



清华松岗系列丛书

C/C++程序员 Java 编程

麦中凡 陶 伟 编著
曹广通 郝爱丽

JAVA

清华大学出版社

C/C++程序员 Java 编程

麦中凡 陶 伟
曹广通 郝爱丽 编著

清华大学出版社

(京)新登字 158 号

C/C++程序员 Java 编程

麦中凡 陶 伟 曹广通 郝爱丽 编著

本书介绍 Java 语言的语法、编程知识以及用 Java 开发 Internet 应用程序的方法。

本书共分 7 章,主要内容包括:Java 语言概述、Java 语言的诸多新机制、用 Java 写小应用程序、图形用户界面编程、网络编程以及用线程编制程序以展示 Java 高度的交互性和动态性等。书中配有大量示例及说明。书后配有 7 个附录,提供了大量的相关信息。

全书内容简明清晰,写作深入浅出,特别适合于具有 C 和 C++编程基础的人员用来快速学习掌握 Java 语言。

5176/10

版权所有,翻印必究。本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

出 版 者:清华大学出版社(北京 清华大学校内,邮政编码:100084)

责任编辑:张孟青

责任校对:李凤茹

印 刷 者:清华大学印刷厂

发 行 者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:10.75 字数:257 千字

版 次:1996 年 10 月第 1 版 1996 年 10 月第 1 次印刷

书 号:ISBN 7-302-02340-9/TP·1161

印 数:0001—6000

定 价:19.00 元

前 言

随着 Internet 迅猛发展,World Wide Web 日益普及,整个计算环境正在经历着深刻的变革。这表现为:高度的分布式,异质网络上不同的机器(操作系统)之间进行信息交换,存在着不同的标准,例如:CORBA、Network OLE、OpenDoc 等等,不同国家、行业被 Internet 连接在一起,相互通讯,共享全世界的计算机资源和信息。

我们知道网络技术的灵魂之一就是保证不同质的信息安全地、自由地交流,客观上需要统一的传输协议和数据描述语言。为此 Internet 的网络传输协议采用了 TCP/IP,保证了异质机器信息交换的畅通无阻。但是,采用 TCP/IP,程序之间传输的信息是无格式的二进制数据流,数据的语义由信息交换的程序双方共同来解释与维护。所以,在 Client/Server 模型中,Client 和 Server 之间只能进行数据交换,很难进行程序代码交换。在 Java 产生之前,Client/Server 之间进行程序代码交换必须解决两个问题:其一,双方的机器和操作系统必须是相同的,或者双方能够建立一种同质计算环境;其二,Client/Server 双方必须解决如何动态地加载传输过来的程序代码。这些问题实际上一直没有得到很好的解决。

1989 年,超文本置标语言 HTML(Hyper Text Markup Language)和环球信息网 WWW(World Wide Web)的产生是 Internet 数据描述语言的一次飞跃。环球信息网把全世界信息资源用 HTML 格式统一起来,HTML 文件之间复杂的连接关系,构成了庞大的、内容丰富的多媒体信息空间(Cyberspace)。人们可以更方便地利用 Internet 上的信息资源。

1996 年,Sun 公司的 Java 语言的正式发表则是一次 Internet 的技术革命。Java 的诞生,从根本上解决了 Internet 的异质、代码交换以及网络程序的安全性等诸多问题。首先,Java 是一种平台无关语言。Java 程序编译后,生成字节代码(byte-code),运行在 Java 虚拟机(实际上是一个解释器)上。一个操作系统平台只要提供 Java 虚拟机,Java 程序就可以在上面运行。从理论上讲,Java 程序可以运行于所有的操作系统平台上,从根本上解决了 Internet 的异质问题。其次,Java 采用了可移动代码技术,在网络上不仅可以进行无格式的数据信息交换,而且可以进行程序交换。Java 是比较纯的面向对象语言,它的绝大多数程序实体都是对象,利用对象的封装性可以大大降低网络上程序交换的复杂性。第三,Java 可以和 HTML 无缝地集成,把静态的超文本文件变成了可执行的应用程序,极大地增强了超文本的可交互操作性。最后,Java 是一种更安全的语言,它消除了 C 和 C++ 中众多的不安全因素,提供了诸多安全保障机制,例如:异常处理、代码检查等,比其它语言更适合网络应用软件的开发。

Java 不仅仅是一个程序设计语言,更是一个网络操作系统。这是因为:第一,Java 可以不依赖于任何现有操作系统,可以直接运行在某种机器或 CPU 上(例如:Java 芯片,80x86);第二,Java 有自己的内存管理功能;第三,Java 支持多线程程序,可以对处理器资源进行分配和管理;第四,Java 可以对设备进行管理,例如:屏幕,键盘,鼠标等;第五,Java 支

持网络文件管理,Java 可以打开远程机器上的文件,Java 的文件的概念已经扩充到整个 Internet 网络。可以预见,Java 将从根本上改变网络应用程序的开发和使用方式。

本书设定为已经掌握 C 和 C++ 的程序设计、并对局域网有一定了解的人员编写的。其目的是介绍 Java 的语法,Java 语言和 C++ 的区别和联系,Java 引入的一些新的语言机制(相对于 C++ 而言),Java 程序设计的入门知识,Java 和 HTML 的关系等等。至于 Java 复杂的程序包 Package(相当于 C++ 的类库),则需要精读 Java 的 API 参考手册,多调多练才能掌握。

由于编写时间仓促,加之水平有限,虽再三校对,疏漏之处在所难免,希望广大读者批评指正。

编 者

于北京航空航天大学计算机系

1996 年 6 月

目 录

第零章 Internet 基础知识	1
0.1 国际互连网 Internet	1
0.2 环球信息网 WWW(World Wide Web)	1
0.3 超文本置标语言 HTML	3
0.4 Java 提高了 WWW 的交互性	3
0.5 TCP/IP 网络协议	4
0.6 IP 地址	5
0.7 域名	5
第一章 Java 语言概述	6
1.1 发展历史	6
1.2 语言特点	8
1.3 运行方式	12
1.4 Java 带来的影响	13
第二章 Java 语言规格说明	17
2.1 程序结构	17
2.2 词法问题	17
2.2.1 注释	17
2.2.2 标识符	18
2.2.3 关键字	18
2.2.4 字面量	18
2.2.5 运算符及特殊分隔符	20
2.3 类型	20
2.3.1 数值类型	21
2.3.2 布尔类型	21
2.3.3 数组	21
2.4 类	23
2.4.1 类类型之间的强制转换	24
2.4.2 方法	25
2.4.3 覆盖方法	27
2.4.4 重载确定	27

2.4.5	构造函数	28
2.4.6	用 new 运算符创建对象	30
2.4.7	静态方法、变量和初始化	31
2.4.8	访问指定符	33
2.4.9	变量作用域规则	34
2.4.10	修饰符	34
2.5	界面	36
2.5.1	界面作为类型	37
2.5.2	界面中的方法	37
2.5.3	界面中的变量	37
2.5.4	组合界面	37
2.6	包	38
2.6.1	定义一个编译单元的包	38
2.6.2	使用其它包中的类和界面	38
2.7	表达式	39
2.7.1	运算符	39
2.7.2	强制和转换	42
2.8	语句	42
2.8.1	声明	42
2.8.2	表达式	43
2.8.3	控制流	43
2.8.4	异常	44
第三章	用 Java 写小应用程序	50
3.1	小应用程序概述	50
3.2	最简单的 applet	51
3.3	在 applet 中绘图	52
3.4	事件	53
3.5	读取 applet 参数	57
3.6	在 applet 中加入按钮	59
3.7	在 applet 中显示图象	60
3.8	动画和声音	64
3.9	与其他 applet 的通信	66
3.10	在 HTML 文件中嵌入小应用程序	70
3.10.1	小应用程序嵌入 HTML 文件的格式	70
3.10.2	小应用程序的运行	72
3.10.3	<EMBED> 标签	73

第四章 图形用户界面	74
4.1 一个信息对话框.....	75
4.2 显示所有的 Java GUI 构件	77
4.3 处理 GUI 事件	83
4.4 滚动条.....	86
4.5 定义自己的 GUI 构件	89
4.6 一个完整的 GUI 程序剖析	92
第五章 网络编程	99
5.1 URL 和 URL 类	99
5.2 读/写 URLConnection	100
5.3 套接字(socket)	104
5.4 数据报文(Datagram)	108
第六章 Java 线程编程	111
6.1 创建线程	111
6.2 启动和终止	112
6.3 挂起和恢复	114
6.4 调度(scheduling)	116
6.5 调度的例子	116
6.6 synchronized,wait() 和 notify()的综合应用	117
第七章 Java 开发工具 JDK	121
7.1 JAVAC——Java 语言编译器.....	121
7.2 JAVA——Java 语言解释器	123
7.3 JAVADOC——Java 文档产生器	126
7.4 JAVAH——Java 本地方法的 C 文件产生器	127
7.5 JAVAP——Java 类反汇编程序.....	129
7.6 JDB——Java 程序调试工具	130
7.7 JAVAPROF——Java 剖析工具.....	133
7.8 APPLETVIEWER——Java applet 观察器	133
附录 A 浮点.....	136
附录 B Java 语言的文法	138
附录 C URL 简介	146
附录 D HTML 文档	148
附录 E Java Internet 资源表.....	156
附录 F 词汇表.....	157
参考资料	164

第零章 Internet 基础知识

0.1 国际互连网 Internet

Internet 是世界上最大的计算机网络,它起源于美国国防部 1969 年开始实施的 ARPAnet 研究项目。ARPAnet 项目的一个最主要的研究成果是推出了 TCP/IP 通信协议,并且 Internet 的 TCP/IP 软件及 Internet 技术规范对任何人都是公开的。正是由于 Internet 的开放性,它才能迅速演变为国际上最流行的互联环境。

据统计,到 1995 年 5 月,入网的国家和地区已达 168 个。连入 Internet 的网络已达 46 318 个。计算机达 644 万台。用户数目超过 6000 万。每天的通信量超过 100GB。我国到 1996 年 7 月,已有一百多所大学连入 Internet。这将从根本上改变我国科研的环境和方式,极大地推动我国科学事业的发展。

Internet 能提供范围极广的服务,如:Web,FTP,E-Mail, Telnet, Wais, Gopher 等。其中 Web 是 Internet 环境中最为流行的一种服务业务。人们几乎把 Web 与 Internet 视为等同。

0.2 环球信息网 WWW(World Wide Web)

1989 年 3 月,在欧洲核物理实验室(CERN)工作的 Tim Berners 在 Internet 上开发出一种名为 WWW 的高级浏览服务。用户可把 Web 看成存放于 Internet 的各个计算机上的超媒体文档的集合。一个指定的 Web 文档可以包含链接项,通过这些链接项来指定存储在其它计算机上 Web 文档。这种服务所遵循的一套通信协议,后来逐渐发展为一个名为 WWW 标准的国际性的协议。它主要采用以下的协议和标准来进行信息的定位、存取和显示。

- * URL(Uniform Resource Locator):统一资源定位。这是用来标识某一特定信息页所用的一个字符串。

- * HTTP(Hyper Text Transport Protocol):超文本传输协议。这是为分布式超媒体信息系统设计的、面向对象的传输协议。

- * HTML(Hyper Text Markup Language):超文本置标语言。这是在 WWW 中用来建立超媒体文件的语言。它通过标签和属性对一段文本的语义进行描述。

- * CGI(Common Gateway Interface):公共网关界面。它遵循 CGI 标准编写的服务器侧的可执行程序,能运行于任何服务器上。

Internet 上的所有服务都是基于客户/服务器计算模式的,Web 更不例外。Web 浏览器

(Browser)是 Web 服务器的客户。Web 浏览器提出对超媒体(包括文本、声音、图象等)信息的访问请求,Web 服务器给出相应的回答。

据 1995 年的有关统计,在 Web 上使用最广泛的是 NCSA 服务器,而近期发展最快的是 Netscape 服务器。1993 年出现于 Unix 上的 NCSA Mosaic 是第一个真正的 Web 浏览器。后来,Netscape 推出 Navigator,Sun 推出了 HotJava,Micr soft 推出 Explorer.NCSA 的 Web 服务器几乎支持 Internet 上的所有服务,如 FTP,Telnet Gopher 等。NCSA Mosaic 的推出引起了 Internet 革命性的变化,它促使 Web 成为 Internet 上最为流行的服务业务。

1994 年末推出的 Netscape 服务器是目前发展最快的 Web 服务器,其优势在于响应速度快、效率高和易于使用,特别是负载大的情况下,仍然保持高的响应速度。主要原因是 Netscape 服务器采用进程池工作模式。与 NCSA 所采用的派生模式(Forking Mode)不同,它在响应客户请求时不必派生出多个副本,而只创建一些轻量进程,明显减少了对内存的需求,因而能在大负载下保持高效率。

图 0.1 给出了一般的基于客户/服务器模式的 Web 环境示意图。服务过程大致可以分为以下四个步骤:

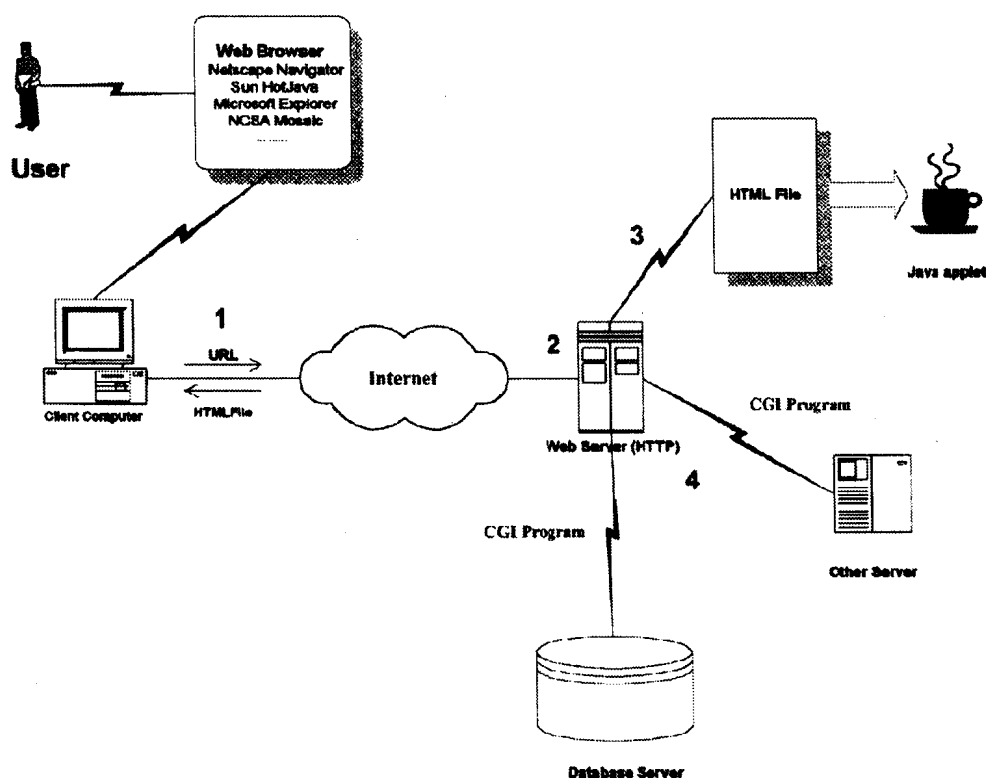


图 0.1 基于客户/服务器模式的 Web 环境示意图

- (1) Web 用户用浏览器指定 URL 向 Web Server 发出请求。
- (2) Web Server(或称 HTTP Server)把 URL 转换成页面所在服务器上的文件路径名。
- (3) 若文件是简单的 HTML 文件,由 Web Server 直接把它送给 Web 浏览器,包括 Ja-

va applet。

(4) 若文件是一个遵守 CGI 标准的驻留程序,则由 Web Server 运行它,并把结果送至 Web 浏览器。

0.3 超文本置标语言 HTML

超文本置标语言 HTML 是国际标准 ISO 8879:1986 SGML(Standardized Generalized Markup Language)的实际应用之一。SGML 定义了结构化文本类型和标示这些文本类型的置标语言系统。HTML 是一种置标语言,用以生成超文本文档。在这种文档中,可以加入指向任何文档(文本、图象、动画和声音等)的链接。用 HTML 编写的文档在 Internet 媒体类型中称为“text/html”。

HTML 作为一种置标语言,它定义了超文本的文档的结构和一系列标签及其属性。其标签大致可以分为以下几个部分:

- * 文档结构定义。
- * 字体字形定义。
- * 版面布局定义。包括版面颜色、分段换行、原文引用等。
- * 链接定义。包括普通文档链接(Link)、图形化的文档链接以及内部文档链接等。链接的文档类型可以是正文、图形、声音等。

HTML3.0 支持 Java,CGI 程序,使 Web 的灵活性更大了。

0.4 Java 提高了 WWW 的交互性

继 HTML 和 WWW 之后,Java 语言可称得上是 Internet 上的第三个重要的技术发明。Java 是指由 Sun 公司开发的一种面向对象的编程语言,它极大地提高了 Web 交互性,使 Web 从静态的文本变成了可执行的程序。嵌入在 Web 页面的 Java 程序一般被称为小应用程序(applet)。它可以和主页(Hompage)一起在网上被传输,applet 一旦被浏览器加载,便能使页面“活起来”。

Java 有一个重要的特点是平台无关。Java 程序均被编译成一种虚机器代码,运行在一个解释器上。所以,只要一个操作系统带有 Java 解释器,或者 Web 浏览器带有 Java 解释器,Java 程序就可以在上面运行。这对于复杂的 Internet 是非常重要的,因为 Web 服务器并不知道客户端是何种机器,何种操作系统。Java 语言的平台无关性成功地解决了 Internet 上的异质问题,如图 0.2 所示。

此外,Java 语言具有简单性、分布式、可移植性、健壮性及安全性等特点。这些特点我们在后面的章节还要详细介绍。Java 对 Web 网上的信息交流起着一种革命性的作用。

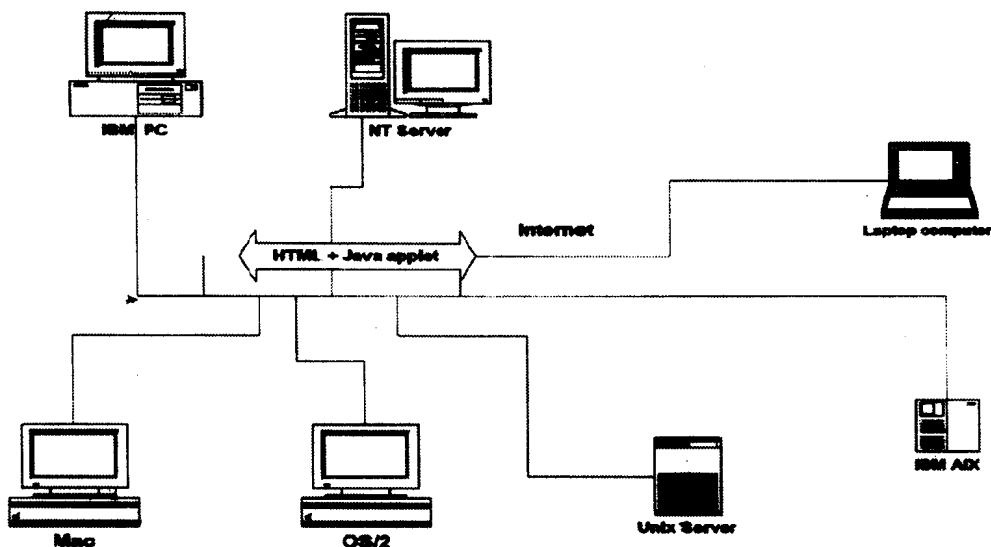


图 0.2 Java 的平台无关性

0.5 TCP/IP 网络协议

TCP(Transmission Control Protocol)/IP(Internet Protocol)是一种网络通信协议。它规定了网络上的所有通信设备的通信规范,定义了一个主机与另一个主机之间的数据交换的格式以及传送方式。TCP/IP 是 Internet 的基础协议,也是一种计算机数据打包和寻址的标准方法。我们可以把 TCP/IP 理解为有两个信封,如图 0.3 所示。

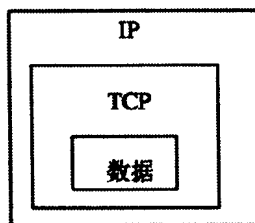


图 0.3 TCP/IP 数据打包示意图

TCP 和 IP 就像信封,要传递的信息被划分成若干段,每一段塞入一个 TCP 信封,并在该信封面上记录有段号的信息,再将 TCP 信封塞入 IP 大信封,发送上网。在接受端,一个 TCP 软件收集信封,抽出数据,按发送前的顺序还原,并加以校验。若发现差错,TCP 将会要求重发。因此,TCP/IP 在 Internet 中几乎可以无差错地传送数据。

对 Internet 用户来说,并不需要了解网络协议的整个结构,仅需了解 IP 的地址格式,即可与世界各地进行网络通信。

0.6 IP 地址

为了使连入 Internet 的众多计算机在通信时能够相互识别,Internet 中的每一台主机都分配有一个唯一的 32 位地址。该地址称为 IP(Internet Protocol)地址,也称作网际地址。IP 地址由 4 个数组成,每个数可取值 0—255,各数之间用一个点号“.”分开,例如:202. 112. 131. 234。实际上,每个 IP 地址是由网络号和主机号两部分组成的。网络号表明主机所连接的网路,主机号标识了该网络上特定的那台主机。如上例中 202. 112 是网络号,131. 234 是主机号。

IP 地址有三种基本类型,由网络号的第一组数字来表示。A 类地址的第一组数字的取值范围为 1—126,B 类地址的第一组数字的取值范围为 128—191,C 类地址的第一组数字的取值范围为 192—223。

0.7 域名

IP 地址是以数字来代表主机的地址的,比较难记。为了使用和记忆方便,也为了便于网路地址的分层管理和分配,Internet 在 1984 年采用了域名管理系统(Domain Name System),入网的每台主机都具有类似于下列结构的域名:主机号. 机构名. 网络名. 最高层域名。域名用一组简短的英文表达,比用数字表达的 IP 地址容易记忆。常见的最高层域名有:COM(商业)、EDU(教育)、MIL(军事)、GOV(政府)等。国别的最高层域名有:UK(英国)、AU(澳大利亚)、CN(中国)等。

例如:北航有一台计算机的 Internet IP 地址是 202. 112. 128. 51,域名为 manager. buaa. edu. cn,其含义是:中国(cn)教育网(edu)北航网(buaa)一台叫 manager 的主机。

加入 Internet 的各级网络依照域名管理系统的命名规则对本网内的主机命名和分配网内主机号,并负责完成通信时域名到 IP 地址的转换。对使用者来说,可以直接使用域名,Internet 上的域名服务器 DNS(Domain Name Service)会自动把域名转为 IP 地址。

Internet 域名服务系统是一个分布式信息系统,采用客户/服务器结构。用户可向本地域名服务器查询地址。如果没有,它向上一级服务器查找,最后查出所需的地址。

第一章 Java 语言概述

1.1 发展历史

美国硅谷有一句行话,每 10—15 年有一次轮回。这次轮回就是从 Java 开始。Java 是由 Sun Microsystem 公司开发的新一代的程序设计语言,正在逐步成为 Internet 应用的主力开发语言,成为 Internet 上的世界语。甚至有些分析家惊呼,Java 哪里还是什么程序设计语言,它简直是一个操作系统,彻底地改变了应用软件的开发模式。的确,Java 带来了自 PC 机以来又一次技术革命,为迅速发展的信息世界增添了新的活力。

Sun 的 Java 语言开发小组成立于 1991 年,其目的是开拓消费类电子产品市场,例如,交互式电视、烤面包箱等。Sun 内部人员把这个项目称为 Green,那时 World Wide Web 还在图纸上呢。该小组的领导人是 James Gosling,41 岁,于 1984 年加盟 Sun Microsystem 公司,以前在一家 IBM 研究机构工作。他是 Sun NeWs 窗口系统的总设计师,虽然当时 NeWs 在技术上非常成功,但在商业上却完全失败,没有 Java 那么好的运气。Gosling 是一位非常杰出的程序员,GOSMACS 也是 Gosling 写的,它是第一个用 C 实现的 EMACS 的文本编辑器。

在研究开发过程中,Gosling 深刻体会到消费类电子产品和 workstation 产品的开发哲学之间的差异:消费类电子产品要求可靠性高、费用低、标准化、使用简单,而 workstation 用户要求强大的计算能力,不在乎价格以及操作的复杂性。消费类电子产品用户并不关心 CPU 的型号,也不欣赏专用昂贵的 RISC 处理器,他们需要建立在一个标准基础之上,一系列可选的方案,从 8086—80586 都可以选取。

从 C++ 开始

为了使整个系统与平台无关,Gosling 首先从改写 C++ 编译器着手。但是 Gosling 在改写过程中感到仅 C++ 是无法满足需要的,于是在 1991 年 6 月份开始准备开发一个新的语言,那么给它起一个什么名字呢? Gosling 回首向窗外望去,看见一棵老橡树,于是建一个目录叫 Oak,这就是 Java 语言的前身(后来发现 Oak 已是 Sun 公司的另一个语言的注册商标,才改名为 Java,即爪哇,太平洋上一个盛产咖啡的岛屿的名字)。

Gosling 在开始写 Java 时,并不局限于扩充语言机制本身,更侧重于语言所运行的软硬件环境。他要建立一个系统,这个系统运行于一个巨大的、分布的、异质的网络环境中,完成各电子设备之间的通信与协同工作。Gosling 在设计中采用了虚机器码(Virtual Machine Code)方式,即 Java 语言编译后产生的是虚机器码,虚机器码运行在一个解释器上,每一个

操作系统均有一个解释器,这样一来,Java 就成了平台无关语言。这和 Gosling 设计的 Sun NeWs 窗口系统有着相同的技术味道,在 NeWs 中用户界面统一用 Postscript 描述,不同的显示器有不同的 Postscript 解释器,这样便保证了用户界面的良好的可移植性。

当 Patrick Naughton 加入该小组后,整个工作进展神速。Naughton 也是 Sun 公司的技术骨干,曾经是 Open Windows 项目的负责人。经过 17 个月的奋战,整个系统胜利完成。它是由一个操作系统、一种语言(Java)、一个用户界面、一个新的硬件平台、三块专用芯片构成的。通常情况下,这样的项目在 Sun 公司要 75 个人干三年。项目完成后,在 Sun 公司内部做了一次展示和鉴定,观众的反应是:在各方面都采用了崭新的、非常大胆的技术。许多参观者对 Java 留下了非常深刻的印象,特别得到 Sun 的两位领导人 Scott McNealy 和 Bill Joy 的关注,但 Java 的前途未卜。

初战告败

接着,Java 项目的工作人员开始做应用规划,着手成立一家公司,从事消费类电子产品的开发,这家公司就是 FirstPerson Inc.。公司的第一个项目就是 Time-Warner 的投标项目——顶置盒式操作系统和请求式电视技术。虽然 FirstPerson Inc. 在技术上毫不逊色于任何公司,但是由于经营策略上的失误,败给了 SGI。FirstPerson 一直到 1994 年初仍在坚持开发专用的顶置盒式操作系统,但实践证明,这个市场尚未成熟。例如,好几家交互式有线电视网的实验都以失败告终。

Naughton 立即认识到研究方向应该转移。在 Naughton 的努力下,研究开始向在线服务、CD-ROM 和基于网络的台式机方向转移。最后 FirstPerson 解散,一部分人去了 Sun Interactive 开发数字视频数据服务器,而另一部分开始把 Java 用于基于网络的台式机系统。

Java 语言的转折点

到了 1994 年,WWW 已如火如荼地发展起来。Gosling 意识到 WWW 需要一个中性的浏览器,它不依赖于任何硬件平台和软件平台。它应是一种实时性较高、可靠安全、有交互功能的浏览器。于是 Gosling 决定用 Java 开发一个新的 Web 浏览器。

这项工作由 Naughton 和 Jonathan Payne 负责,到 1994 年秋天,完成了 WebRunner 的开发工作。WebRunner 是 HotJava 的前身,这个原型系统展示了 Java 可能带来的广阔市场前景,给 SunLab 两位领导: Bert SutherLand 和 Eric Schmidt 留下了深刻印象,这一点非常重要,因为这两个人是 Sun 的首席技术官员。特别是 Sun 的总裁 McNealy 对 Java 抱有坚定的信念,为 Java 进入 WWW 立下了汗马功劳。正如 McNealy 所料,WebRunner 改名为 HotJava,并于 1995 年 5 月 23 日发表后,引起了产业界巨大的轰动,Java 的地位也随之而得到肯定。McNealy 卓越的洞察力和果断的决策,挽救了这项差点锁进文件柜的技术。

又经过一年的试用和改进,Java 1.0 版终于在 1996 年年初正式发表。

1.2 语言特点

Java 语言的白皮书是这样描述 Java 的：

Java 是一个简单的、面向对象的、分布的、解释的、健壮的、安全的、独立于平台的、可移植的、高性能的、多线程的以及动态的程序语言。

似乎程序设计语言的优点,Java 都占全了。从实事求是的角度,Sun 还漏掉了一个特点,就是 Java 的运行速度比较慢。这往往是解释性语言难以摆脱的阴影。下面我们就结合与 C++ 的比较,逐一讨论 Java 的上述特点。

简单性

Java 是一种简单的语言。何谓语言的简单性(Simple)? 它首先表现在易于学习,这就要求语言中的概念尽量少,并且这些概念应尽量为广大程序员所熟悉。Java 在形式上和 C、C++ 极为相似。其次是运行系统小,Java 的基本解释器只有 40K,加上标准库和线程支持也不过 215K 而已。

为了保证语言的简单性,Java 设计者们删除了 C 和 C++ 中的许多语言功能。这些功能往往很少被使用,或者极易导致不良的程序设计。例如,Java 不支持 goto 语句,取而代之的是带有标号的 break 和 continue 语句以及异常(exception)处理。Java 不支持头文件,取消了 C 的预处理器。为了保证 Java 具有更为统一、纯正的面向对象风范,Java 取消了 C 的 struct 和 union 语句,还有 C++ 的运算符重载和多继承也被排除在 Java 之外。Java 为了追求语言简单性的另一惊人之举,是把 C 程序员推崇倍至的指针取消了。指针实际上是 C 和 C++ 最容易出错的方面,往往有时出了错很难定位,例如有一段 C 程序:

```
/* 上面有若干行 C 代码 */
int *p, ;
p = (int *)malloc(10 * sizeof(int));
for (i=0; i<end; i++) //end 是一个变量
    *p++=0
/* 下面有若干行 C 代码 */
.....
a = index[n];    /* index 是一个整型数组 */
```

如果这段程序在前面不对 end 取值做检查,则极容易出错。当 end>10 时,数组越界,在 Unix 中可能出现 Segment Violation 错误,在 DOS 和 Windows 环境中就可能“死机”。注意,这里只是可能,而我们往往会遇到这样的情况,这段程序在调试工具里运行正常,一旦出了调试工具就不行。

我们常常会为这种情况伤透脑筋,就拿上面一段程序为例,在调试工具中,p[10]存储是一段不重要的数据,所以不会出错。而出了调试工具,p[10]存储一个重要 int 型变量 n 的值,这一段程序破坏了 n 的值,从而导致 index[n]访问错误,我们一般很难发现是 p[10]修改 n 的值。

由于 Java 取消了 struct, 数组和字符串都被当作对象, 无需再保留指针机制。Java 自动处理对象的引用(reference)和递引用(dereference)。另外, 自动无用单元收集也是 Java 的重要特色, 它把程序员从复杂的内存管理中解放出来。恰恰是由于 Java 采取了上述措施, 使程序员彻底从指针悬挂、非法指针引用、内存丢失的困扰中解脱出来。

面向对象

Java 是一个面向对象(Object-Oriented)的程序设计语言, 这意味着程序员着手考虑的是对象中的数据和定义在数据上的操作, 而不仅仅是过程。对于 C 程序员, 使用 Java 必须要改变思考问题的方式。

Java 完全具备面向对象的四大特点: 封装、继承、多态和动态。Java 的封装性比 C++ 好, Java 无全程变量, 无主函数 main。在 Java 中绝大部分成员是对象, 只有简单数字类型、字符类型、布尔类型例外。Java 提供给用户一系列的类(class), 一个或多个 class 可以组成一个包(package)。Java 的 class 和 C++ 的一样有层次结构, 子类可以继承父类属性和方法。Java 类中的方法均缺省为虚函数。Java 的类是可以动态加载的。

由于 Java 是比较纯粹的面向对象语言, 它没有支持 C++ 的模板(template)机制。我们大家都熟悉, 在 C++ 中可以定义一个模板类 Stack<T>, 实例化出多种类型, 例如, Stack<int>, Stack<float>等。虽然 Java 不支持模板, 但在 Java 中所有类是 Object 的子类, 这样一来, 我们可以定义 Stack 操作的对象是 Object 类的对象, 至于 Object 类的对象的实际类型则由语言运行时动态确定。

运算符重载(operator overload)在 C++ 中一向被认为是一种优雅的多态机制, 但在实践中, 人们发现, 运算符重载会使程序变得难以理解。经过激烈的争论后, Java 设计者最后取消了运算符重载, 仅在字符串连接运算中留了一些运算符重载的遗迹, 例如用“+”实现字符串连接。

分布式

Java 支持网络上的应用程序, 是一种分布式(Distributed)程序设计语言。Java 提供了一个 java.net 包, 通过这个包中的类, 可完成各种层次上的网络连接。例如, URL 类支持 Java 应用程序通过 Internet 打开和访问远程对象。使用 Java, 打开远程文件和打开本地文件几乎一样简单。Java 提供一个 Socket 类, 这个类可以提供可靠的流式网络连接。这样, 我们可以非常方便地创建分布式的 Client 和 Server 应用程序。

解释型

Java 编译器产生的是字节代码(byte-code), 而不是特定的机器码, 是一种解释型(Interpreted)语言。Java 程序的字节代码必须运行在一个解释器上, 因此, Java 是一种解释型语言。Java 的字节代码是一种与平台无关的对象文件格式, 它可以高效地在不同平台之间