

开发专家
之 Sun ONE

JavaWeb服务 应用开发详解



飞思科技产品研发中心 编著

本书附赠
光盘包括
全部实例
代码



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

开发专家之 Sun ONE

Java Web 服务应用开发详解

飞思科技产品研发中心 编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

出版说明

“开发专家”是电子工业出版社计算机图书研发部长期以来精心培育的计算机科学技术类本版品牌。这个品牌是由多个专题系列组成的横向大系列，涵盖了计算机技术的各个方面，特别是一直受到极大关注的程序开发类系列，例如《开发专家之数据库》、《开发专家之网络编程》、《开发专家之 Delphi》以及《开发专家之 Sun ONE》等。这些专题系列基于各自的角度，从纵向上包含了该专题的所有内容。因此，整个“开发专家”的品牌架构纵横交错，囊括了所有的计算机技术和所有的技术层面，海纳百川而又极具可扩展性。

“开发专家”的作者队伍主要依托于“飞思科技产品研发中心”。“飞思科技产品研发中心”是由专业的策划人员、权威的技术专家和资深的作者队伍共同构成。在图书的出版上，形成了以研发为基础、以出版为中心、以服务为支持的专业化出版框架和流程。通过深入的市场调查和技术跟踪，在综合了技术需求和读者焦点等因素的基础上，形成各系列丛书的写作重点和大纲，然后聘请业界的最前沿学者进行写作。同时，策划工作全程介入写作进程，严格控制写作质量，用最专业的技术背景、最深刻的理论基础、最具代表性的案例、最能为专业读者接受的形式，为读者提供品质最佳的图书产品，体现了出版者和作者的完美结合。

多年来，计算机图书研发部始终把创造社会效益摆在首位，秉承一切为国内计算机技术专业读者服务的精神，为推动国内 IT 技术发展、为体现国内技术的原创水平，穷尽所有的创意与努力，将出版者的命运与读者的支持紧紧地连在了一起。

在此，我们临出版之残酷竞争而不惧，旌旗猎猎而异军突起，这与广大读者的支持是分不开的。为使我们的脚步更坚实、使我们的队伍永葆活力和创造力，我们期待着您能为我们的前进贡献出您的意见和建议。同时，我们也在等待着您的加入。

我们的联系方式：

电 话： (010) 68134545

E-mail: support@fecit.com.cn

网 址: <http://www.fecit.com.cn> <http://www.fecit.net>

电子工业出版社计算机图书研发部

前 言

关于本套丛书

从来没有任何事物像互联网那样，对人类的活动产生如此深刻的影响，无论是政府、企业，以及个人，莫不如此。与此同时，IT 工业也正面临着一场变革——传统应用向基于 Internet/Web 的服务模式转化。

翻开历史，我们可以看到互联网的形成和发展就是以分布性、开放性和平台无关性为基础，这是 Internet 与生俱有的属性。随着互联网应用的发展，又引入了诸如 RPC/COM/CORBA 等技术，但这些技术在实际应用中，又存在着很多不足和局限。它们的特定协议也难以通过防火墙，因而不适于在 Web 上的应用开发。为了进一步开发基于 Web 的应用，相继出现了 Sun 公司的 Sun ONE (Open Net Environment 开放网络环境) 和 Microsoft 公司的 .NET 两大 Web 服务技术体系。其中，Sun ONE 以 Java 技术为核心，更接近或者满足于互联网在智能化 Web 服务上对分布性、开放性和平台无关性的要求，同时其在健壮性、安全性、组件化等方面也更为成熟稳定，获得了众多 IT 厂商和产品的支持，是目前惟一在市场上得到了广泛应用的技术体系。

Sun ONE 体系结构以 Java 语言为核心，包括 J2SE/J2EE/J2ME，并基于一系列开放和流行标准、技术及协议。要特别指出的是，Sun ONE 体系结构本身作为开放式体系结构，在得到 IBM/BEA/Oracle/Sybase 等这些 IT 巨擎支持的同时，更得到了互联网上 Open Source 社区的青睐。我们很容易地从网上免费获得和使用包括 Java 集成开发环境、Java 数据库，甚至是中间件 (Application Server) 服务器等产品，以及它们的源代码。这对于加速国内中小企业的信息化建设和自有知识产权产品开发、提高企业应用和软件行业的整体水平，无疑是一次难得的机会。

综观国内的技术发展，广大的 Java 程序开发人员以及正在转向 Java 体系进行开发的技术人员虽然已面临这一令人激动和鼓舞的转型期，却苦于没有足够的相关资料和文献，尤其对国内的最新 Java 技术动态和技术现状知之甚少，而图书市场上 Java 的书籍尽管汗牛充栋，但精品罕见，能反映出 J2EE 以及 Sun ONE 的框架全貌的书籍更是奇缺。

电子工业出版社计算机图书研发部为进一步推动国内 Java 技术的应用与发展，不失时机地推出了《开发专家之 Sun ONE》系列丛书。

本套丛书以 Sun ONE 整体架构为基础，全面体现了 Sun ONE 的技术核心——Java 的应用开发。丛书从各个角度深入 Java 应用开发的各个层面，涵盖了 Java 技术的所有重要思想和实践，体现了最新的 Java 技术进展和动态，大幅度提升读者的理论和应用水平。同时，丛书重点突出实用性。书中引入了大量的行业应用范例，使读者不仅能快速掌握开发技能，而且对于开发者进行综合系统分析也有所裨益。

关于本书

Java 是一种跨平台的语言,使用 Java 语言可以实现程序编写一次后可以不做任何修改就在各种不同的操作平台上运行;而 XML 则可以说是一种跨应用的语言,使用 XML 可以方便地在各种不同的应用之间互相通信。如果将这两种技术结合起来,就可以实现非常诱人的跨平台、跨应用各类解决方案。这两种技术的结合,在 Web Services 概念出现后,才真正变得标准化,易于实施。

C/S(客户机/服务器)构架曾是企业信息化建设的一个非常好的解决方案,但随着 Internet 和宽带网络的迅猛发展,C/S 构架的缺陷很明显地显现出来,它对客户机的要求限制了信息的流通,且给软件的升级维护带来了很大的麻烦。但随之而来的基于 Web 的 B/S(浏览器/服务器)构架很好地解决了这个问题,它将计算集中于中央服务器上,客户只需要一个浏览器(即瘦客户),虽然目前还有很多应用工作在 C/S 构架上,B/S 无疑成了未来系统的主流构架。

软件构架日益庞大且呈现分布化,软件的组件化和 Web 服务尤显重要。随着 B/S 构架带来的基于 Web 的组件技术,如 Sun 的 EJB、Microsoft 的 DCOM 等技术也迅速发展起来。但是,这些组件技术也存在一些方面的缺陷,如各项组件技术各成体系,相互调用比较困难等等。解决这些问题的方案就是本书的主题——Web 服务。

本书基于目前最流行的 Web 服务框架,用深刻的理论和丰富的实例揭示了 Java 在 Web 服务中的应用与开发技术。全书分为两篇共 14 章,包括 Web 服务概述、XML 语言及其编程接口、SOAP、应用 Apache SOAP 进行编程、WSDL、UDDI 等专业开发者最关心的 Java 技术内容,同时使读者对整个分布式架构有一个全面而透彻的认识,触类旁通地理解基于 Java 和基于 .NET 的 Web 服务。书中的例程对实际的应用开发有非常强的借鉴作用。

本书适合于专业和准专业的 Java 程序员阅读,也是正在转向进行 Java 开发的各类程序员的必备 Java 参考书。

本书由飞思科技产品研发中心策划并组织编写,黄洋成、刘忠、沈书荣、钱和良、徐惠清、蒋远涛、周凯、汪益华、陈和勋、勾立玉等参加了本书的写作工作。同时,在本书的写作过程中得到了董毅先生的大力支持和协助,他提出了大量的参考性意见更使本书增色不少,在此表示衷心的感谢。

当然,限于作者水平,加之时间仓促,书中不足之处难免,敬请读者批评指正。

我们的联系方式:

电 话: (010) 68134545 68134811

E-mail: support@fecit.com.cn

网 址: <http://www.fecit.com.cn> <http://www.fecit.net>

飞思科技产品研发中心

目 录

第一篇 XML 与 Java

第 1 章 强大的 Java	3
1.1 Java 的背景介绍	3
1.2 Java 的技术特征	4
1.2.1 语言特点	4
1.2.2 Java 的面向对象介绍	5
1.2.3 Java 和 C++ 比较	7
1.3 Java 程序的运行与开发环境	11
1.3.1 Java 运行环境	11
1.3.2 Java 开发环境	12
1.4 Java 程序设计基础	12
1.4.1 语言基础	12
1.4.2 类、接口与包	13
1.4.3 字符串与数组	17
1.5 Java 的技术应用	17
1.5.1 Java 应用程序	17
1.5.2 Java 小应用程序	17
1.5.3 JDBC	18
1.5.4 Servlet 和 JSP	21
1.5.5 JavaBeans	23
1.5.6 EJB	23
第 2 章 互联网时代的数据标准——XML 语言	27
2.1 XML 介绍	27
2.1.1 XML 的发展之路	28
2.1.2 XML 的特点	33
2.1.3 XML 的应用简介	38
2.2 XML 语法	40
2.2.1 语法基础	40
2.2.2 XML 文档的构成	46
2.2.3 XML 文档的元素	49
2.2.4 XML 文档的属性	53
2.2.5 XML 文档的实体	58
2.3 XML 的命名空间	59
2.3.1 命名空间的表示法	59
2.3.2 命名空间的范畴	60

2.4	文档类型定义	60
2.4.1	文档类型定义介绍	61
2.4.2	DTD 的设计	62
2.4.3	正式的 DTD 结构	65
2.4.4	DTD 的缺点	68
2.5	XSL 介绍	69
2.5.1	XSL 简介	69
2.5.2	XSL 文档	70
2.5.3	XSL 样式模板	71
2.5.4	编写 XSL 文档	76
第 3 章	XML 编程接口	79
3.1	DOM (文档对象模型)	79
3.1.1	DOM 的介绍	79
3.1.2	DOM 的层次模型	79
3.1.3	DOM 的使用	81
3.2	SAX (XML 简单应用程序接口)	120
3.2.1	什么是 SAX	120
3.2.2	Java 的 SAX API	121
3.2.3	SAX 的事件模型	121
3.2.4	SAX 的使用	122
第 4 章	Java 与 XML	139
4.1	JAXP 编程	139
4.1.1	JAXP 的介绍	139
4.1.2	使用 JAXP	140
4.2	JDOM 编程	153
4.2.1	理解 JDOM	153
4.2.2	JDOM 应用	154
第 5 章	XSLT 和 XPATH	159
5.1	XSLT 概述	159
5.1.1	XSLT 的设计目的	159
5.1.2	为什么需要 XSLT	159
5.1.3	XSLT 的特点	160
5.1.4	XSLT 的机理	160
5.2	样式表结构	161
5.3	内容模式	164
5.3.1	根结点子元素	164
5.3.2	剥去空白域 (Whitespace Stripping)	164
5.4	模板规则	164
5.4.1	模式	164

5.4.2	定义模板规则	165
5.4.3	应用模板规则	166
5.4.4	调用命名的模板	166
5.5	产生结果树	166
5.5.1	字面结果元素的产生	167
5.5.2	用 <code>xsl:element</code> 生成元素	167
5.5.3	<code>xsl:attribute</code> 生成属性	167
5.5.4	命名属性集	168
5.5.5	产生文本	169
5.5.6	产生处理指令	169
5.5.7	产生注释	170
5.5.8	复制	170
5.5.9	计算生成的文本	171
5.5.10	编号 (numbering)	171
5.6	循环	171
5.7	条件处理	173
5.7.1	<code>xsl:if</code> 条件处理	173
5.7.2	<code>xsl:choose</code> 条件处理	173
5.8	排序	174
5.9	变量和参数	175
5.10	输出	176
5.10.1	XML 输出方法	177
5.10.2	HTML 输出方法	177
5.10.3	文本输出方法	177
5.11	其他细节	178
5.11.1	消息 (Message)	178
5.11.2	扩展 (Extension)	178
5.11.3	退步 (Fallback)	179
5.12	XPath	179
第 6 章	Java XML Pack	181
6.1	Java XML Pack 简介	181
6.2	JAXM	181
6.2.1	核心	182
6.2.2	供应者管理工具	182
6.2.3	使用 JAXM	182
6.3	JAXB	185
6.4	JAX-RPC	186
6.5	JAXR	186

第二篇 Web 服务篇

第 7 章 Web 服务概览	189
7.1 什么是 Web 服务	189
7.1.1 Web 服务架构	189
7.1.2 Web 服务的特征	190
7.1.3 Web 服务的分类	190
7.1.4 Web 服务堆栈	191
7.2 Web 服务带来了什么	192
7.3 Web 服务体系结构对比	193
7.3.1 Microsoft.NET 开发框架	193
7.3.2 Sun ONE 开发框架	196
7.3.3 Sun ONE 与 .NET 的战略对比	199
第 8 章 简单对象访问协议	203
8.1 SOAP 初步	203
8.1.1 HTTP+XML+RPC+Serialization	204
8.1.2 XML-RPC	204
8.1.3 WDDX	205
8.1.4 SOAP: XML-RPC+WDDX	206
8.2 SOAP 协议	207
8.2.1 SOAP 术语	208
8.2.2 SOAP 消息	209
8.2.3 SOAP 编码	211
8.2.4 SOAP RPC 表示	213
8.2.5 绑定到 HTTP 的 SOAP	214
8.3 SOAP 和现有技术的比较	215
8.3.1 CORBA	215
8.3.2 COM/DCOM	215
8.3.3 SOAP 与 CORBA、DCOM/COM 的比较	216
第 9 章 应用 Apache SOAP 进行编程	217
9.1 Apache SOAP 的获取	217
9.2 Apache SOAP 的安装、配置和运行	218
9.2.1 Servlet 容器 (Tomcat 3.2.3) 的安装	218
9.2.2 Java Mail API 和 JavaBeans Activation Framework 的安装	221
9.2.3 XML Parser (Apache Xerces) 的安装	222
9.2.4 Apache SOAP 的安装	222
9.2.5 Apache SOAP 的运行	224
9.3 一个简单的 SOAP 程序	224
9.3.1 SOAP 服务的编程	224

9.3.2	SOAP 服务的发布	225
9.3.3	客户端程序	226
9.3.4	例子的运行	227
第 10 章	深入 Apache SOAP 编程	229
10.1	基础知识	229
10.1.1	SOAP 服务	229
10.1.2	SOAP 服务的发布	230
10.1.3	调用 SOAP 服务	246
10.1.4	Http Tunnel Tools	248
10.2	SOAP RPC 编程	252
10.2.1	SOAP 服务	252
10.2.2	SOAP HTTP Client	254
10.2.3	SOAP SMTP Client	256
10.3	SOAP Message 编程	264
10.3.1	Message Service	265
10.3.2	SOAP Message Client	266
10.3.3	带有附件的消息	272
10.4	Script 服务	278
10.5	EJB 服务	284
10.6	COM 服务	296
10.6.1	编写 COM 组件	297
10.6.2	客户端程序	299
10.7	使用复杂的数据类型	299
10.7.1	在服务器上注册新类型	300
10.7.2	在客户端注册新类型	308
10.7.3	使用复杂数据类型的例子	308
10.7.4	构造自己的（反）序列化器	323
10.8	定制 SOAP 消息传输层	330
10.8.1	使用 HTTP 代理	330
10.8.2	使用 HTTPS	335
10.8.3	使用 Transport Hooks	336
10.9	维护 Session 信息	339
10.10	在 RPC 中使用 MIME 类型	345
10.11	构造自己的 Provider	352
10.12	Apache SOAP 服务器配置	362
10.12.1	配置管理器的作用	362
10.12.2	配置管理器类的实现	362
10.12.3	在 SOAP 服务器上使用自定义的配置管理器	365
10.13	Axis 简介	366

10.13.1	服务的发布	366
10.13.2	服务的调用	367
第 11 章	SOAP 互操作	369
11.1	传输问题	369
11.2	XML 的问题	370
11.3	SOAP 自身的问题	370
第 12 章	Web 服务描述语言	389
12.1	WSDL 语法结构	389
12.2	WSDL 工具	393
12.2.1	WSDL Generator Tool	394
12.2.2	Service Proxy Generator Tool	394
12.2.3	Service Implementation Template Generator Tool	394
第 13 章	通用描述、发现和集成	395
13.1	UDDI 商业注册	395
13.2	UDDI 数据模型	395
13.2.1	商业实体信息	396
13.2.2	商业服务信息	397
13.2.3	绑定模板信息	397
13.2.4	技术规范信息	398
13.3	UDDI 程序员 API	399
13.4	UDDI4J	401
第 14 章	Web 服务流语言	411
附录 A	SOAP V1.1 协议规范	417
A.1	简介	417
A.1.1	设计目标	417
A.1.2	符号约定	417
A.1.3	SOAP 消息举例	418
A.2	SOAP 消息交换模型	419
A.3	与 XML 的关系	419
A.4	SOAP 封装	420
A.4.1	SOAP encodingStyle 属性	421
A.4.2	封装版本模型	421
A.4.3	SOAP 头	421
A.4.4	SOAP 体	422
A.4.5	SOAP 头和体的关系	423
A.4.6	SOAP 错误	423
A.4.7	SOAP 错误代码	424
A.5	SOAP 编码	424
A.5.1	XML 中的编码类型规则	425

A.5.2	简单类型	427
A.5.3	多态 accessor	429
A.5.4	Compound types 复合类型	429
A.5.5	默认值	437
A.5.6	SOAP root 属性	437
A.6	在 HTTP 中使用 SOAP	438
A.6.1	SOAP HTTP 请求	438
A.6.2	HTTP 头中 SOAPAction 域	438
A.6.3	SOAP HTTP 应答	439
A.6.4	HTTP 扩展框架	439
A.6.5	SOAP HTTP 举例	439
A.7	在 RPC 中使用 SOAP	440
A.7.1	RPC 和 SOAP 体	440
A.7.2	RPC 和 SOAP 头	441
A.8	安全性考虑	441
A.9	SOAP 封装举例	441
A.9.1	请求编码举例	441
A.9.2	应答编码举例	442
A.10	参考文献	444
附录 B	Web 服务描述语言 (WSDL) 1.0	445
B.1	简介	445
B.1.1	WSDL 文档示例	446
B.1.2	标记规则	447
B.2	服务定义	448
B.2.1	WSDL 文档结构	448
B.2.2	类型	452
B.2.3	消息	453
B.2.4	端口类型	455
B.2.5	绑定	457
B.2.6	端口	458
B.2.7	服务	458
B.3	SOAP 绑定	459
B.3.1	SOAP 示例	459
B.3.2	SOAP 绑定如何扩展 WSDL	461
B.4	HTTP GET 和 POST 绑定	465
B.4.1	HTTP GET/POST 示例	465
B.4.2	HTTP GET/POST 绑定如何扩展 WSDL	467
B.5	MIME 绑定	468
B.5.1	MIME 绑定示例	469

B.5.2	MIME 绑定如何扩展 WSDL	470
B.6	参考资料	472
B.6.1	有关 URI 的说明	472
B.6.2	WSDL 示例的线上格式	473
B.6.3	扩展性元素的位置	473
B.6.4	架构	474
附录 C	UDDI 技术白皮书	481
C.1	简介	481
C.1.1	概览	481
C.1.2	UDDI 的商业注册与 UDDI 商业注册中心	481
C.1.3	使用 UDDI	482
C.1.4	背景	482
C.1.5	UDDI——技术发现层	483
C.1.6	下一步	484
C.1.7	商业发现与 UDDI	484
C.1.8	进一步的工作	484
C.2	技术概述	485
C.2.1	四种信息类型	485
C.2.2	商业实体信息: businessEntity 元素	485
C.2.3	服务信息: businessService 元素和 bindingTemplate 元素	486
C.2.4	规范描述的指针和技术标识	486
C.2.5	程序员 API	487
C.2.6	构建于 SOAP 之上	487
C.2.7	查询 API	488
C.2.8	UDDI 调用模型	488
C.2.9	远程 Web 服务调用失败后的恢复	488
C.2.10	发布 API	489
C.2.11	安全: 识别与授权	489
C.3	其他信息	490
C.4	UDDI 信息模型	490
C.5	资源	491
附录 D	UDDI 执行白皮书	493
D.1	新的国际贸易环境	493
D.2	新的挑战	493
D.3	当前的解决方案	494
D.4	新的解决方案	494
D.5	UDDI 计划	495
D.6	前景	495
D.7	总结	496

开发专家之

Java ONE

第一篇 XML 与 Java

XML 语言的诞生给互联网带来了新的生机。这种描述性的语言使得互联网上的信息变得更有意义，更易于用户查找，也使“软件即服务”的概念成为了现实。而像 Java 这种以网络为基础的跨平台语言则更能利用其自身的优势与 XML 语言相结合，从而加强了 Java 语言的网络功能和应用的范围。在国外，已经有 41% 的 Java 程序员在利用 XML 开发 Java 应用程序。

读书笔记

日期: _____	进程: _____	备注: _____

<http://www.fecit.com.cn> E-mail: fecit@fecit.com.cn

Tel: (010) 68134545 68134811

第 1 章 强大的 Java

对于 Java 这个面向网络的语言来说, XML 的产生为它提供了更广阔的应用空间。本章从 Java 的背景知识讲起, 系统、简要地介绍了 Java 语言的一些基础知识。在介绍了基础知识后还将向读者讲述 Java 语言的应用范围和应用前景, 这些技术都是近几年来刚刚兴起的新技术, 并且有着非常明朗的应用前景。像 JSP、EJB、JDBC 等技术都和 XML 语言结合得非常紧密。

1.1 Java 的背景介绍

1991 年, SUN Microsystems 公司的 Jame Gosling、Bill Joe 等人, 在为电视、控制烤面包箱等家用消费类电子产品开发一个分布式代码系统, 这样可以通过 Internet 与家电产品进行信息交互, 并对其进行控制。开始, 他们准备采用 C++ 语言来编程实现, 但 C++ 太复杂, 而且安全性差, 最后基于 C++ 开发了一种新的语言——Oak (Java 的前身)。

Oak 是一种用于网络的精巧而安全的语言, Sun 公司曾用它参加了一个交互式电视项目的投标, 结果被 SGI 打败。可怜的 Oak 几乎要销声匿迹, 恰在这时 Mark Andreesen 开发的 Mosaic 和 Netscape 启发了 Oak 项目组的成员, 他们将 Oak 语言继续完善, 取名为 Java。1994 年下半年, 他们用 Java 语言编制了纯 Java 的浏览器——HotJava, 成功地演示了使用 Java 撰写复杂应用的能力, 以及如何下载并运行 applets。此后一些著名的计算机公司纷纷购买了 Java 语言的使用权, 如 Microsoft、IBM、Netscape、Novell、Apple、DEC、SGI 等。1995 年 Java 语言被美国的著名杂志 PC Magazine 评为年度十大优秀科技产品 (计算机类就此一项入选), 随之大量出现了用 Java 编写的软件产品, 受到工业界的重视与好评——“Java 是 20 世纪 80 年代以来计算机界最激动人心的一件大事”。

SUN Microsystems 公司的总裁 Scott McNealy 认为 Java 为 Internet 和 WWW 开辟了一个崭新的时代。

环球信息网 WWW 的创始人 Berners-Lee 说: “计算机事业发展的下一个浪潮就是 Java, 并且将会很快发生。”看来, 使用 Java 已成大势所趋!

Apple、HP、IBM、Microsoft、Novell、SGI、SCO、Tandem 等公司均将 Java 并入各自开发的操作系统, 而负责开发并推广 Java 技术的 SunSoft 公司 (这是 SUN 下属的一个子公司), 将通过颁发许可证的办法来允许各家公司把 Java 虚拟机和 Java 的 Applets 类库嵌入他们开发的操作系统, 这样各类开发人员就能更容易地选择多种平台来使用 Java 语言编程, 不同的用户也就可以脱离 Web 浏览器来运行 Java 应用程序, 这无疑是很受广大用户欢迎的, 也为 Java 语言的应用开拓了极为广阔的前景。

1.2 Java 的技术特征

1.2.1 语言特点

在 Sun 的 Java 语言白皮书中明白无误地说明了 Java 语言的以下一些特点：简单、面向对象、分布式、解释执行、健壮、安全、体系结构中立、可移植、高性能、多线程以及动态性。

1.2.1.1 简单 (Simple)

Java 语言的简单特性主要体现在以下三个方面：

- Java 的语言风格来源于 C++，因而 C++ 程序员能够非常快速地上手。
- Java 摒弃了一些 C++ 中容易引发程序错误的地方，如指针；同时又增加了一些新的特色，如内存管理等。
- Java 提供了丰富的基础类库，使编程更为简单。

1.2.1.2 面向对象 (Object-Oriented)

面向对象是 Java 最重要的特性，Java 语言的设计是完全面向对象的，它摒弃了 C++ 中全局变量等与面向对象思想冲突的一些内容。我们将在后面 Java 与 C++ 的比较中详细描述。

1.2.1.3 分布式 (Distributed)

Java 有一个很周全的程序库，且很容易地与 HTTP 和 FTP 等 TCP/IP 通信协议相配合。Java 应用程序能在凭借 URL 打开并访问网络上的对象，对程序员来说，其访问方式与访问本地文件系统几乎完全一致。为分布式环境尤其是 Internet 提供动态内容是一项非常繁杂的任务，而通过 Java 却能很容易地实现这个目标。

1.2.1.4 健壮 (Robust)

Java 对程序在编译和运行时的错误进行严格检查，极大地保证了 Java 应用程序的健壮性。具体而言，其健壮性主要体现在以下这几个方面：

- 强大的错误检测功能，具体包括 Null 指针检测、数组边界检测、异常出口检测以及字节代码检测等。
- 提供内存管理功能，大大减少内存出错的机会。

1.2.1.5 安全 (Secure)

Java 是被设计用于网络及分布式的环境中，安全性自然是一个很重要的考虑。Java 的安全性可从两个方面得到保证：一方面，在 Java 语言中，像指针和释放内存等 C++ 功能被摒弃，从而避免了非法内存操作；另一方面，当 Java 用来创建浏览器内容时，语言功能和一些浏览器本身的功能结合，使它更为安全。