

Mastering XSLT

XSLT

从入门到

精通

〔美〕 Chuck White 著

王健 王军 等译

- * 以实现为重点，全面讲述了XSLT的文档转换功能
- * 学习将XML转换为HTML、XHTML、PDF和其他格式



电子工业出版社

Publishing House of Electronics Industry

<http://www.phei.com.cn>

Mastering XSLT

XSLT从入门到精通

〔美〕 Chuck White 著

王 健 王 军 等译

電子工業出版社.

Publishing House of Electronics Industry

北京 • BEIJING

内 容 提 要

XSLT是万维网联盟(W3C)最新提出的一种用来转换XML文档结构的说明性计算机语言,随着网络时代的发展,它的应用场合越来越广泛,影响力也越来越大。本书共分四部分,从**XSLT**的基本概念开始,详细讨论了基本知识点、具体技术和实际应用。书中大量的实例将有助于读者准确掌握**XSLT**语言,提高在实际商业环境中的应用和开发能力。

本书适合不同层次的读者。



Copyright©2002 SYBEX Inc., 1151 Marina Village Parkway, Alameda, CA 94501. World rights reserved. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior agreement and written permission of the publisher.

本书英文版由美国SYBEX公司出版,SYBEX公司已将中文版独家版权授予中国电子工业出版社及北京美迪亚电子信息有限公司。未经许可,不得以任何形式和手段复制或抄袭本书内容。

版权贸易合同登记号: 01-2002-2651

图书在版编目(CIP)数据

XSLT从入门到精通/(美)怀特(White, C.)著;王健、王军译.-北京:电子工业出版社,2003.1
(从入门到精通)

书名原文: Mastering XSLT

ISBN 7-5053-8145-8

I. X… II. ①怀… ②王… ③王… III. 可扩充语言, XSLT-程序设计 IV. TP312

中国版本图书馆CIP数据核字(2002)第085829号

责任编辑: 杨 荟

印 刷: 北京天竺颖华印刷厂

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编: 100036

北京市海淀区翠微东里甲2号 邮编: 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 50.875 字数: 1290千字

版 次: 2003年1月第1版 2003年1月第1次印刷

定 价: 83.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换,若书店售缺,请与本社发行部联系。联系电话: (010) 68279077

致 谢

没有前面无数XSLT开发人员极富创造性的努力，本书的形成根本无从谈起。很幸运，自从XSLT语言开创以来我就一直使用着它，因此在我碰到问题时，很快就知道该去哪儿查找答案。

这里有许多人要感谢。首先要感谢许多Sybex的朋友，他们为本书倾注了极大的精力。感谢Tom Cirtin对我的信任，从而使我能够克服完成本书过程中的无数困难。还要感谢Teresa Trego和Susan Berge，他俩的辛苦工作远远超出了标准的工作量，他们应该受到的也远不止这一声简单的感谢。同时还要感谢的是Greg Beamer，他使我保持诚实，并在不可思议的工作量重压之下创造奇迹。特别要感谢的还有Dimitre Novatchev，他不仅很早就开始关注FXSLT，而且允许我为本书完整地复制它的内容。还要感谢Michael Qualls在ASP问题上的指导，感谢Teun Duynstee提供了关于如何注解XSLT的代码示例，Michael Case提供了转换为数学符号的代码。感谢XSL-List成员（这里就不一一列举了），使得所有能够想像到的XSLT问题有了寻求答案的去处。感谢TopXML的Trace Wilson和Marrowsoft的Martin Rowlinson所提供的Xselerator（一种XSLT IDE）早期版本。当然，始终要感谢的是我的经纪人Chris Van Buren。还要感谢Chelsea Valentine对第1章、第2章、第3章、第4章、第5章和第7章所做的贡献。最后尤其要感谢的是我的妻子Eileen，她在整个过程中始终支持着我。

译者序

XSLT是一种用来转换XML文档结构的说明性计算机语言，随着网络时代的发展，它的应用场合越来越广泛，影响力也越来越大。本书从**XSLT**的基本概念开始，在此基础上讨论了知识点、技术和实际应用。全书共分四个部分：第一部分详细介绍了**XSLT**的基础知识，第二部分介绍了**XSLT**实际使用中的不同处理技巧，第三部分讨论了创建**HTML**转换的各种情况，第四部分探讨如何解决特殊的输出问题，每一部分都是在前面章节的基础上由浅入深，循序渐进。

本书涉及内容之广泛，结构安排之紧凑，问题分析之准确，使我们感到收获颇丰。本书作者**Chuck White**深厚的技术功底、独具匠心的构思以及流畅透彻的文字，使得本书具有很高的实用价值。

本书中译本是经过精心策划、翻译和制作出版的精品图书，不同层次的读者都可以从本书获得所需的知识。本书内容和描述极其精炼，完全是作者多年实际经验的积累。书中大量的实例将有助于读者准确掌握**XSLT**语言，提高在实际商业环境中的应用和开发能力。我们相信，你会成为本书热心和忠实的读者。

许多人为本书中译本的形成付出了辛勤的劳动。本书第一部分到第三部分主要由王健翻译，第四部分由王军翻译，丛刚、朱海玲、赵博、张习连翻译了本书的部分内容，宋爱华、周涛、孙永强、王珺、张吉祥、陈旌、刘芳、周小暄和何文等审校了全书。

在翻译过程中，我们本着严谨求实的态度，参考了大量书籍和资料。对于书中出现的大量术语，我们都做了细致认真的推敲和斟酌。但是由于时间仓促，译者水平有限，难免有失误及不当之处，恳请广大读者批评指正，并真诚地希望本书能对你有所帮助。

前 言

从不同的观点来看，XSLT或XSL转换可以说是对XML最好的发展，也可以说是迄今发明的最恐怖的计算机语言之一。不管你是否喜欢它，它已经来到了我们身边，本书就是帮助你学习它、精通它，而且希望你会喜欢它。

XSLT是一种功能强大的语言，它使得将XML文档转换为多种格式成为可能。你可以将该文档转换为另一个XML文档、用于World Wide Web的HTML文档、语音、乃至是视频/多媒体表示格式（例如SMIL）。你还可以将它转换为PDF文件或RTF（Rich Text Format），从而可以用文字处理应用程序（例如WordPerfect和MS Word）来读取它。事实上，你几乎可以将XML文档转换为所有能想像得到的基于文本的其他格式。

XSLT与许多时下流行的其他计算机语言并不相似，所以熟悉一种这类相对更流行的语言并不一定会成为学习XSLT的优势。XSLT无疑与HTML毫不相似，尽管掌握HTML会有助于开发到HTML的转换。XSLT与C++或Java也没有相似之处，因为它并不是面向对象的语言。XSLT就是通常所说的说明性计算机语言。新闻舆论大量报道的大多数计算机语言（例如C++）都是程序性语言。

在阅读完本书之后，希望你并不会在乎XSLT到底是什么。你将在编写程序代码时用到它，它将快速地为你的XML项目开发新的激动人心的应用程序，而一点儿也不必考虑XSLT与其他语言在语义上的区别。

本书读者

只要你在阅读这个前言，本书就可能适用于你。而假如你的工作中要使用到XML，那么本书就肯定适用于你了，因为XSLT是大多数XML开发工作的一个极其重要的组件。可以通过程序方法（例如DOM+JavaScript、XMLPerl、OmniMark和XSLScript）转换XML，它们允许用户编写转换脚本，但是现在XSLT已经压倒性地成为转换XML文档的约定和标准。

假如你对XML一点儿也不了解，那么本书的内容可能会有点儿难以理解。XML知识可以说是使用XSLT的一个先决条件。

假如你是个XSLT初学者，那么你会发现本书提供了一条平缓的学习曲线，可以教会你想学到的相应知识。本书中的许多示例都采用步进式方法，指导你逐步熟悉XSLT应用程序的开发过程。归根结底，本书的目的是帮助你精通这门语言。

假如你曾经接触过XSLT，并且认为它很是枯燥乏味，那么我想你会发现本书将使你更容易理解这门语言。我就有过这样的经历。最初，我也没有一下子喜爱上这门语言，但现在我已经使用它好几年了，而且从1997年8月它最初作为一个提议（www.w3.org/TR/NOTE-XSL）一直到现在，我始终关注着这门语言的变迁。对于我来说，它并不是一门直观的语言。我知道这门语言对于初学者来说可能会很费劲，因此希望本书已经抓住了如何生成XSLT代码的本质，能够提供一种使你迷上它众多魅力的方法。

假如你已经通晓XSLT，你会发现这本书是一个有关XSLT所有内容的综合性资源。本书采用实用性方法，从商业角度介绍XSLT在现实世界中的可能运用。本书的许多示例都来自现实的生产环境，而生成它们的XML源文档也是如此。我本身具有一定的商业背景，因此这里的示例都认真地考虑了其所具有的商业意义。下面的一些问题将在这里得到回答：如何制做表单？如何将XML转换为对数据库更友好的格式？如何创建货单？一直教到你会使用为止。

在阅读完本书时，你将已经学到了有关XSLT的一切内容，从最基本的XSL转换到非常复杂的技巧，所有这些都将证明XSLT是一门成熟的函数型程序设计语言。

XSLT已经发展到了具有许多针对普通问题的标准化解决方案的程度，而这些也都包括在本书之中。例如，已经没有理由再要你自己想出如何创建具有交替改变颜色的HTML表行或如何分组数据的方法。已经有人为你做了这些事，而且这些技巧遍布全书，因此你可以把时间花在自己最擅长的方面——思考。

本书组织方式

本书共分4部分23章，由浅入深地介绍了XSLT的概念、语法，以及围绕你大部分日常生产工作的处理技巧。

第一部分“XSLT基础” 这是一个XSLT指南，内容涉及基本转换过程、样式表结构、模板本质和XSLT/XPath数据模型。它还讨论了XSLT非常重要的“姐妹”语言XPath，并介绍了变量、参数和函数。

第二部分“处理技巧” 从生产环境中实际可能用到的角度，介绍了可以在XSLT中使用的不同处理技巧，包括管理不同种类的输出（例如XML、文本与HTML）、循环与迭代、分组与索引、多文档管理、模块化、排序与编号以及如何生成XSLT文档编制与注释。它还探讨了大量现有的对此语言的扩展以及如何处理反向兼容和正向兼容。

第三部分“生成HTML文件” 涉及创建HTML转换的纷繁难懂之处。你将学到如何生成基本的HTML（包括表单和表格），以及如何应对某些突然出现的意外事件（例如处理特殊字符和编码问题）。

第四部分“特殊输出考虑” 探讨如何为其他的一些XML词汇表（例如SVG、MathML与XSL-FO）开发XSLT，以及如何输出像RTF与PDF这样的非XML置标。你还将精通文本串处理技巧以及特殊字符的处理。

在本书的最后部分，你将看到三个附录。前两个分别涉及XPath资源以及基于Web的资源。第三个附录是对Functional XSLT库的一个介绍，那是针对Dimitre Novatchev所开发的复杂数学变换和字符串处理的一个预开发XSLT解决方案代码库，读者可以在www.sybex.com找到这个代码库以及Dimitre编写的有关如何运用该代码库的一个说明。

此外，你还将在线发现一个针对与XSLT相关的开发与处理工具的丰富资源，以及一个包含公用XSLT模板、例程及任务（例如生成独特关键字的模板、逗号分隔列表、HTML表格等）的代码库。你还将发现一个XSLT/XPath函数以及XSLT元素与属性的集合。

本书约定

本书使用了几个明显的约定来帮助你尽快找到所需的信息。这里显示的提示、说明和警告用来突出重要的主题。

提示：提示将为你提供忠告、简化操作或有意思的信息块，它们将在你的XSLT开发中对你有所帮助。

说明：说明提供笔者想引起你注意的细节。

警告：警告表明应该格外小心，否则会出现一个特殊问题，而该问题很可能会迫使你重新考虑是否继续从事开发工作。

有意思但稍微有点儿偏离主题的信息将显示为补充文本，它应该受到比说明更多的注意。在这类信息中，你可能会看到并不与XSLT直接相关的某个XML词汇表的详细资料、有关当前主题的历史观点或其他辅助信息。

在使用小片段代码时，笔者通常把它称为代码段，不是源片段、XSLT片段或是结果片段，它在书中出现的样子如下所示：

```
<xsl:element>some content</xsl:element>
```

另外，在本书正文中提到元素时都没有包括标志。在描述元素时，将把它称为thisElement，而不是<thisElement/>或<thisElement></thisElement>。

为了节省空间，本书的许多源代码（尤其是XML源文件）都已经过了编辑或删减。读者可以从www.sybex.com下载所有这些源代码清单，它们包含未经编辑的源代码。要从Sybex网站访问这些清单，请利用标题、作者名或ISBN（4094）搜索本书，然后单击Downloads按钮。在接受特许契约后，就可以下载本书中所列的任何代码了。

在代码清单中，延续箭头（➡）表明代码绕到下一行，而假如你断开代码行的话，转换或脚本就会失败。

准备开始

XSLT是一门功能强大的语言，但是它可能对我们中的许多人来说都有点儿陌生。它的一些元素是自解释性的，但是该语言的纷繁难懂之处当然不会那么显而易见。有时你会很吃惊地发现，它竟然能够令人难以置信地轻松创建功能强大的交互式应用程序。而在其他时候，进行最简单的转换也可能会使你的精神受到创伤。

老实说，假如你试图通过领会笔者以及其他对本书做出贡献的开发人员所做的事来学习XSLT，然后简单地把它应用到自己的应用程序中，那么你会取得有限的成功。XSLT并不能很好地适合于靠领会其他人思想的开发人员。它非常灵活，因此要想发挥它的最大功效，需要学会独立地思考问题。不论你找到多少极好的模板，你自己的应用程序总是会对它们有些特殊的要求，而这些要求又是别人已开发的模板所不能满足的。这里的关键始终是根据别人的成果来完成自己的任务。

本书旨在帮助读者考虑XSLT解决方案。XSLT具有非常多的规则，陷入它们对于那些不是整天与XSLT打交道的人来说就是浪费时间（而那些整天与XSLT打交道的人通过每天使用这些规则也将最终完全学会它们）。相反，笔者希望读者能从本书获得的是对于如何使用XSLT的一些洞察力。你将学到XSLT它自己是如何思考的。

本书的目的是使这门语言易于理解并充满乐趣。假如我成功了，而且你想要告诉我的话，我非常愿意收到你的来信。假如在某些方面笔者尚有不成功之处，而且你想要告诉出版社或我的话，我恳请你这么做。它在下一个版本就会得到改进。你可以通过 chuck@tumeric.net 与我联系，告诉我你对于本书的想法，不管它是对你有所帮助，还是你觉得有些地方需要改进。本书是无数个小时的汗水和研究的成果，再怎么强调都不足以说明它是多么值得你一读。你还可以访问www.tumeric.net或www.sybex.com，获得有关此书的最新信息及勘误表。

享受你与XSLT的摔跤比赛，并让我知道你们的战况！

目 录



第一部分 XSLT基础	1
第1章 XSLT的作用	1
历史简介	1
XSL标准	2
什么是转换	4
为什么要转换XML	5
XPath数据模型	6
输出方式	7
多文档处理	13
处理XSLT	20
更多信息	28
小结	28
第2章 样式表结构	30
合格的XSLT对象名称	30
xsl:stylesheet元素	33
理解最高级元素	34
什么是模板	35
简化样式表语法	47
正向兼容性	47
可扩展性	48
小结	48
第3章 XSLT模板	50
本章的一个源程序清单	50
浏览文档树	52
包含模板的元素	68
定义模板规则	69
应用模板规则	90
解决模板规则冲突	93
内置模板规则	94
最后一个for-each语句	96
保持模块化	99
更多信息	103
小结	103

第4章	XSLT数据模型	104
	XPath形成基础	104
	XPath树形模型	107
	文档顺序与XSLT	118
	XSLT中的空白处理	118
	XML信息集	121
	XPath展望	123
	更多信息	124
	小结	124
第5章	表达式	126
	表达式在XSLT中的作用	126
	XSLT与表达式	128
	表达式数据类型	130
	位置路径	133
	使用缩写的XPath语法	165
	小结	167
第6章	XSLT中的变量与参数	168
	理解xsl:variable元素	168
	理解xsl:param元素	176
	使用xsl:with-param通过结果树片断存储数值	177
	参数范围	179
	使用变量值与参数值	180
	更多信息	182
	小结	183
第7章	函数	184
	函数概述	184
	XPath函数	187
	XSLT专用函数	202
	小结	205
第二部分 处理技巧		207
第8章	输出管理	207
	理解xsl:output元素	208
	理解输出方式	210
	输出格式良好的XML	211
	输出HTML	221
	输出文本	225
	更多信息	230
	小结	230

第9章	循环、迭代与条件控制	231
	使用循环与迭代	231
	理解递归	232
	嵌套迭代	253
	通过xsl:if管理条件控制	254
	通过xsl:choose管理条件控制	257
	更多信息	262
	小结	262
第10章	分组与索引	263
	分组的局限性	263
	以位置分组	264
	以内容分组	279
	使用关键字建立分组变量	280
	使用generate-id()函数进行索引	290
	通过分组创建行	292
	多级分组	302
	XML Spy工作流程：转换和分组Word文档	307
	XSLT 2.0分组功能前瞻	314
	更多信息	314
	小结	315
第11章	多文档管理与模块化	316
	嵌入样式表	316
	使用多个源文档和XSLT文档	318
	使用通用模板调用函数	330
	选择合并文档中的不同节点	335
	输出文档	340
	合并不同文件的XML数据	344
	有条件地合并文档	346
	合并两个文件并求和输出	357
	疑难解答	361
	更多信息	362
	小结	362
第12章	排序与编号	364
	排序	364
	编号	384
	小结	402
第13章	生成文档编制与注释	403
	生成基本的注释	403
	生成自编制代码	407

利用XSLdoc生成文档编制	431
更多信息	432
小结	432
第14章 XSLT扩展	433
为什么要扩展XSLT	434
EXSLT简介	434
使用扩展	434
扩展元素	436
扩展属性	457
扩展函数	459
更多信息	478
小结	480
第15章 XSLT中的低效运行	481
为什么需要fallback	481
使用xsl:fallback元素	482
排序fallback指令	484
使用element-available()函数	489
使用function-available()函数	492
更多信息	494
小结	494
第三部分 生成HTML文件	495
第16章 生成HTML	495
生成基本的HTML文件	495
处理空元素	510
HTML元素和属性中的大小写处理	512
HTML作为输入	513
利用属性值模板生成图像	514
为输出选择和分组元素	515
使用JavaScript和VBScript	516
使用XHTML	518
更多信息	528
小结	529
第17章 生成表格	530
生成基本表格	530
改变表元与表行颜色	548
多列排列数据	555
提取一行数据	561
调用模板创建表格	565

将数据分组装入表格	568
更多信息	575
小结	575
第18章 使用表单	576
创建下拉列表和复选框	576
创建可交互的表单	583
建立链接管理系统	610
更多信息	616
小结	617
第19章 HTML: 特别考虑	618
编码和XML	618
处理脚本和样式元素内容	633
管理换行	637
更多信息	644
小结	644
第四部分 特殊输出考虑	645
第20章 输出字符串和特殊字符	645
字符串和XPath	645
分析URL字符串	646
在x个字符后换行	654
统计字符串中的字符	657
将数字转换为字符串	658
大小写处理	667
理解特殊字符	668
使用引号和撇号	674
拆分字符串成元素	675
删除空白	678
更多信息	679
小结	679
第21章 数学计算	680
生成随机数	680
基本的XSLT计算	685
生成合计	686
减法、除法以及其他基本运算	691
创建各部分的小计	695
使用十六进制数和转换	700
查找最小和最大值	702
将RGB颜色值转换为十六进制	704

	生成MathML输出	709
	更多信息	710
	小结	710
第22章	使用XSLT输出SVG	711
	使用XSLT整合SVG文档	711
	使用XSLT和SVG创建一个图表	721
	使用XSLT将SVG嵌入HTML	733
	使用SVG和ASP	735
	设置MIME类型	735
	更多信息	736
	小结	738
第23章	生成RTF和其他非XML置标	739
	生成RTF	739
	产生CSV文件	746
	使用XSL-FO来创建RTF和PDF文档	747
	更多信息	756
	小结	757
附录A	XPath	758
附录B	XML/XSL资源	784
附录C	使用XSLT进行函数程序设计	794

第一部分 XSLT基础

第1章 XSLT的作用

可扩展样式表语言（XSL, Extensible Stylesheet Language）最早由万维网联盟（W3C, World Wide Web Consortium）于1999年提出。起初，这一新型的XML样式表语言包含1个标准，该标准定义2个单独的处理过程。由于XSL建模在文档样式语义和规范语言（DSSSL, Document Style Semantics and Specification Language）之后，所以它定义了如何转换和表示XML文档。其中第1个处理过程是转换，定义如何操作文档。第2个处理过程是表示，定义如何在不同的用户代理（例如打印机或浏览器）中显示XML文档。

经过一定的发展，XSL工作组发现，有时会需要使用上述处理过程中的1个或2个过程。他们发现这2个处理过程之间有一个明显的区别，也正是这一区别最终导致它们与XSL标准的分离。在验证该区别之后，XSL工作组便把XSL标准划分成2个单独的规范文档：可扩展样式表语言转换（XSLT, Extensible Stylesheet Language Transformations）和XSL。

本章主要包括以下内容：

- 探讨XSLT的作用
- 理解转换过程
- 介绍XSLT模板
- 用MSXML、Xalan和Saxon处理XSLT

历史简介

要理解XSLT，首先必须理解它在XSL标准中的位置。XSL含有大多数核心的XML标准，由W3C技术人员定义和维护。虽然它还只是一个相当初期的标准，但是它的起源却比W3C还要早。正如XML是由标准通用标记语言（SGML, Standard Generalized Markup Language）发展而来一样，XSL建立在对DSSSL多年工作的基础上。这里的DSSSL是一种复杂的标准化样式语言，由国际标准化组织（ISO, International Organization for Standardization）提出。

说明：本书通篇使用“标准”一词，因为笔者已经向不正确的术语屈服。虽然大家都喜欢到处乱抛“标准”一词，但W3C倒更像是一个推荐委员会，而不是一个标准委员会。作为一个会员团体，他们多少都会达成一些共识，但最终的结果却并不是真正的标准，因为他们无法强制推行自己的建议。

20世纪90年代初，DSSSL作为一种国际标准被提出，用于处理SGML文档。SGML¹与XML相似，允许通过处理（通常是格式化）完整地将文档内容（数据）分开。可以这么说，含有嵌入式代码的Microsoft Word文档描述的是文档外表特征（也就是打印机应该怎么“处理”它），而SGML文档则只是描述文档的结构。

DSSSL允许以一种标准方式来描述文档的处理过程。通常，文档处理最常见的2种方式是格式化和转换，而DSSSL便是第1种标准化这2种文档处理的语言。

提出一种新的样式表语言，而不再坚持DSSSL，并不是一个故意为难用户的武断举动。创建这一新语言是有一定原因的。DSSSL是为SGML而不是XML创建的，因此它不是一个正式的XML词汇表，也不遵循XML的语法规则。在最初作为Note向W3C（www.w3.org/TR/NOTE-XSL）提议时，XSL是

“打算通过向大多数数据描述和显示需求提供一个说明性解决方案，以使其更易于为‘置标’级的用户所理解”。

版权©1997万维网联盟，（麻省理工学院，Institut National de Recherche en Informatique et en Automatique，Keio大学）。版权所有。<http://www.w3.org/Consortium/Legal/>。

此文档是W3C讨论专用的NOTE。

这就意味着，用户必须能够运用此语言把一种XML文档转换为另一种XML文档，或者转换为文本、RTF（Rich Text Format）或PDF。事实上，从理论上讲，用户必须能够把任何一种XML文档转换为其他类型的文本文档或置标。

XSL标准

自首次提出以来，XSL标准已经有了一定的发展。要想正确使用XSL，最初只需要知道1个规范文档，而现在则是3个规范文档。这3个规范具体如下：

XSL 一种高级的格式化语言，用于定义如何显示数据。可以在www.w3.org/TR/xsl处找到XSL 1.0推荐版本。

XSLT 一种转换语言，用于提供一套规则，把XML数据转换为另一种XML文档、HTML文档或其他文本格式，例如可移植文档格式（PDF，Portable Document Format）和Java代码。可以在www.w3.org/TR/xslt处找到XSLT 1.0推荐版本。

XPath 一种与XSLT截然不同的表达式语言，它为识别和寻址XML数据提供正规的语法。可以在www.w3.org/TR/xpath处找到XML Path Language 1.0推荐版本。尽管XPath是一种独立于XSLT的语言，但是XSLT对XPath却有着核心性的依赖，因为XSLT置标规则建立在XPath结构基础上。

说明：实际上，在XSLT脱离XSL之后，XPath曾经只是XSLT的一部分。这是因为有人坚信，XPath可以用来开发其他XML操作语言和处理方法，例如XQuery。用户会在不同场合看到XPath的使用，包括像SQL Server 2000和Oracle这些时兴的数据库，以及为数众多的Web脚本语言。