XSLT 从入门到精通

陈建勋 编著

- 是一本详实、全面的XSLT参考书籍
  兼具学习、应用、检索三重特点，是XML设计师不可缺少的工具书
  也是任何想掌握XSLT的初学者最佳的引路明灯
  如果你的电子商务环境以XML为主体，更是不可不知XSLT

- XSLT指的是XSL(T)转换语言，是XML家族技术成员之一
  可视为处理XML数据的高级语言，目的是为了转换文件的结构
  XSLT可以配合XSL-FO格式对象转换出FO文件，对各种文件或各类图书进行排版
  也可以独立于XSL-FO格式对象之外，转换任何XML语言写成的XML文件

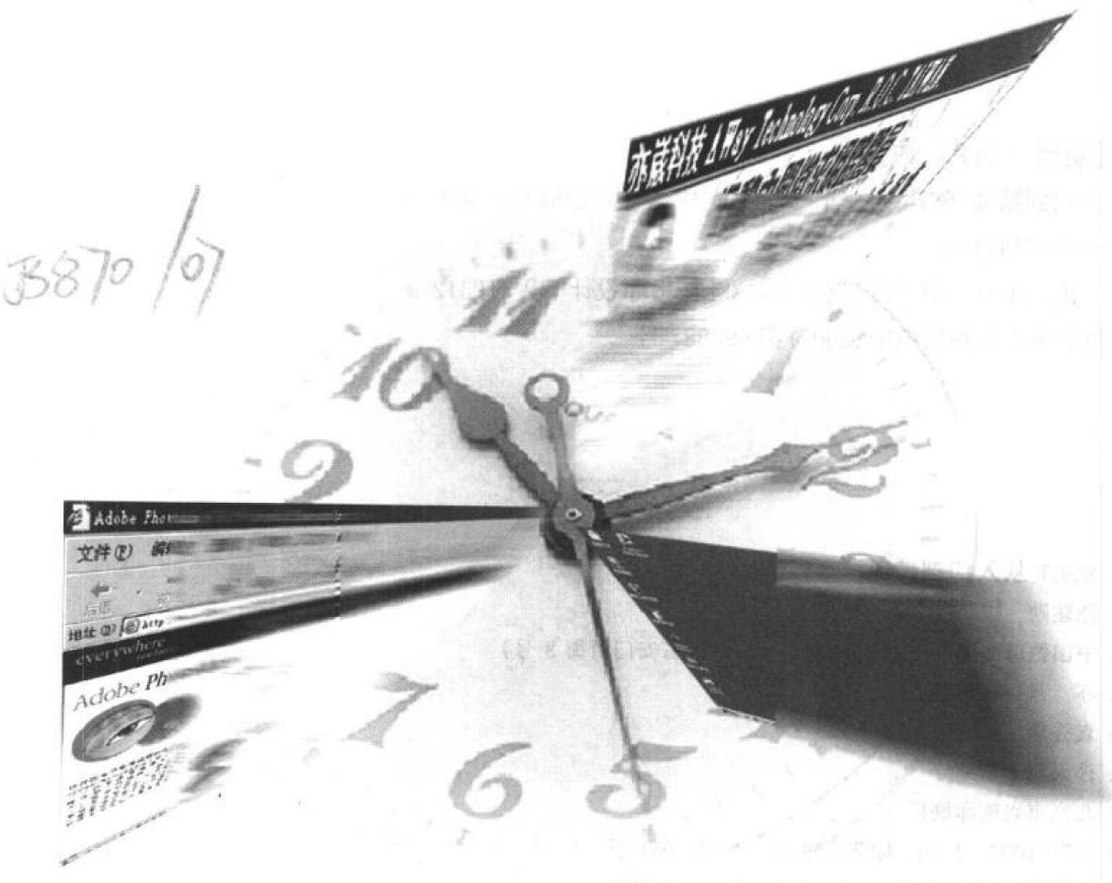
- 这本书是XSLT转换语言的语法大全，讨论的主题如下
  XSLT入门指引
  XSLT组件完整介绍
  函数完整介绍
  对比样式详解
  XPath表达式详解
  XSLT和互联网结合实例探讨
  网页排版系统
  通用统一字符编码

内附本书范例光盘

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

XSLT 从入门到精通

陈建勋 编著



中国铁道出版社

2002·北京

（京）新登字 063 号

北京市版权局著作权合同登记号：01-2000-5295 号

版 权 声 明

图书在版编目（CIP）数据

XSLT 从入门到精通/陈建勋编著. —北京：中国铁道出版社，2001.12

ISBN 7-113-04471-9

Ⅰ.X… Ⅱ. 陈… 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(2001)第 093887 号

书　　名：XSLT 从入门到精通

作　　者：陈建勋

出版发行：中国铁道出版社（100054，北京市宣武区右安门西街 8 号）

责任编辑：苏　茜　郭毅鹏

特邀编辑：袁秀珍

封面设计：孙天昭　董默峥

印　　刷：北京市兴顺印刷厂

开　　本：787×1092　1/16　印张：39.25　字数：961 千

版　　本：2002 年 1 月第 1 版　　2002 年 1 月第 1 次印刷

印　　数：1～4000 册

书　　号：ISBN 7-113-04471-9/TP·656

定　　价：62.00 元

出 版 说 明

本书共分为两个部分：第一部分是带领读者走进 XSLT 世界的教学课程，着重探讨 XSLT 转换语言各种构件的用法，以及 XSLT 转换语言在互联网上的实务应用，可视其为入门教材。第二部分是全书精华之所在，完整分析 XSLT 和 XPath 组件的各项细节和彼此间的互动关系，是 XML/XSLT 设计师不可缺少的参考资料。

本书所附光盘内容分为两部分：一部分为在 Windows 环境下运行的程序范例；一部分为在 Linux 环境下运行的程序范例。由于时间匆忙，在 Linux 环境下运行的程序范例为繁体版，若出现乱码，请用“东方快车”等汉化软件进行转换，敬请读者谅解。

本书由台湾博硕文化股份有限公司提供版权，经中国铁道出版社计算机图书项目中心审选；王秀平、梁秀玲、关超、史广顺、童冠圣、陈辑超、马超、杨小平、杨小军、段小明、杨军、陈贤淑及孟丽花等同志参与了本书的整稿及编排工作。

2002. 1

目　录

第一部：XSLT 学习之旅

目录

第二部：XSLT 参考文献

目录

目录

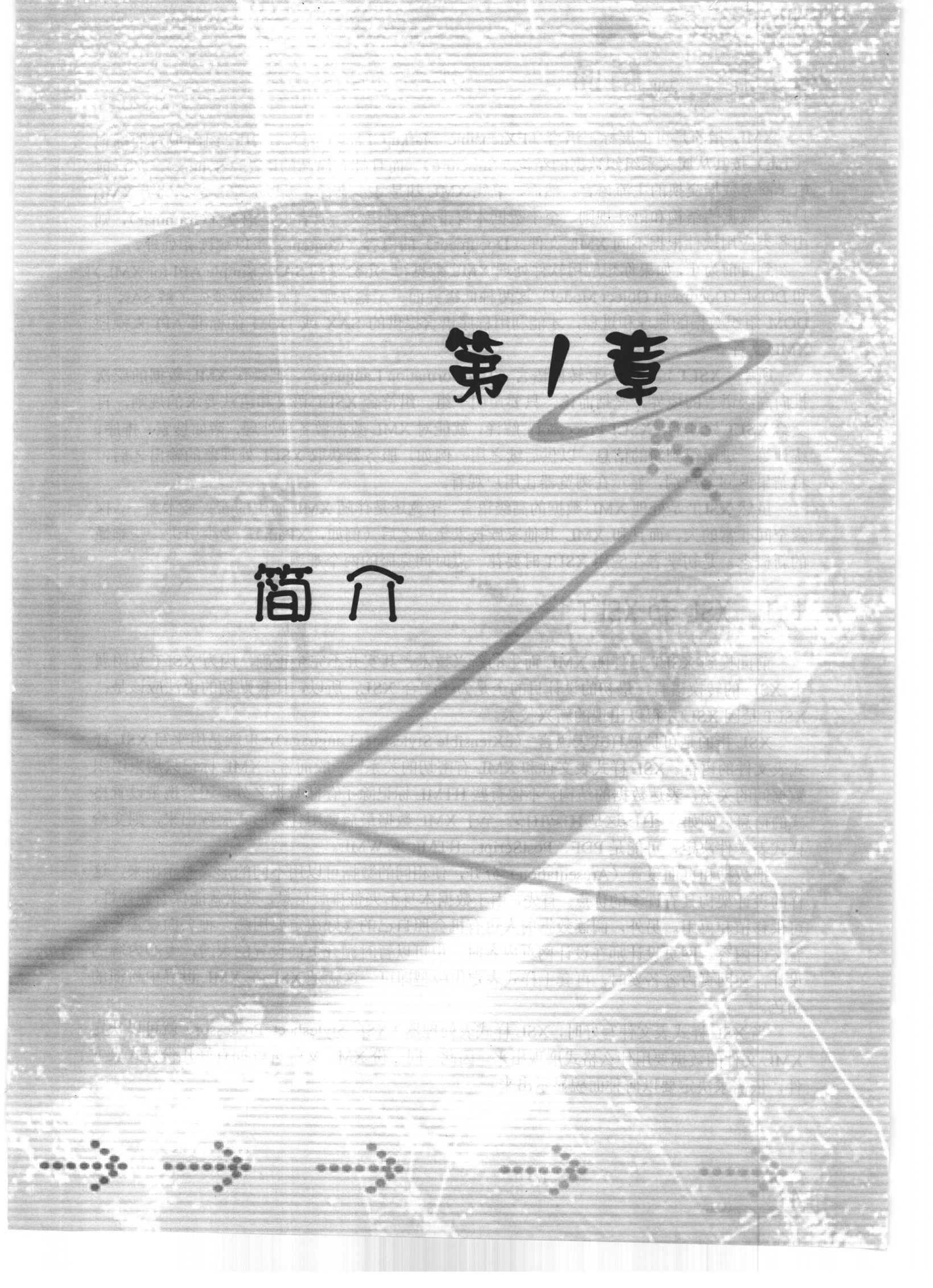
第一部着重探讨 XSLT 转换语言各种构件的重点用法（含 XPath），以及 XSLT 转换在互联网上的实务应用。核心目标是希望读者读完第一部之后，就有能力自行设计和撰写 XSL 模板文件，从而领会 XSLT 这种新兴技术的实用潜力。

第一部的各章各节并没有详列各种构件的相关细节，这就是为什么这一部的标题是“XSLT 学习之旅”的原因。偶尔谈到一些难以一笔带过的细节时，会出现比较严谨的论述，这是为了避免初学者误解其意而采取的必然措施；读者宜耐心研读。

第一部共计 6 章，章名次序如此：第 1 章<<简介>>，第 2 章<<模板结构>>，第 3 章<<模板规则>>，第 4 章<<设计模板规则>>，第 5 章<<XSLT 处理器>>，第 6 章<<网页排版系统>>。

XSLT 学习之旅

第一部



第1章

简介

XML 指的是可扩展标记语言（EXtensible Markup Language），运用一组组成对的标记（Tag）按其从属关系组织数据的结构（Structure），而且拥有简洁的语法（Syntax），是目前电子商务数据交换的主流语言。然而，单凭 XML 却是寸步难行、无以成事，这是因为 XML 只定义数据的结构和语法规则，至于数据本身带有何种语义，或者说是语法（Grammar），则由各类应用软件根据不同 XML 文件（Document）的内容（Content）予以解读和处理。

到目前为止，如果你想应用软件处理 XML 数据，一定会碰到 SAX（Simple API for XML）和 DOM（Document Object Model）这两种低级界面。无论是哪一种，你必须先了解 SAX 或 DOM 组织数据的原理和过程，才能利用程序语言提供的 SAX 或 DOM 的标准 API 来操控 XML 文件。

所幸，XSLT 定义了一套转换语言（Transformation Language），把操作 XML 数据的层次推向高级语言的境界。概括而论，只要声明一组一组符合 XSLT 语义和语法要求的规则，再交给 XSLT 处理器（XSLT Processor）解译，就能对 XML 数据做各种处理，诸如搜索、排序、统计等，进而得到新的信息，以供后续之用。例如，服务器获取 XSLT 处理器的输出之后，将结果返回客户端，显示在浏览器让用户观看。

虽然 XSLT 是处理 XML 数据的高级语言，毕竟还是伴随 XML 而生的新兴技术之一，探索空间非常的大，而且和 XML 其他家族技术集成之后（例如，XPath），多少引进一些难懂的新概念，是初学者在应用 XSLT 时要花一点时间克服的两个难题。

1.1 XSL 和 XSLT

前面提到 XSLT 是伴随 XML 而生的新兴技术，其实并不完全正确，因为 XSLT 是附属于 XSL 的转换语言，最初的设计目的全是为了配合 XSL，所以，比较贴切的说法应该是：XSLT 是因 XSL 才得以出现的新兴技术。

XSL 指的是可扩展样式表语言 （eXtensible Stylesheet Language），也就是用来写 XSL 样式表文件的语言。XSL 样式表文件和 XML 有密切的关系。一般而言，XML 标记只用来组织数据间的关系，表达数据的结构，不像有些 HTML 标记除了存储数据之外，还会携带设置格式的信息，例如，<H1>这是 H1</H1>。至于 XML 数据最后要用什么形式展现出来，则交给样式表文件决定，可能是 PDF、PostScript、HTML 或 WML。

把数据和页面设置（Aresentation）分开，使相同的数据可以用不同的形式显示出来，这样的设计架构具有许多的优点。首先，由于数据本身不夹带排版信息，数据就能保有纯粹性，可供移植和再用。另外，阅读数据的人也有机会照自己的喜好决定要用哪一种页面设置来观看文件内容。网页设计师在设计网页内容时，也可以暂时抛开页面设置的束缚，专心内容的设计，在网页内容备妥时，再套上样式表定出版型即可。这就是 XSL 在 XML 世界中扮演的角色。

当 XSL 样式表文件写好时，XSL 样式表处理器（XSL Stylesheet Processor）就可以决定 XML 文件的数据要用什么格式展现出来。因此，同一份 XML 文件可以拥有好几份样式表文件，依需求决定要以何种面貌显示出来。

1.1.1 XSL 的处理流程

XSL 样式表处理器解译 XSL 样式表文件的过程在概念上可分为两个步骤：

- 第一阶段：结构树转换（Tree Transformation）
- 第二阶段：设置格式（Formatting）

第一阶段在转换结构树时，XSL 样式表处理器会用到 XSL 转换器（Transformer），而第二阶段在设置格式时，XSL 样式表处理器会用到 XSL 格式器（Formatter）。在结构树转换阶段，XSL 转换器会建造三棵结构树（Tree）：一是来自 XML 文件的来源树（Source Tree），二是来自 XSL 样式表文件的样式表树（Stylesheet Tree），三是从来源树和样式表树得到的结果树（Result Tree）。得到结果树之后，就进入设置格式的阶段。XSL 格式器会解读结果树，根据结果树中的格式对象（Formatting Object，FO）决定数据的排版格式，最后得出结果，诸如 PDF 文件（.pdf），或 PostScript 文件（.ps）。XSL 的格式对象统称为格式语言（Formatting Language）。格式对象最初是放在 XSL 样式表文件中，通过 XSL 转换器的转换，将格式对象置入结果树中，XSL 格式器再解读格式对象携带的信息做出该有的排版效果。我们可以用图 1.1 的流程图来说明这两个阶段。

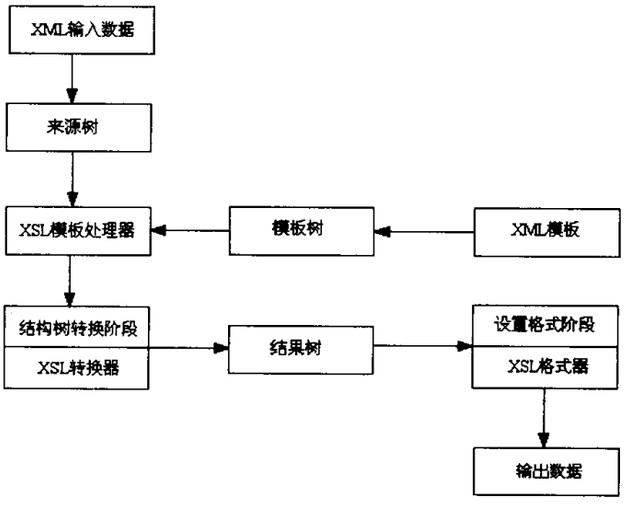


图 1.1　XSL 样式表处理器的处理流程

这是 XSL 样式表处理器标准的分解过程，就实务而言，这两个阶段不必分得如此清楚，也可以合成一个步骤，在转换的同时就把该有的格式设计好。另外，这两个阶段也可以拆开来分别进行，也就是 XSL 转换器和 XSL 格式器彼此独立毫不相干。换句话说，XSL 样式表文件中不见得要放格式对象，这是因为结果树不见得会交给 XSL 格式器设计排版。由于这两个阶段的工作可以分开来设计，而且目前设置格式阶段的规范并未臻于完善，因此，很多 XSL 工具都只有针对第一阶段的结构树转换，设计出符合规范要求的 XSL 转换器。

XSLT 出场的时刻就是第一阶段的结构树转换。XSLT 指的是可扩展样式表转换语言（EXtensible StyleSheet Language: Transformation），简称 XSL（T）转换语言，目的就是让 XSL 转换器把来源树按照样式表树的意志转换成结果树；转换的过程完全是结构树对结构树的关系。

由上述讨论可知，XSL 可分为 XSL 转换语言和 XSL 格式语言两种，分别在结构树转换阶段和格式设置阶段起关键性的作用。虽然 XSL 样式表文件中可以同时含有 XSL 转换语言和 XSL 格式语言的组件（element），然而这本书探讨的是 XSLT，将 XSLT 视为独立的技术，因此，书中不会触及 FO 的部分。

一般而言，只针对结构树转换阶段设计的 XSL 转换器就另称为 XSLT 处理器，而 XSLT 处理器使用的模板则仍称为 XSL 模板，然而，由于其目的仅止于结构的转换，因此，称为 XSLT 转换模板或 XSLT 模板也未尝不可。

此外，图 1.1 展现出来的 XSL 处理流程中最重要的共同性质就是所有牵涉在其中的文件都是 XML 文件。输入数据是 XML 文件，XSL 模板也是 XML 文件，而输出的数据也是 XML 文件。因此，无论在哪个阶段，你接触的都是 XML，这样便形成了自给自足的系统。在编辑 XML 时，只需用同一套编辑工具即可；在运用 XML 时，也只需 XML 的知识。

1.2 XSLT 的用途

上一节是让读者了解 XSLT 在 XSL 排版设计中所扮演的角色。如果 XSLT 要和 XSL 搭配使用，则格式对象和 XSL 转换语言一定会出现在 XSL 样式表文件中。但是，如果 XSLT 要独立使用，XSL 样式表文件中只需有 XSL 转换语言，即可达到转换的目的。所以，到了这里，读者可以暂时忘记 XSL 格式语言了。也许你会问，既然不考虑排版的问题，单凭 XSLT 又能转换出什么？要探讨这个问题，就让我们先从 XSLT 的定义着手。

1.2.1 XSLT 的处理流程

根据 XSLT 1.0 规范的定义，XSLT 是一种转换语言，可以把 XML 文件转换成别的 XML 文件。XSLT 处理器的处理流程可用图 1.2 来说明：

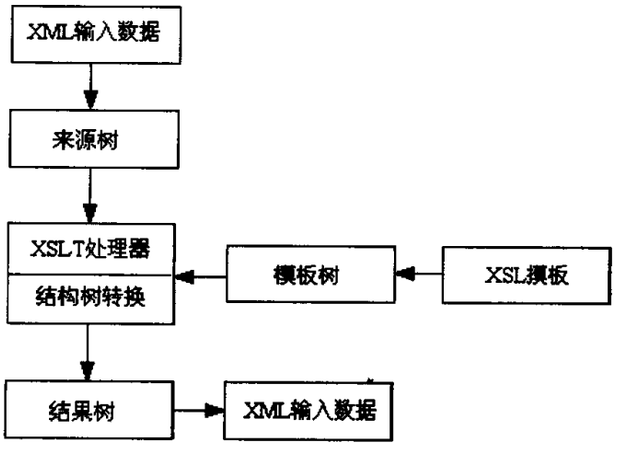


图 1.2 XSLT 处理器的处理流程

XSLT 规范的定义是从表象来说的，其实从一种 XML 数据转换成另一种 XML 数据的过程中，无论输入还是输出，都是 XML 数据。然而，两者间最足以道称的差别在于转换前和转换后的数据结构可以相差非常的大。把握这个核心概念，XSLT 真正的目的其实就是用于