽人意。因此，发展信息检索方面的应用是当前中国企业信息化建设的重要任务。

目前，Web 的构造语言主要是非结构化的 HTML 语言，网上信息检索主要是依靠搜索引擎，搜索引擎多是基于关键词的全文检索，搜索结果中常常包含了许多冗余信息，而一些有用的相关信息却检索不出来。例如，用 Google 查找电影“Greentea”，如果以“Green Tea”作为关键词进行检索，用户会得到 GreenTea Gum、Green Tea Health Benefit、Green Tea Press、Film2green Tea 等 1 600 000 项查询结果。一个用户希望了解动态规划算法，虽然查到了有关算法的很多文档，但他还是不得不浏览整个文档，查找动态规划这一小部分，这些都大大影响了检索效率。XML 作为 SGML 的子集，为 Web 简化提供了一种定义和描述结构数据的方法，如果上述电影、文献、资料等使用 XML 文档描述，那么检索的查全率和查准率都将大大提高。

以上诸多问题都催生了研究基于 XML 的信息检索的必要性和紧迫性，这也是本书研究的主要动力。作为大型的 XML 文档，不同的用户和不同的应用场合需要有不同的处理引擎系统，这样如何改进 XML 的数据查询和信息检索技术，以便可以提供一个一致的查询处理引擎就成为了本书研究的核心问题。

1.2 XML 介绍

XML（eXtensible Markup Language，可扩展标记语言）是由 W3C（World Wide Web Consortium，万维网联盟）于 1998 年 2 月发布的一种标准。由于 XML 将 SGML 的丰富功能与 HTML 的易用性结合到了 Web 的应用中，所以自推出以来，迅速得到软件开发商的支持和程序开发人员的喜爱，显示出其强大的生命力。XML 较好地解决了 HTML 无法表达数据内容等问题，使它在政府、金融、证券、邮电、保险、税务、司法、出版以及电子商务等方面得到了广泛的应用。

XML 是一种类似于 HTML 的标记语言，它主要用来描述数据而不是显示数据。XML 的标记不是在 XML 中预定义的，用户必须定义自己的标记。XML 使用文档类型定义（DTD）或模式（Schema）来描述数据，XML 使用 DTD 或 Schema 后就是自描述语言。图 1.1 给出了一个简单的 XML 文档。

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<bookstore>
<book>
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
</bookstore>
```

图 1.1 XML 示例文档

XML 有其自身的优缺点。

XML 的优势之一是它允许各个组织、个人建立适合自己需要的标记集合，并且这些标记可以迅速地投入使用。这一特征使得 XML 可以在电子商务、政府文档、司法、出版、CAD/CAM、保险机构、厂商和中介组织信息交换等领域中一展身手，针对不同的系统，厂商提供各具特色的独立解决方案。

XML 的最大优点在于它的数据存储格式不受显示格式的制约。一般来说，一篇文档包括 3 个要素：数据、结构、显示方式。对于 HTML 来说，显示方式内嵌在数据中，这样在创建文本时就要时时考虑输出格式，如果因为需求不同而需要对同样的内容进行不同风格的显示，则要从头创建一个全新的文档，这样重复工作量很大。此外 HTML 缺乏对数据结构的描述，对于应用程序理解文档内容、抽取语义信息都有诸多不便。

XML 把文档的三要素独立开，分别处理。首先把显示格式从数据内容中独立出来，保存在样式单文件（Style Sheet）中，这样如果需要改变文档的显示方式，只要修改样式单文件即可。XML 的自我描述性质能够很好地表现许多复杂的数据关系，使得基于 XML 的应用程序可以在 XML 文件中准确高效地搜索相关的数据内容，忽略其他不相关部分。XML 还有其他许多优点，比如它有利于不同系统之间的信息交流，完全可以充当网际语言并有希望成为数据和文档交换的标准机制。

当然，XML 作为一个新建立的标准，还有一些不足之处。例如，它在强调了数据结构的同时，语义表达能力上略显不足，如定义了<地址>这样一个标记，如果没有在文档中实际定义内容，用户就无法知道是要表达家庭住址还是 E-mail 地址。另外，XML 的有些技术尚未形成统一的标准，充分支持 XML 的应用处理程序很少，甚至浏览器对 XML 的支持也是有限的。

所以，XML 还并不能完全取代 HTML，毕竟 HTML 是最为方便、快捷的网上信息发布方式。况且 HTML 是描述数据显示的语言，而 XML 是描述数据及其结构的语言，二者在功能上也是截然不同的。

XML 的主要成分包括：

（1）元素。
（2）属性。
（3）指令。
（4）注释。
（5）实体。
（6）CDATA。
（7）命名空间。

1.2.1 元素（Element）

一个元素由一个标识（标签）来定义，包括起始标识和结束标识以及其中的内容（值）。例如：

<author>Peter</author>

注意：在 HTML 中，标识是固定的，而在 XML 中，标识需要自己创建；XML 元素中还可以再嵌套其他元素，这样使相关信息构成等级结构。

Tag（标识）是定义 XML 元素的基本概念，它用来定义元素。在 XML 中，标识必须成对出现，将数据包围在中间。标识的名称和元素的名称是一样的。例如下面的元素：

<author>Peter</author>

其中 author 就是标识，起始标识用<author>表达，结束标识用</author>表达。

XML 元素可以嵌套，被嵌套在内的元素称为上层元素子元素。例 1.1 给出了一个综合的元素示例。

```
<person>
  <sex>female</sex>
  <name>
    <firstname>Anna</firstname>
    <lastname>Smith</lastname>
  </name>
</person>
```

例 1.1　XML 元素综合示例

例 1.1 中包含 5 个元素，其中：

（1）sex、name 是 person 的子元素。

（2）firstname 和 lastname 是 name 的子元素。

（3）name 元素没有值。

根元素

如果一个元素从文件头的序言部分之后开始一直到文件尾，包含了文件中所有的其他元素信息，称之为根元素。一个 XML 文档中根元素有且只有一个。

XML 元素的命名需要符合一定的规范，主要有：

- 元素的名字可以包含字母、数字和其他字符。
- 元素的名字不能以数字或者标点符号开头。
- 元素的名字不能以 XML（或 xml、Xml、xMl）开头。
- 元素的名字不能包含空格。
- 元素名称不能是关键字。
- 元素名字中间不能有冒号。

1.2.2　属性

属性给元素提供进一步的说明信息，它必须出现在元素的起始标识中。属性以名称/取值对出现，属性名不能重复，名称与取值之间用等号“=”分隔，并用引

号把取值引起来。

例如：

<salary currency="US$"> 25000 </salary>

上例中的属性说明了薪水的货币单位是美元。

注意：属性值必须用引号引着，单引号、双引号都可以使用。例如，一个人的性别，person 元素采用以下两种写法：

<person sex="female">

<person sex='female'>

上面的两种写法在一般情况下没有区别，使用双引号的应用更普遍一些。但是在某些特殊情况下就必须使用单引号，比如下面的例子：

<gangster name='George "Shotgun" Ziegler'>

元素的属性可以有多个，每个属性用逗号隔开。

子元素与属性都可以表示一个元素的某些特性，元素的数据可以存放在子元素中，也可以放在属性中，如例 1.2 中的两种写法。

<person sex="female"> <name>Anna</name> </person>	<person> <sex>female</sex> <name>Anna</name> </person>

例 1.2　子元素与属性表示数据的两种方法

例 1.2 的左侧写法中 sex 以属性形式出现，右边写法 sex 作为 person 的子元素出现，两种写法表达的意义相同。但是用属性表达数据受到一些制约，如属性不能包含多个值（而子元素可以），属性也不能描述结构（子元素可以通过嵌套描述复杂的结构）。

另外，由于属性不能附加子元素及属性，所以不容易扩展。如果使用 DTD 作为模式规范时，属性值很难通过 DTD 进行测试。

在描述大量数据与复杂结构时，应尽量采用元素来描述数据；如果要描述元素特殊的性质，如唯一性时，可以采用属性来描述。

1.2.3　指令/处理指令

处理指令给 XML 解析器提供信息，使其能够正确解释文档内容，它的起始标识是“<?”，结束标识是“?>”。

XML 声明就是一个处理指令：

<?xml version="1.0"?>

处理指令还可以有其他的用途，比如定义文档的编码方式是 GB 码还是 Unicode 编码方式，或是把一个样式单文件应用到 XML 文档上用以显示。例如：

<?xml—stylesheet type="text/xsl"href="mystyle.xsl"?>

1.2.4 注释

注释是 XML 文件中用作解释的字符数据，XML 处理器不对它们进行任何处理。注释是用“<!--”和“-->”引起来的，可以出现在 XML 元素间的任何地方，但是不可以嵌套：

```
<!--这是一个注释-->
```

1.2.5 CDATA

在 XML 文档中的所有文本都会被解析器解析，只有在 CDATA 部件之内的文本会被解析器忽略。

CDATA 全称 Character Data，翻译为字符数据。我们在写 XML 文档时，有时需要存放大量的文本、特殊符号，如“<”，这就需要用到 CDATA 语法。语法格式如下：

```
<![CDATA[这里放置需要显示的字符]]>
这里讲的所有文本包含 tag、数据、空格等。
例如：
<![CDATA[
    <AUTHOR   sex="female">
            Peter
    </AUTHOR>
]]>
```

1.2.6 XML 的语法规则

XML 文档使用自描述的和简单的语法。例 1.3 是一个完整的 XML 文档示例。我们主要介绍常见的语法规则，XML 的详细语法规则可以参考 W3C 的文档。

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
  <to>Lin</to>
  <from>Ordm</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

例 1.3　完整的 XML 示例

语法 1　XML 文档的基本结构由序言（Prolog）部分和一个根元素组成。一般的序言包括：

- XML 声明
- 字符集编码方式
- 若干注释
- DTD（或 XML Schema）声明

在例 1.3 中，第一行：

<?xml version="1.0" encoding="ISO-8859-1"?>

就是一个包含 XML 声明以及编码方法的序言。

语法 2 一个 XML 文档只能有一个根元素。

语法 3 一个 XML 文档首先应当是“格式良好的”（Well-Formed）。

语法 2 保证了 XML 文档根的唯一性，语法 3 则保证 XML 文档是可判断的、没有歧义的。

“格式良好的”XML 文档除了要满足根元素唯一的特性之外，还包括：

- 起始标识和结束标识应当匹配，结束标识是必不可少的。
- 大小写应一致：XML 对字母的大小写是敏感的，<employee>和<Employee>是完全不同的两个标识，所以结束标识在匹配时一定要注意大小写一致。
- 元素应当正确嵌套。
- 属性必须包括在引号中。
- 元素中的属性是不允许重复的。

语法 4 XML 文档应该是有效的。

XML 文档的“有效性”是指一个 XML 文档应当遵守 DTD 文件或 Schema 的规定，“有效的”XML 文档肯定是“格式良好的”。

详细的有效性在 DTD 和 Schema 中解释。

语法 5 所有的空标识也必须被结束。

空标识就是标识对之间没有内容的标识，可以看成空元素，比如
、<img>等标识。在 XML 中，规定所有的标识必须有结束标识，针对这样的空标识，XML 中处理的方法是在原标识最后加/。例如，
应写为
。

语法 6 使用 XML，空白将被保留。

在 XML 文档中，空白部分不会被解析器自动删除。

语法 7 使用 XML，CR/LF 被转换为 LF。

使用 XML，新行总是被标识为 LF（Line Feed，换行）。

DTD（Document Type Definition）是用来规定文档语法规则的。这是 XML 结构化查询的基础。一个 XML 文件必须遵守文件类型描述 DTD 中定义的各种规定。例 1.4 是一个 XML 文档中的 DTD。