

探索你不知道的Java世界



Java

深度历险

王森 著



华中科技大学出版社

<http://press.hust.edu.cn>



Java深度历险

探索你不知道的Java世界

7年来, Java已发展得非常成熟, 市面上有太多介绍Java程序语言与Java各种应用的好书。Java在企业端(Enterprise)与嵌入式系统(Embedded System, 手机与PDA等)上扮演举足轻重的角色。而这些出色的应用, 都是因为架构在一个稳定的根基——Java 2 SDK (Software Development Kit)之上。然而时至今日, 并没有任何一本书详细地解释Java 2 SDK以及JRE (Java Runtime Environment)如何相互配合, 让Java程序畅行无阻; 也没有一本书解释Java编译器在编译Java程序时的细节; 更没有一本书告诉您Java程序如何与其它系统技术做整合。但这些知识能够让您将Java技术发挥得淋漓尽致。

本书特色:

◆ 揭开JDK, JRE与JVM三者的关系

探讨Java应用程序的执行流程, 让您以后不会再为JDK与JRE的版本问题苦恼。

◆ 剖析类加载器与类库运作机制

类别载入器兼具Java程序的弹性与安全性两项重大任务, 您不能不了解它。

◆ import与package机制的探讨

撰写Java程序的人, 绝对会遇到import与package机制。只要明白这个机制的运作原理, 您会发现Java是一个设计优良的技术。

◆ Ant与Java

Ant是专门针对Java所开发的项目管理工具, 通过简单的XML控制文件, 可以管理项目的组成、编译、包装以及执行等工作。用了之后您一定会爱上它。

◆ Java与Office

想用Java来控制 Microsoft Office 吗? 本书教您如何使用 Borland c++ Builder, 让您一步步撰写可以调用 Microsoft Office的Java程序。

◆ Visual Studio.net与Java

本书教您如何使用Visual Studio.net撰写一个可以重复使用Java类别的c++应用程序。

◆ Java 2 SDK源代码的追踪

本书讲解Java 2 SDK源代码的下载与编译方式, 并解说Java 2 SDK源代码的构架。如果您想追踪Java 2 SDK源代码, 本书所介绍的内容将是您的起点。

ISBN 7-5609-2832-3



9 787560 928326 >

定价: 29.80元(含1CD)

Java

深度历险

王 森 著

华中科技大学出版社

中国·武汉

图书在版编目(CIP)数据

Java 深度历险/王 森 著

武汉:华中科技大学出版社,2002年10月

ISBN 7-5609-2832-3

I. J...

Ⅰ. 王...

Ⅲ. Java 语言-程序设计

Ⅳ. TP312

本书封面贴有华中科技大学出版社(原华中理工大学出版社)
激光防伪标签,无标签者不得销售。

版权所有 翻印必究

责任编辑:周 筠 <http://yeka.xilubbs.com>

技术编辑:方 舟 (kingofark)

责任校对:张兴田

出版发行:华中科技大学出版社 (武昌喻家山 邮编:430074)

录 排:华中科技大学惠友科技文印中心

印 刷:湖北新华印务有限公司

开 本:787×1092 1/16

印 张:17 插 页:2 字 数:350 000

版 次:2002年10月第1版

印 次:2002年10月第1次印刷

印 数:1—6 000

定 价:29.80元(含1CD)

ISBN 7-5609-2832-3/TP·484

个别术语的译法对照表

在转译过程中，技术编辑针对大陆和台湾两地技术术语的差异做了许多斟酌。您在书中可能会发现一些术语并不符合自己的习惯，这些都是我们本着技术转译在技术上严谨和准确的原则，经过反复思考之后保留下来的译法。效果究竟如何，还有待广大读者的评价。我们欢迎这方面的讨论。下面是一份关于术语的整理表格。（方舟注：这段话借用了《Effective C++ 中文版》里面孟岩先生的句子。）

英文术语	大陆惯用译法	本书译法
Adapter	适配器	配接器
Hierarchy	层次（体系）	阶层（体系）
instance	实例	实体
library	库、函数库	程序库
load	加载	载入
loader	加载器、装载器	装载器
native	本地、本机	原生

插图中出现的重要术语之繁简对照表

下面列出书中画面截图里包含的重要繁体术语及其简体含义，以便不熟悉繁体术语的读者参考。

繁体术语	简体术语	繁体术语	简体术语
變數	变量	巨集	宏
編輯(E)	编辑	類別	类
由於某		錄製	录制
檔案(E)	文件	命令提示字元	命令提示符
檔名	文件名	批次檔	批处理文件
檔案資料夾	文件夹	取代(E)	替换
呼叫	调用	說明(H)	帮助(H)
環境變數	环境变量	我的最愛(A)	收藏夹(A)
檢視(V)	查看	新增(N)	新建(N)

简体版序言

我本来以为,把使用繁体中文的书籍转换成简体中文,只不过是 Word 上按下繁简互换的功能钮罢了。后来读了侯捷老师所发表的《科技翻译面面观——从计算机领域谈起》(<http://www.jjhon.com/article01-6.htm>)一文,我发现,就算是用语比较精确的计算机领域,两岸的名词仍存在颇大的差异,在其它领域就更不用说了,许多常用的词儿甚至有相反的意思。不过,让我更感到讶异的是,与华中科技大学出版社合作出版这本书之后,才发现这些差异远比我想象中的还要大上许多。虽然都是中文,但是在用语转换上并不简单。

很幸运的是,华中科技大学出版社的周筠编辑找到了方舟先生来进行这本书的转换工作。我没有办法用我所知的语言来表达方舟先生所带给我的感动,方舟先生整理了一份转译备忘录,这是因为他很认真地看过每一个字,很认真地测试过书上所提到的每一个细节,挑出每一个打字错误,甚至很认真地思考我用的句子能否被大陆的朋友所接受,也认真地比较两岸每个技术用语的优缺点。他整理的这份备忘录如此详尽,使我深信华中科技大学出版社和方舟先生是这本书简体版的“再生父母”,如果没有他们,这本书肯定又成为我口中常说的“转译公害”。

如果在阅读这本书的时候,您觉得语句很通顺,那一定是方舟先生的功劳。方舟先生为了维持原著的完整性,肯定曾经在很多地方想额外加注,却隐忍了下来,所以如果您觉得这本书仍有缺失,那一定是我的错,请给我一封 E-mail,尽情给我严厉的批评。

在这本书之前,我总共写了两本与 J2ME 相关的书籍,不过我发现繁体版和简体版的发售时间竟然相差了半年以上,使得大陆的朋友在新知识的取得上严重落后。不过您可以放心,这版书的繁体版和简体版的编校工作几乎是同时进行的,所以两种版本的发行时间绝不会再相差这么长时间,

甚至简体版的误植部分会比繁体版的少(当然,又是方舟先生的功劳☺)。希望这一次,不会再有两岸信息不同步的遗憾。

Sun Microsystems Taiwan 教育训练服务 技术顾问

王森 moli0205@mail2000.com.tw

繁体版序言

我的第一份“正式”工作就是在升阳计算机 (Sun Microsystems) 台湾分公司的教育训练中心从事讲师工作。在这之前，我除了当管理学院的研究研究生之外，闲暇之余还会到处兼职当顾问，接接 Project，写程序糊口。这本书很多最初的概念都源自于当时所服务的公司里工程师所遇到的难题，或是自己使用 Java 撰写应用程序时所遇到的百思不解的问题。

承蒙洪志鹏总经理愿意收留我这样的人，到了 Sun 工作之后，利用内部网络所看到的世界，竟然是一个我从没想过的天堂。我猜这是很多人愿意一开始就到大型软件公司上班的最大原因。在 Sun 呆了一阵子你就会发现，Sun 跟 Microsoft 是不一样的公司，由于两家企业所发迹的领域不太相同，导致 Sun 在设计软件的时候比较重视架构，尤其是应用在企业领域上的软件架构，您可以从两个地方看出来，第一个是在 Java 的认证考试中，最高级的认证叫做 SCJA (架构师认证)，第二个是市面上关于 Java 的书籍，排除入门书之后，许多重要的书籍都和 Design Pattern 有关系，比方说 *J2EE Design Pattern*、*EJB Design Pattern*，关于系统的书籍就少得可怜，讲 JVM 的书屈指可数，我手边讲述 Java 系统的书籍，我认为是好书的也只有两本。微软则刚好相反，微软重视系统更甚于架构，您会发现 .NET 刚出来，就有一堆讲述 CLR (Common Language Runtime) 或 .NET 内部运作的书籍，而讲述架构的书籍就少得多了。

很可惜的是，我本身就是一个系统的爱好者。最近这一年唯一让我拍案叫奇的好书，只有 Jeffery Richter 所撰写的 *Applied Microsoft .NET Framework Programming*，我就是喜欢 *Applied Microsoft .NET Framework Programming* 这种书籍的读者。看完 Jeffery Richter 的书之后，最大的悔恨就是 Java 出现了将近 7 年，可是却没有任何一本类似的书籍让我们更了解 JDK，或是 Java 的运作原理。这本《Java 深度历险》就是在这种悔恨之下

形成的，虽然不是很厚，但是里面的每一篇都是我这几年来研究 Java 运作的心得。所以这里要感谢《Run!PC》的技术编辑吕志敏先生，没有他每个月定期的催稿，我就没有办法定期地整理我的心得，并将它们发表在杂志上，更没有办法在获得很多读者的响应之后，重新整理这些内容成册，并适当地加以修订和整合。我希望喜欢系统的朋友读完这本书之后，不再有跟我一样的悔恨，而且对 Java 这个将近 10 年前就有的设计更具信心。

在 2002 年 JavaTwo 大会上，我用《数学家、物理学家，以及工程师的差别》来介绍自己。数学家的心情套用在我的身上，就知道我是个非常懒惰的人，你可能会以为我手边有 JDK 的源代码，所以我就会先花功夫去读懂 JDK 的源代码，然后把结果告诉你。事实上并非如此！撰写这本书，尤其是写第五章“package 与 import 机制”的时候，因为我懒得去看 Java 编译器的源代码，所以我采用的方式是归纳法。我先设计了很多奇奇怪怪的 use cases，然后对 Java 编译器进行测试，利用所产生的结果来想象 Java 编译器的运作方式，最后归纳出属于我自己对 Java 编译器运作方式的“理论”。最后我花一点点时间来阅读源代码，发现竟然和我的推论不谋而合，所以这一章大概是我自以为最有成就感的一章。这种写作方式也让我发现，现代科学家在探究我们所身处的世界时所使用的方法，竟然是如此地让人兴奋。

记得有一次讲完一场研讨会之后，有个听众来问我：“王森先生，我们公司使用 Borland JBuilder 开发应用程序，可是遇到很多问题，比方说常常出现 Class Not Found 的错误信息”。原来，JBuilder 虽然是一套 RAD 开发工具，但是它却不像其它如 Visual Basic、Delphi、Borland C++ Builder 一样好上手，因为它有其进入门槛。而这个进入门槛，来自于存在于 JBuilder 底部的 Java 2 SDK。如果对 Java 2 SDK 没有深入的了解，想要平顺地驾驭更高阶的开发工具绝对是一件很难的事情。我希望这本书能带您跨过这个门槛。

这本书并未对 Java 程序语言作讨论，因为坊间这类的好书已经非常多了。本书着眼于其它 Java 书从来没有提到的议题，期望带领您真正对 Java 进行“深度历险”，也希望众多 Java 爱好者能对这本书给予响应，可以支

持我整理出更多有趣的议题。

要感谢的人很多，很多我都在我先前的著作中提及了，我在心里默默地感激他们。如果您认为这本书带给您很多新的知识，请您也跟我一同感谢这些我生命中不可或缺的人们，以及催生这本书的所有工作伙伴。

Sun Microsystems Taiwan 教育训练服务 技术顾问

王森 moli0205@mail2000.com.tw

目录

第一章	深入 Java 2 SDK	1
	■ 前言	1
	■ 执行 java.exe 时所发生的怪事	3
	■ JDK、JRE、JVM 之间的关系	9
	■ 您所执行的是哪一个 java.exe	17
	■ 常见的错误	22
	■ 本章小结	28
第二章	深人类装载器	29
	■ 前言	29
	■ 为何要自己全盘掌控动态性	30
	■ 我们在不知不觉中用到动态性	31
	■ 预先载入与按需载入	35
	■ 让 Java 程序具有动态性的两种方法	39
	■ 用显式的方法来实现动态性：使用 Class.forName() 方法	40
	■ 用显式的方法来实现动态性：直接使用类装载器	49
	■ 自己建立类装载器来载人类	53
	■ 类被哪个类装载器所载入	56
	■ 一切都由 Bootstrap Loader 开始：类装载器的阶层 体系	59

■ 委托模型	69
■ 类装载体系	80
■ 类装载器的功用	86
■ 本章小结	87
第三章 Java 与 Microsoft Office	89
■ 简介	89
■ 本章目的	91
■ 基本技能	92
■ 架构	93
● 功能模块建立步骤一：设计桥接用的接口类	94
● 功能模块建立步骤二：撰写使用桥接接口类的 Java 程序	97
● 功能模块建立步骤三：产生编译动态链接库时所需要的 C/C++ 头文件	98
● 功能模块建立步骤四：建立套表打印模块的基本架构	99
● 功能模块建立步骤五：Visual Basic for Application	109
● 功能模块建立步骤六：完成套表打印功能模块	116
■ 本章小结	127
■ Java Native Interface 参考资源	128
第四章 用 Visual Studio.net 来操纵 Java 虚拟机	131
■ 前言	131
■ 简介	132
■ 用 Visual Studio.NET 撰写主程序	134
■ 无法调用 Java 虚拟机	145
■ 顺利调用 Java 虚拟机	146
■ 调用 Java 类库	150

	■ 本章小结	154
第五章	package 与 import 机制	155
	■ 前言	155
	■ 初探 package 与 import 机制	156
	■ 深入 package 与 import 机制	177
	■ 编译时期(Compile-time)的 Package 运作机制	177
	■ Java 的动态链接本质	193
	■ 运行时期(Run-time)的 Package 运作机制	196
	■ 查看	199
	■ 本章小结	201
第六章	Ant	203
	■ 前言	203
	■ 关于 Ant	205
	■ 基础工具的安装	209
	■ Ant 的下载与安装	209
	■ Java 图形用户接口应用程序开发流程	213
	■ 混淆器	214
	■ 下载及安装 RetroGuard	216
	■ 双击后即可执行的 JAR 文件	220
	■ 项目的目录结构	225
	■ 设置属性文件 (build.properties) 与构造文件 (build.xml)	231
	■ 本章小结	238
附录	Java 2 SDK 版源代码概观	239
	■ 简介	239
	■ 如何取得 Java2 SDK 源代码	241

■ Java 2 SDK 1.3.0 源代码的结构	248
■ Java 2 SDK 1.4.0 源代码的结构	250
■ 如何编译 Java 2 SDK 1.3.0 源代码	251
■ 网络资源	258

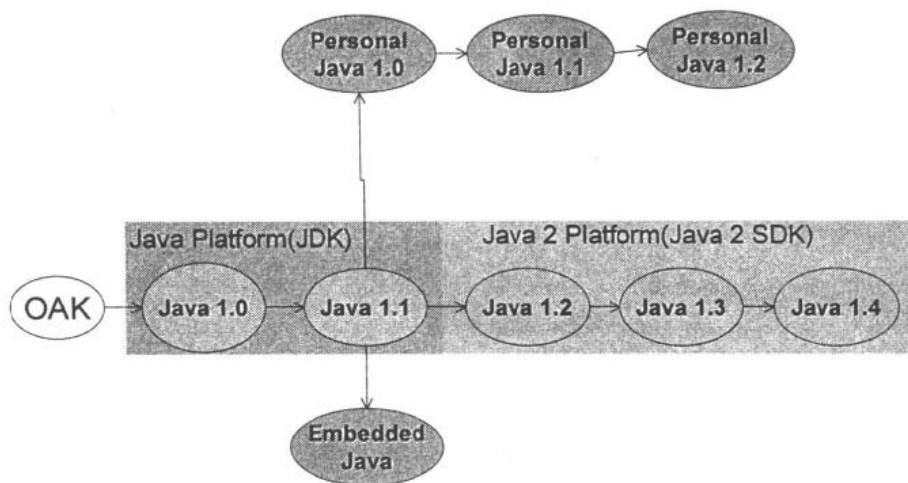
第一章

深入 Java 2 SDK

你越讨厌的事情，就越容易遇上。

■ 前言

Sun Microsystems 所发表的 Java 开发工具 —— Java 2 SDK，永远都是 Java 初学者最早接触到的开发工具。一般人习惯称这套工具为 JDK (Java Development Kit)。Java 版本及开发工具的演化见下图。



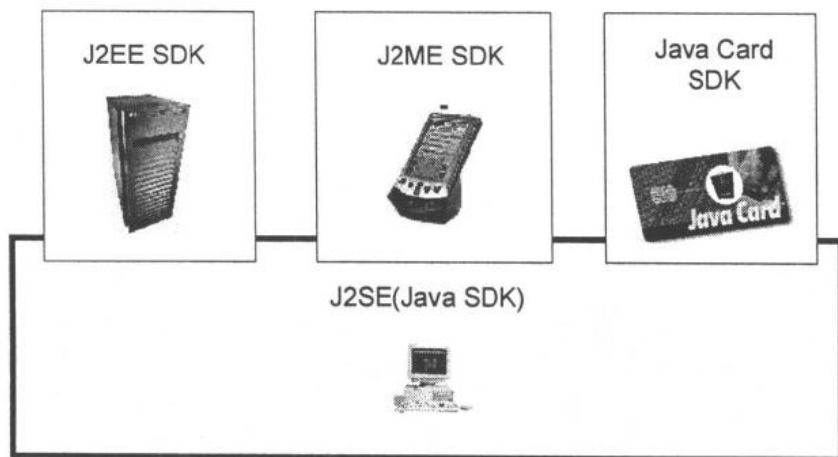
JDK 是在 Java 2 Platform 之前的 Java Platform 所使用的开发工具名称 (在本文中有时会写做 JDK，有时会写做 Java 2 SDK，但指的将是同一种东西)。我记得曾经有人戏称它是 Java Developer Killer 的缩写，除了挖

苦 Sun 所制作的开发工具没有微软设计的开发工具来得方便之外，其实还说明了另外一件事，那就是 Java 从 1995 年发表后到现在，即使每一版的 JDK 都会附上为数庞大的官方说明文件，却从来没有任何一份文件或一本书籍详细说明这套官方开发工具的特性。

没有文件或书籍来描述 JDK 的特性并不代表这些特性不重要。毕竟，任何最新的标准类库，或是最新版本的虚拟机，一定都会伴随着最新版的 JDK 发行。就算您想跳过 JDK，直接使用如 Borland JBuilder 或 Forte for Java 这类的高级开发工具，JDK 仍然如影随形。以 Borland JBuilder 来说，当您 will Borland JBuilder 安装完成之后，在 JBuilder 的所在目录下也会内含一套 Java 2 SDK，如下图所示。



因此，我们可以得知，不管您如何讨厌 JDK，只要想开发 Java 相关的应用程序，您就无法逃离 JDK 的掌握。事实上，不管您开发的是 J2SE、J2EE、J2ME，甚至 Java Card 的应用程序，除了需要各种版本对应的开发包之外，一定需要 JDK 的辅助，如下图所示。



那么，JDK 到底是什么东西呢？从技术的观点来说，因为高阶开发工具都是架设在 JDK 上，因此高阶开发工具的行为或是引发的错误信息都是源自 JDK。为了更正确地掌握高阶的 Java 开发工具，我们必须了解 JDK 的特性和组成。从求知的观点来看，Java 程序员每天输入无数的 `java xxx.java` 与 `java xxx`，到底 Java 程序是如何运作的？在我们看不到的底层，到底发生了什么事情？如果我们可以清楚地得知所有的来龙去脉，将会使我们更了解这套开发工具。

上述两个观点，都是本章所希望告诉您的。让我们开始深入了解 JDK 吧！

■ 执行 `java.exe` 时所发生的怪事

当您在使用 JDK 时，您是否曾经发现执行 `java.exe` 的时候，会有如下这些奇怪的现象：