

Java 开发指南

——应用XML和JSP开发电子商务



〔美〕 Bill Brogden 著
Chris Minnick
邱仲潘 等译

建立诱人的Web站点
开发电子商务



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

Java

Java Developer's Guide to E-Commerce with XML and JSP

Java开发指南——应用 XML和JSP开发电子商务

〔美〕 Bill Brogden 著
Chris Minnick

邱仲潘 等译

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 提 要

本书深入浅出地向Java开发人员介绍了如何在Java程序中使用XML的技术,其作者没有展开理论论述,而是通过介绍一个商业化Web站点使用的实用程序,使读者通过实例了解如何生成结构合理的XML,如何生成文档类型定义,如何在XML设计中使用元素、属性与实体,如何用Java操纵文档对象模型,如何用Java集合搜索XML数据,如何用SAX处理大量XML文件,以及如何建立具有吸引力的商业化Web站点,以成功地开展电子商务。

本书非常适合开发人员阅读。



Copyright©2001 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。

图书在版编目(CIP)数据

Java开发指南——应用XML和JSP开发电子商务/(美)布朗登(Brogden, B.)著;邱仲潘等译.—北京:电子工业出版社,2001.5

书名原文:Java Developer's Guide to E-Commerce with XML and JSP

ISBN 7-5053-6697-1

I. J… II. ①布… ②邱… III. Java语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2001)第028875号

书 名: Java开发指南——应用XML和JSP开发电子商务

著 者: [美] Bill Brogden, Chris Minnick

译 者: 邱仲潘 等

责任编辑: 李 莹

印 刷 者: 北京天竺颖华印刷厂

装 订 者: 三河金马印装有限公司

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

北京市海淀区万寿路173信箱 邮编: 100036 电话: 68279077

北京市海淀区翠微东里甲2号 邮编: 100036 电话: 68252397

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 20.75 字数: 530 千字

版 次: 2001年5月第1版 2001年5月第1次印刷

书 号: ISBN 7-5053-6697-1
TP·3737

定 价: 35.00元

版权贸易合同登记号 图字: 01-2001-0416

凡购买电子工业出版社的图书,如有缺页、倒页、脱页,请向购买书店调换,若书店售缺,请与本社发行部联系调换。

献给妻子Rebecca多年来一如既往的支持。

Bill Brogden

致 谢

感谢下列人员的帮助：

感谢妻子Rebecca多年来一如既往的支持。

感谢LANWrights公司的同事Ed Tittel与Dawn Rader提供的指导与编辑经验。

感谢Toivo Lainevoori提供了第5章和第6章的内容。

最后，感谢Sybex公司的朋友们让本书得以问世。

Bill Brogden

感谢Ed Tittel与William Brogden提供的指导与经验。感谢Dawn Rader的编辑。特别感谢我的家人（你们知道你是谁）、妻子Margaret（也知道她是谁）和Roger Smith（有时知道他是谁）多年来给我的鼓励与帮助。

Chris Minnick

译 者 序

本书翻译过程中得到了周阳生、刘文红、邹能东、彭振庆、黄应生、黄志坚、李耀平、江文清等同志的大力帮助，刘文琼、温连英、邱冬金等同志完成了本书的录入工作，刘云昌、刘联昌兄弟帮助进行了书稿与打印稿的校对，在此深表感谢。

前 言

多年前，我（**Bill Brogden**）参加了一次环境数据管理会议。这个会议的目的之一是解决科学家交换数据的巨大难题，特别是在不同国家之间。另外，在不同计算机品牌之间、不同磁带格式与数据存储格式之间，任何研究人员协调来自不同源的数据时，都会感到这是个巨大难题。

在这次会议上，一个环境部门的领导（具有文学方面的背景）指出，他很奇怪，为什么科学数据磁带中没有描述磁带内容的内置信息。如果无法从不同源取得详细信息，则数据磁带毫无用处。我们的科学家知道这种情形很不理想，但不知如何是好。

让数据用普通语言描述自己的思路就是现在的XML。扩展标记语言（XML, Extensible Markup Language）自1998年由万维网联盟（W3C, World Wide Web Consortium）首次发布以来，已经得到广泛接受，表明科学与工业的许多领域早已经向往这种数据描述的变革。

XML迅速普及的原因之一是Internet令如此众多的人们熟悉了HTML，这个Web页面的标准标记语言。HTML让用户学会通过标注用普通文本增加额外信息。有了这个基础之后，在数据描述中使用XML标注就只是一步之遥了（但却是不同寻常的一步）。

XML最明显的优点是用户可以根据特定任务的需要扩展这个语言。这在特定领域需要面对各种数据类型时就特别有吸引力。第二个优点不那么明显，就是不管增加何种扩展，XML工具都可以继续验证与处理数据，只要符合XML语言规则。

实践证明，这种灵活性正是许多实用应用程序所需要的。公司需要与其他公司交流数据时，可迅速把旧的电子数据交换（EDI, Electronic Data Interchange）格式换成XML。这种所谓的公司对公司（B2B, business-to-business）应用将在Internet中占有大量的XML编码数据。

Java与XML用户有许多感兴趣的Web站点。关于参考材料与新闻，见oasis.oasis-open.org。这个站点属于结构化信息标准改进组织（Organization for the Advancement of Structured Information Standards），这是全球最大的XML应用标准化方面的独立非盈利机构。

Java与XML

如果你对Java感兴趣，则可能遇到了许多XML方面的新闻，因为Java与XML从一开始就关系密切。**Bill Brogden**是推动XML标准化的主力之一，他在Sun公司工作，所从事的XML分析器方面的大量早期工作都是用Java完成的。

但是，XML普及工作遍及整个计算机行业，包括IBM与Microsoft之类的大公司和各种非正式工作组。个人和小组同样完成了许多重要推动工作。例如，“XML Deviant”小组是用Internet邮件清单组织起来的，负责开发SAX（XML简化API, Simplified API for XML）。这个项目的开发工作最初是用Java完成的。

Java Servlet与JavaServer Page

本书用Java Servlet与JavaServer Page实现XML数据基于Web的应用。Servlet可以用原先由CGI脚本完成的功能扩展Web服务器。JSP可以在同一页面中组合HTML标记的元素和Java代码段。Servlet与JSP技术的发展和XML一样迅速。

本书创作期间，目前的Java Servlet API为v2.2，最后版本是1999年12月推出的。这个API在旧版基础上作了许多改进，特别是在集成Web应用程序和使用XML的标准化配置方面。

JavaServer页面的标准化技术进行得慢得多。API在0.92版中停留了相当一段时间。由于Sun公司缺乏明确的发展方向，许多软件公司开发了自己的技术，将HTML中的自定义标注与数据库搜索之类的功能组合起来。

随着Sun公司开发出Java 2 Enterprise Edition，JSP在整个Web应用项目中占有重要地位，因此API开发项目迅速升到v1.0，然后又升到v1.1。随着Java 2 Enterprise Edition的推出，Java Servlet与JavaServer Page很好地集成到Java Web应用程序开发的大框架中。J2EE的另一重要特性是大量利用XML描述Web应用程序。

1999年10月发生了一个重大事件。Sun公司将所有Servlet 2.2与JavaServer Page 1.1代码交给Apache组织进一步开发，作为开放源代码Java Servlet与JavaServer Page引擎的参考实现方法。这个发展独立于J2EE规范，保证Servlet与JSP API总是有免费而最新的实现方法。

本书的读者对象

我们假设你对HTML Web服务器和Java语言有基本了解，你的计算机能够运行小型Web浏览器、标准Java编译器和其他实用程序。XML数据的编辑可以用任何文本编辑器或用支持XML的商业化和开放源代码编辑器。

除了这些要求外，任何处理各种数据的人都可能是XML用户。人类生活的各个方面都在Internet上有一些XML相关项目。

需要什么

我们在许多项目中使用的小型Web服务器是免费的Jakarta-Tomcat开发工具库 (<http://jakarta.apache.org>)。如果还没有Web服务器，不妨使用Tomcat。我们还使用独立的JRun服务器 (www.allaire.com) 和Microsoft的PWS与IIS Web服务器的附件。另一个可用的服务器是Orion服务器 (www.orionserver.com)，其中有很好的JSP标注库教程。

Java中处理XML的实用程序和工具库像雨后春笋般不断涌现。为了标准化，我们坚持用Sun公司的标准JAXP (Java API for XML Parsing) 分析器工具库。这个工具库可以从java.sun.com的Web站点下载。这个工具库的新版本目前正在Beta测试阶段，将在本书出版后不久推出。

本书内容

本书的目的是向Java开发人员介绍如何在Java程序中使用XML的技术，我们不做理论论述，而是采用商业化Web站点可以使用的实用应用程序，边实践边探讨。

第1章“XML与数据描述”介绍XML，包括所有难点和典型用法，并概述用Java处理XML的DOM和SAX方法。

第2章“XML中的产品目录”介绍产品目录中使用XML时的设计解决方案。XML是相当灵活的，有时很难选择数据表示的不同方法。我们最终选择的产品目录设计显示了你自已设计时会遇到的许多重要问题。届时，我们要建立带100多个项目的产品目录，并在后面几章使用。

第3章“联机展示XML产品目录”概述生成动态Web页面的Java Servlet和JavaServer Pages API，然后介绍从XML读取数据的标准Java API，以及将其放进Java Servlet、从产品目录取得信息并用HTML表示的方法。其中的重要特性包括关键字搜索功能。

第4章“填充购物推车”介绍如何展开第3章的表示类并生成取得联机订单的实用购物推车。这个过程引入了Java Servlet中实现的对话跟踪概念。

第5章“记账与订单确认”介绍有了实用购物推车后，如何掏用户的腰包并记录订单（当然要用XML）。这个向客户显示信息的例子使用了JavaServer Pages技术。

第6章“联机整理XML产品目录”介绍通过联机接口更新XML格式产品目录的问题。

第7章“调查客户状况”将指出商业化站点的重要部分是收集客户信息。本章介绍具有分支功能的XML驱动问卷系统。分析累计调查结果的问题可以演示处理大型XML文档时如何使用SAX。

第8章“新闻”同样指出每个商业化站点都要显示当前公司新闻。本章将设计灵活的公司新闻XML系统，并开发支持公司新闻XML系统的Servlet与JavaServer Pages。

第9章“吸引回头客”讲述目前大多数商业化站点都包括当前事件新闻清单，以吸引用户经常访问。不需要专门记者报告几百个方面的当前新闻，只要用XML和Java即可，本章演示的正是这个技术。

第10章“Java Web应用程序”基于前面几章详细介绍的关于Java的Web应用程序，论述如何组织资源。本章还要复习v2.2 Servlet API中发布的Sun公司标准，并先睹下一代Java与XML API的情况。

本书选配光盘内容

自然，关于Java与XML的书需要许多例子。在第3章、第4章、第5章和第6章，我们生成了100多个XML产品目录，当然，这是假想的商业化Web站点的产品。其他章节也包括应用程序的样本XML文件。

本书包含大量可用的Java和JavaServer Pages代码。所有这些程序的源代码都在本书选配光盘相应章节的目录中。由于光盘是图书制造中最后一道工序，因此本书选配光盘中的文

件可能包含一些书中没有的增补内容。如果本书出版之后还有一些书中没有的增补内容，我们将放在下列Web站点：

www.lanw.com/books/javaxml/

本书使用的规则

本书的布局尽量保持简单，使用读者熟悉的排版规则，所有代码使用等宽字体，例如：

```
public void _jspService( HttpServletRequest request, HttpServletResponse  
response )
```

说明：有时在正文旁边使用“说明”格式。

警告：这种格式用于强调可能的问题，如偏离标准。

与作者联系

我（Bill Brogden）欢迎读者与本人联系。可以访问我的e-mail地址**wbrogden@bga.com**或邮政地址**William Brogden, 130 Woodland Trail, Leander, TX, 78641**。

欢迎指出书中的错误和问题，以便及时纠正，请将有关材料发到：

<http://www.lanw.com/books/errata>

关于作者

William (Bill) Brogden（本书第3章、第4章、第7章、第8章、第9章和第10章作者）从Java 1.0发布时起就使用Java。他的第一个Java大项目是个小应用程序，显示了动画的接近实时的棒球队游戏。**Bill**在**LANWrights**公司工作，专职用Java技术建立联机课件。**Bill**编写了多本Java方面的著作，包括**Sybex**公司出版的《**Java Developer's Guide to Servlets and JavaServer Pages**》。

在业余时间，**Bill**喜欢阅读科幻小说和训练**Basset**牧羊犬。他和妻子**Rebecca**与几只爱犬一起住在德克萨斯州的**Austin**市。

Chris Minnick（本书第1章与第2章作者）是**Minnick Web Services**公司（www.minnick-web.com）的总裁。他与别人合写了几本Internet相关书籍，是**Software Development Magazine**（软件开发杂志）的友情编辑，是每年的软件开发产品优秀与生产奖（**Software Development Jolt Product Excellence and Productivity Awards**）评委。**Chris**从1996年开始进行Web应用程序方面的写作与教学。他于1996年在旧金山成立了**Minnick Web Services**公司，并于1999年迁址到德克萨斯州**Austin**市。**Minnick Web Services**公司专门从事中小型企业的数据驱动Web应用程序开发，以及大型顾问公司与法律公司的面向项目的Web应用程序开发。

在业余时间，Chris善于培训婚姻技巧，创作小说，为Austin市有线电视频道制作电视节目，和妻子Margaret一起发表联机幽默小品与旅行电子杂志（www.motelmag.com）。

Toivo Lainevoori（本书第5章与第6章作者）从1990年开始专业从事软件开发，他目前是ObjectSpace公司的高级基础结构专家，用Java、XML等各种技术建立面向对象的分布式系统。他还是XMLPatterns.com Web站点的创始人，用一些已知方案来帮助解决XML方案开发中的问题。

在业余时间，Toivo喜欢阅读、滑雪、回加拿大Toronto老家旅行，和妻子Michele一起在住地洛杉矶的小山坡上散步。

目 录

第1章 XML与数据描述	1
何谓XML	1
XML规则	5
用XSL生成样式单	16
在应用程序中使用XML	19
SAX编程	21
XML一览	21
第2章 XML中的产品目录	24
Web站点的要求	24
为何选择XML	26
产品目录与DTD	28
DTD设计原则	48
转向DTD标准化	52
第3章 联机展示XML产品目录	54
展示技术	54
组织产品目录	63
生成展示的信息流	68
第4章 填充购物推车	77
购物推车问题	77
Java中的购物推车	80
CatalogServ类	84
CatalogBean类	96
ProductFormatter类	99
第5章 记账与订单确认	108
付款过程	108
安全性	109
客户信心	110
收集结账信息	110
更新发货信息	131

第6章 联机整理XML产品目录	139
方案概述	139
数据对象	140
展示代码	164
 第7章 调查客户状况	 180
隐私问题	180
调查管理Servlet	184
 第8章 新闻	 220
设计新闻系统	220
新闻系统	223
Web页面展示	224
增加最新新闻	238
 第9章 吸引回头客	 248
新闻源与标准	248
XML文件抓取器	250
NewsModel类简介	255
管理类	263
显示标题的类	267
改进方法	274
 第10章 Java Web应用程序	 275
Servlet API规范	275
下一代	279
扩展问题	280
 附录A Servlet与JSP API	 283
设置应用程序参数	283
取得请求信息	286
设置响应信息	291
错误与异常	293
HTTP状态与错误码	295
JavaServer Pages API	297
PageContext类	298
javax.servlet.jsp.tagext包	302
 词汇表	 303

第1章 XML与数据描述

- XML简介
- XML与电子商务
- XML规则
- 何时与为何使用XML
- 与XML接口的API

作为Java编程人员，近年来你一定听说过XML。本章简要介绍一些基本技术与术语，使你能了解XML的好处。而使用XML开发应用程序时，可以把本章的“XML一览”和“XML规则”作为XML参考资料。

何谓XML

可扩展标记语言（XML，Extensible Markup Language）于1998年由万维网联盟（W3C，World Wide Web Consortium）首次发布，是标准通用标记语言（SGML，Standard Generalized Markup Language）的子集。XML是Internet上使用的灵活而正式的元语言。

元语言（metalanguage）就是描述语言的语言。例如，可以说，英语字典和英语语法书一起构成英语的元语言。

对于XML，其目的就是描述标记语言。标记语言（markup language）用标注（tag）表示数据的结构。超文本标记语言（HTML，Hypertext Markup Language）是目前使用最广的标记语言，最初是用SGML写成的，但现在已经用XML改写。

HTML的最早设计目标是标识科学与学术文献。如果你进行过任何Web开发，则你一定知道，HTML已经延伸到原有的范围之外，能满足Web主人的不同需求。许多人以为XML是HTML的替代品，其实不完全准确。

HTML包含固定标注集，而XML不包含任何标注。事实上，XML可以生成符合特定应用程序需要的标记语言。本书将生成电子商务应用程序。这个应用程序使用的标记语言包含对电子商务应用有意义的标注（如<price>与<quantity>）。

XML应用程序通常具有下列类型的数据和辅助功能：

- XML数据本身，符合严格的结构
- 可选文档类型定义（DTD，Document Type Definition），定义XML文件的结构
- 可选样式单信息，显示数据输出如何格式化
- XML处理器和操纵与重新格式化数据的各种实用程序

XML与电子商务应用

Web开发人员习惯于考虑标注对文本的作用。我们希望HTML的黑体标注（font）标

识黑体文本。实际上，（font）本身并没有任何含义。标注中的文本如何显示完全是由分析数据的程序确定的，对于HTML，这个程序通常就是Web浏览器。由于HTML文档通常是让人们用Web浏览器阅读的，因此许多HTML标记的作用是指定数据如何格式化，而不是标识信息。XML的目的是将文档中的数据与指定数据如何显示的代码分开，使程序更容易取得数据。

例如，假设你是个灯泡经销商，你的Web站点提供各种灯具制造商的最新价格信息。并且你不是检查制造商的站点中提供的最新价格，而是编写一个程序，读取不同站点的产品信息，并在制造商的价格基础上加价10%，于自己的站点中显示产品价格信息。你的供应商之一是ABC Lighting公司。清单11.1是来自ABC Lighting公司Web站点的HTML产品信息表的一部分。

清单1.1 HTML产品信息表的一部分

```
<table>

<tr>
  <th>Product Name</th>
  <th>Description</th>
  <th>Price</th>
</tr>

<tr>
  <td><b>Flashlight</b></td>
  <td>Portable light, without fire!</td>
  <td>$9.95</td>
</tr>

<tr>
  <td><b>Neon Light</b></td>
  <td>Nothing says "class" like Neon lighting.</td>
  <td>$14.75</td>
</tr>

...
</table>
```

HTML文档只指定文本如何格式化。从静态HTML页面取得信息是个费事的工作。如果要从上例中取得手电筒的价格，可以在第一个单元有Flashlight字样的行中寻找第三列的文本，但如果站点设计改变，或者产品名改变，则你的程序可能就不能用了。

有些照明制造商是可能从数据库中动态生成Web页面的。这时，就要与每个站点的Web主人联系，生成其数据的接口，但这样既费时又费事，为取得数据的每个数据库都可能要用不同方法。如果文档本身包含取得有意义的信息所需的全部信息，则会方便得多。

清单1.2演示了同样的信息如何用XML文档表示。

清单1.2 包含产品信息的XML文档

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE ABC_Lighting:catalog SYSTEM "catalog.dtd">

<ABC_Lighting:catalog xmlns:ABC_Lighting = "http://www.abclighting.com">
```

```
<ABC_Lighting:product>
<ABC_Lighting:name>Flashlight</ABC_Lighting:name>
<ABC_Lighting:description>Portable light, without fire!
</ABC_Lighting:description>
<ABC_Lighting:price>$9.95</ABC_Lighting:price>
</ABC_Lighting:product>

<ABC_Lighting:product>
<ABC_Lighting:name>Neon Light</ABC_Lighting:name>
<ABC_Lighting:description>Nothing says "class" like Neon lighting.
</ABC_Lighting:description>
<ABC_Lighting:price>$14.75</ABC_Lighting:price>
</ABC_Lighting:product>

</ABC_Lighting:catalog>
```

这个文档中的第一行是XML声明，提供信息给XML分析器。XML声明表示文档类型和所用的XML版本。这个语句不是必需的，但习惯上，XML文档用这个语句开始。**standalone** = “no” 属性表示这个文档有DTD。下一行是文档类型声明。这条语句指定文档符合哪个文档类型声明——这里DTD为**catalog.dtd**。注意，尽管都缩写为DTD，但文档类型声明与文档类型定义大不相同。文档类型声明表示XML文档符合哪个文档类型定义。

文档类型声明还显示文档的根元素是什么。根元素是包括文档中其余所有内容的元素。这里，根元素为**ABC_Lighting:catalog**。冒号前面的元素名部分是标注的名字空间。名字空间不是必需的，但可以保证标注的惟一性。如果ABC Lighting公司要开始在自己的Web站点上销售其他公司生产的产品，则可以用名字空间避免外部数据中同名而不同构的元素产生错误。

catalog.dtd的内容可能如下：

```
<!ELEMENT ABC_Lighting:catalog (product)*>
<!ELEMENT ABC_Lighting:product (name, description?, price+)>
<!ELEMENT ABC_Lighting:name (#PCDATA)>
<!ELEMENT ABC_Lighting:description (#PCDATA)>
<!ELEMENT ABC_Lighting:price (#PCDATA)>
```

这个DTD指定了产品目录中可以出现的元素，出现的顺序和可以或必须出现的次数。利用XML数据和DTD，可以方便地从XML文档通过程序标识与读取有用信息，见本书余下部分介绍。

XML资源

下列资源介绍了最新进展、XML未来计划和使用XML的工具：

- World Wide Web Consortium (www.w3c.org).
- O'Reilly & Associates, Inc.'s XML.com (www.xml.com)—one of the best commercial XML information sites on the Web.
- The XML Industry Portal (www.xml.org).
- xmlhack (www.xmlhack.com)—a news site for XML developers.
- Enhydra (www.enhydra.org)—home of the Enhydra Java/XML application server.
- The Unicode Consortium (www.unicode.org).

XML的用途

XML在客户端和服务方均有部署。下面几节介绍各种方法。此外，XML还可以用于数据存储，将在客户端和服务方之后介绍。

客户端XML

在客户端，XML可以在一定程度上定制数据表示，这在HTML中是很难或不可能实现的。例如，个人数字助理（PDA, Personal Digital Assistants）和移动电话之类支持Web的设备，需要页面用不同于标准Web浏览器的方法格式化。将站点发送到小屏幕设备的典型方法可生成站点的全新版本。利用XML文档中的结构化数据，则可以将数据与格式分开，只要对数据采用不同样式单就能够在不同类型设备上定制站点显示。

服务器方XML

XML对当今的服务器有最大的影响。服务器方使用XML的方法之一是应用消息。消息就是应用程序和计算机之间交换的数据。要让应用程序、计算机和公司共享信息，就要确定消息格式。要了解XML对消息系统的影响，一定要了解消息的一些背景知识。确定发送消息的标准是人们开始通信时就遇到的问题，但我只追溯到30年前。

开放系统电子商务技术指南欧洲工作组（EWOS TGEC 066, European Workshop on Open System's Technical Guide on Electronic Commerce）定义的电子商务包括市场、合同交换、后期支持、付款、与行政部门交互（如纳税与海关数据交换）等不同活动。电子数据交换（EDI, Electronic Data Interchange）是20世纪70年代由运输数据协调委员会（TDCC, Transportation Data Coordinating Committee）提出的。在财务之类的行业，已经有30多年的联网历史，EDI已成为标准电子商务消息格式。EDI是公司需要以标准格式交换商业数据而产生的。EDI系统的问题是，建立与维护费用非常昂贵，而且通常需要专用网络。

20世纪80年代，工作组的电子邮件开始产生并在公司内实现。厂家希望将其e-mail方案建成标准的，而大量公司开始依赖电子消息系统。Microsoft Mail与Lotus cc:Mail之类的软件包使小公司可以在办公室内交换消息，但扩展性较差，随着规模的扩大而变得难以管理，也很难提供与局域网（LAN）之外的连接。但是，和其他计算世界一样，消息系统也往分散方向发展。电子消息系统偏离集中的、高度控制的专用网络，使这个技术面向更多用户和更多用途。但分散化也导致兼容性问题、重复劳动和无法利用公司共享知识等一系列问题。

随着Internet的出现，公司对标准化、灵活性的需求越来越迫切，希望公司和员工以标准化、灵活性的方式进行通信和从事电子商务。免费提供的标准化电子消息格式将影响每一种通信，包括商业化通信和非商业化通信。

但是，首先要确定一种语言。XML应运而生。XML适合设计消息格式的主要原因是其具有简单性。XML没有任何可选特性，不与任何操作系统或厂家相捆绑，与多年来为SGML开发的大量工具和应用程序兼容。XML严格执行慎重确定的规则，保证任何XML分析器都能读取与理解任何XML文档。此外，许多用户更熟悉用标记语言，而不熟悉建立EDI系统所需的消息格式。使用XML，任何能编写XML文档的人都可以生成消息格式。

XML在Web文档中的另一用途是指定元内容。元内容（metacontent）就是关于内容的内容，使搜索引擎可以取得更好的结果。例如，如果需要寻找产生在德克萨斯州Austin市的新闻故事，则可以搜索“Austin Texas News”。由于今天的大多数搜索引擎只是检索一个站点中的所有内容，因此这个搜索返回的许多结果可能根本不是你所要的。如果Austin市的新闻文章按结构化XML文档形式编写，则可以进行更精确的搜索，例如City = Austin, State = TX, StoryType = News。

XML与数据存储

XML还可以生成数据库。XML按树式结构存储数据。尽管XML文档不是最有效的数据存储方式，但有其独到的优点。和消息一样，其最重要的优点是简单性。树式结构是组织数据的直观而熟悉的方法。此外，几乎任何类型的数据结构都可以表示为XML数据树——从关系型数据库到面向对象的数据库到层次式结构。用XML存储数据的另一重要优点是XML支持Unicode字符集。结果，任何国际化字符都是合法的XML文档字符。

Unicode是实现国际标准组织（ISO）所定义的通用字符集（UCS, Universal Character Set）的标准方法，是在计算机处理中表示文本的通用字符编码标准。Unicode用UCS转换格式（UTF, UCS Transformation Formats）将字符编码变为实际位。

XML规范指定XML处理器要支持两个UTF: UTF-8与UTF-16。UTF-16用两个字节表示每个字符，而UTF-8用一字节的ASCII字符编码表示ASCII字符，用变长编码表示非ASCII字符。UTF-8在需要保持与ASCII字符的兼容性时有用。UTF-8的缺点是用1到3字节表示非ASCII字符。如果大部分文本为ASCII字符，则UTF-8能节省空间。如果使用非ASCII字符，则UTF-8会浪费空间。XML的缺省编码为UTF-8。文档中要使用的字符编码在XML声明中用encoding属性指定，例如：

```
<?xml version="1.0" standalone="no" encoding="UTF-8"?>
```

XML规则

今天的HTML浏览器通常显示一切，不管HTML标记多么奇怪或差劲。而XML处理器则要求在遇到标记错误时产生致命错误。致命错误就是应用程序停止处理，并显示错误消息。这种严格性通常称为严厉错误处理（draconian error-handling）。尽管这种严厉错误处理让HTML作者和SGML作者不容易适应，但这样可以保证XML文档在每个XML处理器中总是以相同方式解释。

符合XML语法规则的XML文档称为结构合理的。XML作者在规范中编写进了结构合理性，目的是防止XML成为浏览器之战等问题的根源。Microsoft与Netscape之间浏览器之战的结果是HTML作者经常要担心兼容性问题。如果XML中发生这种问题，则XML的价值将大大降低。

XML处理器是个软件模块，允许应用程序访问XML文档中存储的数据。XML处理器可以验证或不验证。验证处理器要检查文档结构与DTD中指定的规则，而不验证处理器则只保证文档符合XML规则。