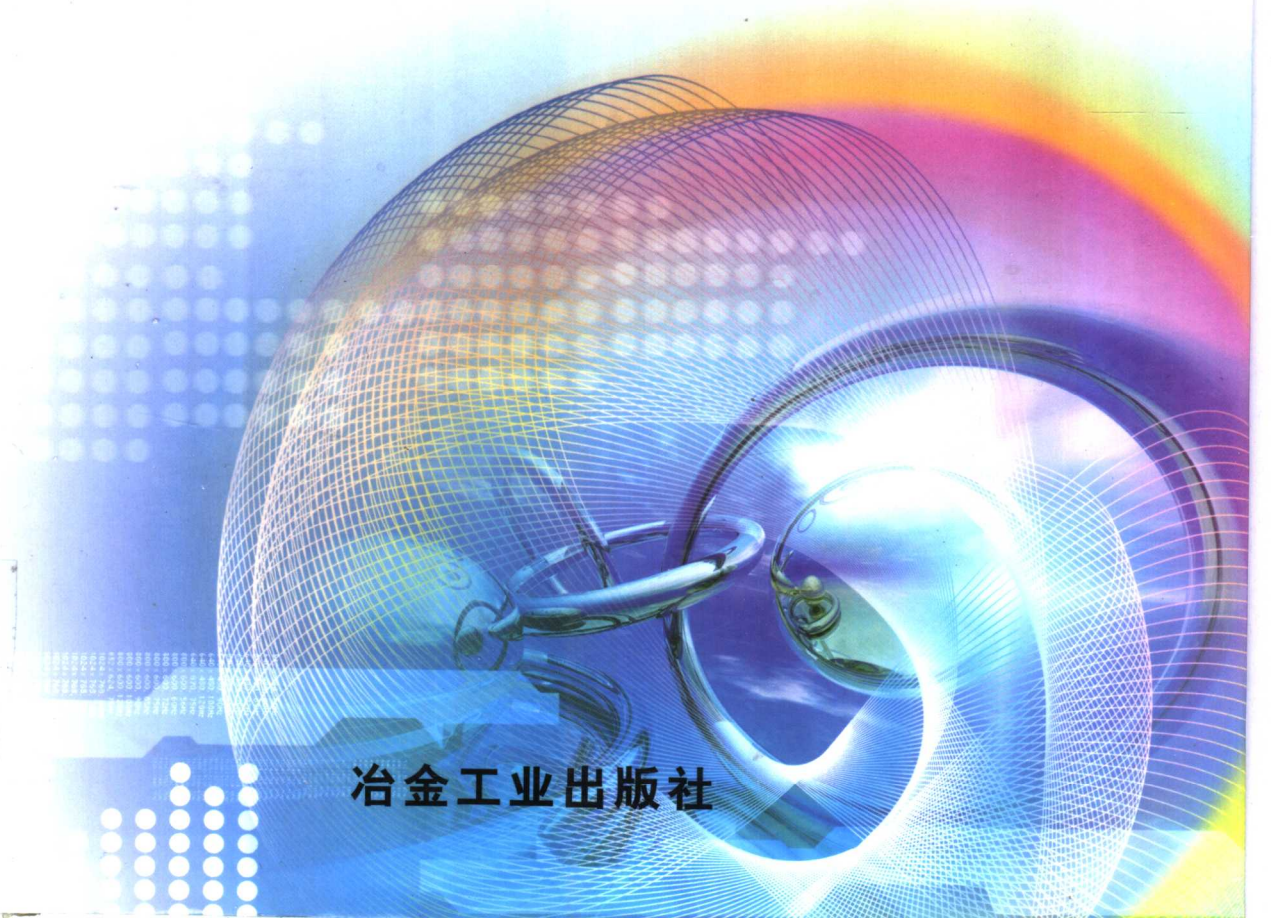
Practical...

Java 函数实用手册

Functional Handbook

张曜　郭立山　游泳明　编著



冶金工业出版社

Java 函数实用手册

张　曜　郭立山　游泳明　编著

北　京

冶 金 工 业 出 版 社

2003

内 容 简 介

Java 是 Sun 公司推出的面向对象的程序设计语言。本书涵盖了 Java 的核心类库。对 Java 平台 1.4 版本的核心类库做了详细、全面而系统的描述，主要包括包描述和类描述，并从类的层次结构示意图、成员变量、成员函数等几方面做了详细的说明。此外，本书辅以示例对类的典型成员函数的使用做了介绍，包括容器工具包、文本字符格式工具包、语言基础包、I/O 系统包、窗口制作和小程序包、网络编程的 net 包、JDBC 和数据库操作的 sql 包、基于 Web 应用的 servlet 包和基于分布式的远程函数调用的 rmi 包。

本书内容丰富，层次清晰，以典型的示例程序和精辟的提示说明，让读者迅速地掌握函数的具体用法，并在附录中提供了 Java 的相关术语，方便读者查找相关函数的用法。

本书可作为初、中级 Java 编程员和从事 Java 开发的技术人员的学习和参考用书，也可作为各大、中专院校相关专业和 Java 函数培训班的参考用书。

图书在版编目（CIP）数据

Java 函数实用手册 / 张曜等编著. —北京：冶金工业出版社，2003.8
ISBN 7-5024-3309-0

I. J… II. 张… III. JAVA 语言—程序设计
IV. TP312

中国版本图书馆 CIP 数据核字（2003）第 058213 号

出版人　曹胜利（北京沙滩嵩祝院北巷 39 号，邮编 100009）
责任编辑　程志宏
湛江蓝星南华印务公司印刷；冶金工业出版社发行；各地新华书店经销
2003 年 11 月第 1 版，2003 年 11 月第 1 次印刷
787mm × 1092mm　1/16；　33 印张；　1589 千字；　516 页；　1-2500 册
65.00 元
冶金工业出版社发行部　电话：（010）64044283　**传真：**（010）64027893
冶金书店　地址：北京东四西大街 46 号（100711）　**电话：**（010）65289081
（本社图书如有印装质量问题，本社发行部负责退换）

前　言

一、关于本书

Java 是 Sun 公司推出的面向对象的程序设计语言，它广泛地应用在字符串处理、绘图、数学计算、数据库访问、网络信息处理等方面。由于 Java 的核心类库函数数量庞大，也给 Java 程序员们的工作带来了一些不便，广大 Java 用户迫切需要一本内容全面、查询方便的 Java 函数实用手册作为工具型的参考书。

同时，Java 专业的广大师生在学习和工作中也有类似的需要。然而目前市面上这类工具型的参考书并不多见。为此，本书作者集近年来 Java 语言的教学和工作经验，按照 Java 函数的应用领域、功能的不同进行分类、汇总，并对每个函数的功能、兼容性、语法、应用示例等几方面作了详细地介绍。

本书内容全面、分类合理，因而对 Java 不同层面的用户都有很大的参考价值。作者也希望本手册能成为广大 Java 用户工作中的得力助手。

二、本书的结构

本书共分 9 章和两个附录，其内容安排如下：

第 1 章：概述。主要介绍了 Java 的相关背景知识以及 Java 的特点、Java 环境设定、Java 常用开发工具。

第 2 章：常用工具。主要介绍了 Java 容器工具、Java 文本字符格式工具、Java 语言基础类、Java 日志记录处理类以及 Java 压缩打包工具类。

第 3 章：I/O 系统。主要介绍了流操作超类、流转换器、过滤流、缓冲流、文件操作、序列化、内存流、管道、语法分析流、nio 包等内容。

第 4 章：窗口制作和 Applet。主要介绍了 Applet 相关类、AWT 窗口编程核心类库、AWT 窗口编程事件模型。

第 5 章：网络编程。主要介绍了 Authenticator 类、ContentHandler 类、ContentHandlerFactory 接口、DatagramPacket 类、DatagramSocket 类、DatagramSocketImpl 类、DatagramSocketImplFactory 接口、FileNameMap 接口、HttpURLConnection 类、InetAddress 类、JarURLConnection 类、PasswordAuthentication 类、ServerSocket 类、Socket 类、SocketImpl 类、SocketImplFactory 接口、SocketOptions 接口、SocketPermission 类、URLClassLoader 类、URLDecoder 类、URLEncoder 类、URLStreamHandler 类等内容。

第 6 章：JDBC 与数据库的相关操作。主要介绍了 SQL 接口和 SQL 类。

第 7 章：Web 应用。主要介绍了通用 SERVLET 类库以及与 HTTP 协议结合应用 SERVLET 类。

第 8 章：远程函数调用。主要介绍了核心类库、服务器类库以及 RMI 注册。

第 9 章：安全与权限。主要介绍了核心类库和主要权限讲解。

附录 A：网络资源。

附录 B：Java 术语。

三、本书的特点

本书根据 Java 语言的应用范围对类库函数进行分类，形成两层体系结构：第一层目录按 Java 各类库函数的应用功能来进行划分；第二层目录则按各类库函数更细致的功能进一步划分。因此读者只要知道函数的大致功能就可以按功能定位到具体的函数位置。同时为了进一步增强本函数手册的查询功能，本书也在附录部分添加了各个知识点的字母排序目录，方便读者查找相关函数的用法。利用这两种查询方法，读者可以很方便地对自己想了解的知识点进行查询。

四、本书的适用对象

本书内容全面，深入浅出，不仅可以作为初、中级 Java 编程员和从事 Java 开发的技术人员的学习和参考用书，也可作为各大、中专院校相关专业和 Java 函数培训班的参考用书。

读者在阅读本书的过程中如遇到疑难问题或有好的意见和建议可发 E-mail 至 **service@cnbook.net**，也可登录网站：**http://www.cnbook.net**，在该网站的论坛进行探讨（本书的源代码也可以从该网站下载）。

由于作者水平有限，编写时间仓促，书中错漏之处在所难免，敬请广大读者批评指正。

编　者

2003 年 8 月

目　录

第1章 概 述

本章提要

- Java 简介
- Java 的特点
- Java 环境设定
- Java 常用开发工具
- 本手册内容概述

本章主要介绍 Java 的简介、特点、环境设定以及 Java 常用开发工具，并附有本手册内容概述。

1.1 Java 简介

Java 是 Sun Microsystems 公司在 1995 年中推出的一套程序语言兼平台。通常以 JDK（Sun 所开发的一套 Java 开发工具）的版本来定义 Java 的版本。JDK 1.0 版于 1996 年初公开，JDK 1.1 版于 1997 年初公开，JDK 1.2 版于 1998 年底公开。基于市场行销的考量，Sun 在 JDK1.2 版公开后旋即将 Java 改名为「Java 2」，将 JDK 改名为「Java 2 Software Development Kit（以下简称 J2SDK）」。J2SDK（原称 JDK）1.3 于 2000 年 4 月公开，此版本仍称做「Java 2」。目前 J2SDK 1.4 已经发布。

Java 技术根据硬件平台与适用环境的差异，分成几个分支。JDK 1.1 的时代，适用于一般消费性电子产品等，嵌入式系统的 Java 平台是 PersonalJava 与 EmbeddedJava，此二者并无明确的界线，大致上来说，运算资源、内存以及显示装置比较丰富者，使用 PersonalJava，例如 Set-Top Box、视讯电话...等；反之，资源较有限者使用 EmbeddedJava，例如呼叫器、移动电话...等。除了 PC 使用的 Java 平台、IA 使用的 PersonalJava 与 EmbeddedJava 平台之外，JavaCard 也是一个 Java 平台，使用于 Smart Card（IC Card）上。

Java2 出现后，推翻了先前的 PersonalJava 与 EmeddedJava 的分法，改分成 Java 2 Platform Enterprise Edition（简称 J2EE）、Java 2 Platform Standard Edition（简称 J2SE）、Java2Platform Micro Edition（简称 J2ME）。J2EE 适用于服务器，目前已经成为企业运算、电子商务等领域中相当热门的技术；J2SE 适用于一般的计算机；J2ME 适用于消费性电子产品。除了这三者之外，JavaCard 依然是独立的一套标准。

1.2 Java 的特点

1.2.1 平台无关性

平台无关性是指 Java 能运行于不同的平台。Java 引进虚拟机原理，并运行于虚拟机，在 Java 接口之间实现不同平台的交流。使用 Java 编写的程序能在世界范围内共享。Java 的数据类型与机器无关，Java 虚拟机（Java Virtual Machine）是建立在硬件和操作系统之上，实现 Java 二进制代码的解释执行功能，提供于不同平台的接口的。

1.2.2 安全性

Java 的编程类似 C++，学习过 C++的读者将很快掌握 Java 的精髓。

Java 舍弃了 C++的指针对存储器地址的直接操作，程序运行时，内存由操作系统分配，这样可以避免病毒通过指针侵入系统。Java 对程序提供了安全管理器，防止程序的非法访问。

1.2.3 面向对象

Java 吸取了 C++面向对象的概念，将数据封装于类中，利用类的优点，实现了程序的简洁性和便于维护性。类的封装性、继承性等有关对象的特性，使程序代码只需一次编译，然后通过上述特性反复利用。程序员只需把主要精力用在类和接口的设计和应用上。Java 提供了众多的一般对象的类，通过继承即可使用父类的方法。在 Java 中，类的继承关系是单一的而非多重的，一个子类只有一个父类，子类的父类又可以有一个父类。Java 提供的 Object 类及其子类的继承关系如同一棵倒立的树形，根类为 Object 类，Object 类功能强大，经常会使用到它及其派生的子类。

1.2.4 分布式

Java 建立在扩展 TCP/IP 网络平台上。库函数提供了用 HTTP 和 FTP 协议传送和接受信息的方法。这使得程序员使用网络上的文件和使用本机文件一样容易。

1.2.5 健壮性

Java 致力于检查程序在编译和运行时的错误。类型检查帮助检查出许多开发早期出现的错误。Java 自己操纵内存减少了内存出错的可能性。Java 还实现了真数组，避免了覆盖数据的可能。这些功能特征大大提高了开发 Java 应用程序的效率。Java 提供：Null 指针检测、数组边界检测、异常出口、Byte code 校验。

1.3 Java 环境设定

Java 虚拟机（JVM）借助类装载器装入应用程序使用的类，具体装入哪些类根据当时的需要决定。CLASSPATH 环境变量告诉类装载器到哪里去寻找第三方提供的类和用户定义的类。另外，也可以使用 JVM 命令行参数-classpath 分别为应用程序指定类路径，在-classpath 中指定的类路径覆盖 CLASSPATH 环境变量中指定的值。

类路径中的内容可以是：文件的目录（包含不在包里面的类），包的根目录（包含已打包的类），包含类的档案文件（比如.zip 文件或者.jar 文件）。

在 Unix 家族的系统上，类路径的各个项目由冒号分隔，在 MS Windows 系统上，它们由分号分隔。

类装载器以委托层次的形式组织，每一个类装载器有一个父类装载器。当一个类装载器被要求装载某个类时，它在尝试自己寻找类之前会把请求先委托给它的父类装载器。系统类装载器，即由安装在系统上的 JDK 或 JRE 提供的默认类装载器，通过 CLASSPATH 环境变量或者-classpath 这个 JVM 命令行参数装入第三方提供的类或者用户定义的类。系统类装载器委托扩展类装载器装入使用 Java Extension 机制的类。扩展类装载器委托自举类装载器（bootstrap class loader）装入核心 JDK 类。

必须特别注意的是，类装载器装入类的次序就是类在

classpath 中出现的次序。类装载器从 classpath 的第一项开始，依次检查每一个设定的目录和压缩文件，尝试找出待装入的类文件。当类装载器第一次找到具有指定名字的类时，它就把该类装入，classpath 中所有余下的项目都被忽略。

例子：

有一个类是 www.Rose，编译好后怎么存放呢?

答案是，可以在 d：盘的 java_class 目录下建立一个 www 子目录，然后把 Rose.class 文件复制到该子目录下：

d：\java_class\www\Rose.class

可以把它们的一组类打包发行，把这个包压缩成 Jar 或 Zip，设置的方法如下：

set classpath="d：\java_class;d：\oracle\ora.jar"

只需把文件名包含到 classpath 中去即可正确地引用该压缩包中的类了。

对于 jdk1.3 以上的版本，引用 java.*，sun.*等系统包不需要设置 classpath.

使用当前目录下的类需要将点包含在 classpath 中，如下所示：

set classpath=".;d：\java_class"

注意：与设置 path 一样，不同路径之间用分号分隔开。

1.4 Java 常用开发工具

1.4.1 集成开发工具

这类工具提供了 Java 的集成开发环境，为那些需要集成 Java 与 J2EE 的开发者、开发团队提供对 Web applications，servlets，JSPs，EJBs，数据访问和企业应用的强大支持。现在的很多工具属于这种类型，也是 Java 开发工具的发展趋势。

首先介绍文本编辑工具，很多初学者都是从 jdk+text editor 开始的.

1. 文本编辑器 UltraEdit（http://www.ultraedit.com/）

UltraEdit 是现在文本编辑器中的优秀代表，它不但可以编辑文本，还可以编辑十六进制代码。

主要特性：

（1）可以打开多个文件，文件大小无限制，每个文件都会有一个页框，非常直观。

（2）既可以记住最近使用的文件，也可以加入到 favorate 文件表中，还可以建立一个项目文件，把相关文件组织起来。

（3）能保持代码的缩进，在任何时候，行号都会在窗口的状态栏里显示，还可以在 view 菜单中选择是否每行显示行号。

（4）可通过配置为不同代码设置不同的颜色。

（5）可以搜索和替换打开的所有文件。

（6）支持多级的撤销和恢复。

类似产品：EditPlus（http://www.editplus.com/）。

2. Jbuilder（http://www.borland.com/jbuilder/）

Jbuilder 是目前最好的 Java 开发工具之一，在协同管理、对 J2EE 和 XML 的支持等方面均走在其他产品的前面。

主要特性：

（1）提供与 Tomcat 集成，使 Web 开发更容易。

（2）提供了对企业应用的开发功能，可以集成多种应用服务器。

（3）提供了更简单的程序发布功能，所有的应用都可以打包。

（4）提供了团队开发能力，可以集成多种版本控制产品。

3. WebGain（http://www.webgain.com/）

它是由 VisualCafe 发展而来，现在又提供了对 EJB 开发的支持，实现了窗口的 SDI 模式。

4. WebSphereStudio（http://www.900ibm.com/developerWorks/cn/wsdd/zones/studio/index.shtml）

WebSphere Studio Application Developer 提供了创建，开发，测试和管理所有 Web 及企业范围的 J2EE 应用的工具。可定制的透视图使 Web 开发者、Java 程序员、EJB 开发人员和管理者共享同一个开发工具。它的核心是应用创建工具、编辑器和向导工具帮助快速开发 J2EE 资源，如 HTML 文件，JSP 页面，Java 类和 servlets，EJB beans 和 XML 描述语言等。用户可以按照 J2EE 规范中定义的模块来组织这些资源到一个项目中。一旦资源创建完成，就可以方便地在开发环境中或输出到远程的服务器上测试和调试它们。

5. VisualAgeforJava（http://www.7b.software.ibm.com/wsdd/zones/vajava/）

VisualAge for Java 可以很好地与 IBM 的其他产品进行集成，可以很好地开发 Java 和 J2EE 的应用。

6. Eclipse（http://www.eclipse.org/）

Eclipse 是替代 IBM Visual Age for Java（以下简称 IVJ）的下一代 IDE 开发环境，但它未来的目标不仅仅是成为专门开发 Java 程序的 IDE 环境，根据 Eclipse 的体系结构，通过开发插件，它能扩展到任何语言的开发，甚至能成为图片绘制的工具。目前，Eclipse 已经开始提供 C 语言开发的功能插件。更难能可贵的是，Eclipse 是一个开放源代码的项目，任何人都可以下载 Eclipse 的源代码，并且在此基础上开发自己的功能插件。也就是说未来只要有人需要，就会有建立在 Eclipse 之上的 COBOL，Perl，Python 等语言的开发插件出现。同时可以通过开发新的插件扩展现有插件的功能，比如在现有的 Java 开发环境中加入 Tomcat 服务器插件。可以无限扩展，而且有着统一的外观、操作和系统资源管理，这也正是 Eclipse 的潜力所在。但是现在 Eclipse 还没有支持对 EJBs 的开发。

虽然目前 Eclipse 项目还没有最后完成，但从已有的版本中已经能领略到 Eclipse 设计主导思想和主要功能特点。现在就了解 Eclipse 不但使广大程序员对这款业界期望很高的 IDE 能一睹为快，更为重要的是如果能参加到 Eclipse 项目的开发中或是阅读它的开放源代码，这对广大程序员来说无疑是一个千载难逢的提高编程水平的好机会。Eclipse 计划提供多个平台的版本，象 Windows，Linux，Solaris，HP-UX 和 AIX，以下只介绍 Windows 版本。

主要特性：

（1）很方便地对源文件进行导入和导出。

（2）源代码的管理更加随心所欲。

（3）支持团队开发。

（4）支持插件开发功能。

Eclipse 最有魅力的地方就是它的插件体系结构。在这个体系中重要的概念是扩展点（extension points），也就是为插件提供的接口。每一个插件都是在现有的扩展点上开发，并可能还留有自己的扩展点，以便在这个插件上继续开发。

由于有了插件，Eclipse 系统的核心部分在启动的时候要完成的工作十分简单：启动平台的基础部分和查找系统的插件。在 Eclipse 中实现的绝大部分功能是由相应的插件完成的，比如 WorkBench UI 插件完成界面的外观显示，Resource Management 插件完成维护或生成项目或文件等资源管理工作，而 Version and Configuration Management（VCM）插件则负责完成版本控制功能等等。虽然以上提到的每一个功能都是绝大多数 IDE 环境所必备的功能，Eclipse 却也把它们都做成了插件模式，甚至用来开发 Java

程序的开发环境（Java development tooling，JDT）也只不过是Eclipse 系统中的一个普通插件而已。整个 Eclipse 体系结构就像一个大拼图，可以不断的向上加插件，同时，现有插件上还可以再加插件。

1.4.2 应用服务器

1. WebSphereApplicationServerEnterpriseEditionv3.0(www.ibm.com)

可互操作的 WebSphere Application Server 产品系列在工业标准的基础上提供下一代应用程序服务器。IBM WebSphere Application Server 系列分为四个 V5 的软件包，以便更好的处理用户的要求。每个版本处理不同的方案集和需要。WebSphere Application Server 包含：

WebSphere Application Server Express。

此版本是静态内容、servlet 和 JSP 页面的轻量级服务器，但不支持企业 bean。

WebSphere Application Server。

此版本处理基本编程和桌面开发者的执行需要以及单个服务器生产方案。此版本的执行环境为 Web 和基于组件的编程以及 Web 服务处理基于标准的编程。

此版本的管理模型假设单个服务器环境没有故障转移或工作负载平衡的群集，也没有多个服务器实例的集中式管理。然而，用户可以在安装控制单元的下一个产品后的任何时间，将独立节点添加到集中管理的网络（单元）。

WebSphere Application Server Network Deployment。

此版本在部门计算方案中处理应用程序服务器执行。它提供多个服务器实例的集中式管理，以及基本群集和高速缓存支持。

WebSphere Application Server Enterprise。

此版本为按核心设计的应用程序、Java 2 平台，企业版（J2EE）的基于标准的编程模型和 Web 服务处理大信息技术（I/T）生产方案。它支持大规模群集、高速缓存、内容分布和动态工作负载管理来帮助在 I/T 计算中心中获取共享资源的有效使用率。它还为通过为业务流程管理、集成适配器、规则管理和消息变换引用超标准编程功能集成应用程序和业务行处理复杂的编程要求。

WebSphere Application Server（基本）产品安装映像包含核心应用程序服务器运行时、本机 JMS 提供程序（嵌入式消息传递功能部件）、IBM HTTP Server、IBM Developer Kit、IBM Cloudscape、XML 和 XSL 解析器、应用程序组装工具（AAT）、部署工具，当 Node Agent 是单元的一部分时与 Deployment Manager 进行通信，和代理高速缓存启动的外部适配器库。

2. ApacheJServ（java.apache.org）

Apache JServ 是一个纯粹的 Java servlet 引擎，完全符合 JavaSoft Java Servlet APIs 2.0 规范。它可以工作在任何“version 1.1 compliant”的 Java 虚拟机上，当然也支持 2.0。在这个版本中只有一个 Apache Web Server 的模块。

虽然 apache jserv 的开发小组现在都从事 tomcat 的开发，而且 apache jserv 只支持 servlet 2.0 的标准，并且要加其他的软件（gnujsp）才能支持 jsp；但是 apache jserv 在现阶段比 tomcat 稳定，因此现在 apache jserv+gnujsp 比较适合应用于 web。

3. WebLogicServer

它负责管理基于 Java 的企业应用的服务器。主要用于提高开发人员的生产力，减轻管理的复杂度，为集成应用提供强化服务。本服务器充分支持跨越 EJB，JMS，J2CA，JDBC 以及其他与 XA 协议兼容组件之间的交易，并实现了 Java Transaction API 以支持其强大的交易基础功能。

WebLogic Server 7.0 包括：

（1）遵从 J2EE 1.3 的标准。

（2）实现了最新的网络服务标准。

（3）确保 JMS 应用的高可用性和高扩展性。

（4）BEA WebLogic Builder——一个新的、方便使用的图形化工具，能够将 J2EE 应用组装，打包并部署到 WebLogic Server 上。

（5）与领先的开发工具结合紧密。

（6）智能编译实用程序——能够生成可部署的 J2EE 组件。

（7）新的实用程序和运行环境为创建、调试以及访问网络服务提供了便利条件。

（8）新的安全框架减轻了强调安全的应用在编写代码方面的负担。

4. Borland AppServe

Java 2 Enterprise Edition（J2EE）是目前市场上最受重视的企业应用程序开发标准。BorlandAppServer4.5 则是可靠的、具延展性的高性能 J2EE Application Server，它结合了 VisiBrokerCORBA 引擎的强大威力与 J2EE 简洁的企业级架构。

BorlandAppServer4.5 已经应用在必需要有延展性、高流量的交易处理以及 24x7 都能运作的电信、银行与金融业之基础架构上，并广获如 Bank of America、Telecordia、Deutche Bank 、UUNET、CISCO Systems…等全球性跨国企业采用。

先进的 Borland AppServer 4.5 架构让软体开发者全心放在建构具有企业逻辑的 EJB。位于底层的 VisiBroker 引擎所组成的基础架构提供了广为业界所认可的 IIOP 协定通讯机制以及诸如 Portable Object Adapter（POA）、Objects-by-value（OBV）与 RMI-over-IIOP 等最新 CORBA 标准的强大威力。未来 J2EE 标准会朝向全面结合 CORBA/IIOP 来发展以便满足 Application Server 与 CORBA 式系统之间作紧密的交互运算的需求。而 Borland AppServer 4.5 现在就提供这项技术。

5. Sybase EAServer

Sybase EAServer4.0 使用了支持 Java 2 Enterprise Edition（Java 2 企业版）1.3 版的先进技术，是一个高性能、可伸缩、安全、开放的应用服务器，它适用于多层架构的电子门户和互联商务解决方案。

它对各种工业标准提供广泛的支持，符合基于组件的多层体系结构，并且在它的最新版本中加强了对 PowerBuilder 组件和 Enterprise JavaBeans（EJBs）的深层支持。这样，用户可以运用它提供的非常灵活的开发能力，充分利用多样化的计算环境，建立更加高效的企业 Web 应用系统。

Enterprise Application Server 的主要特点：

（1）支持所有主要的组件标准。Enterprise Application Server 支持多种组件模型，同一应用中可以结合使用各种组件。

EAServer 支持下面的组件模型：

Java 和 Enterprise JavaBeans。

COM 和 DCOM。

CORBA。

C 和 C++。

支持多种分布式协议。

IIOP 协议，用于分布式组件调用和 CORBA 组件互操作。

HTTP 协议，用于 WEB 访问。

ISAPI 协议，用于扩展 Microsoft WEB 服务器。

NSAPI 协议，用于扩展 Netscape WEB 服务器。

CGI 协议，用于与其他 WEB 服务器实现互操作。

所有符合 SSL 安全标准的协议。

TDS（Tabular Data Stream）协议，用于 Sybase 产品间高效通信。

（2）全面支持 J2EE 1.3。Sybase EAServer 全面支持 J2EE 1.3 标准。借助对 J2EE 的支持，EAServer 降低了开发人员编写大型应用（比如交易管理、生命期管理、资源合并等）的复杂度，因为它在应用服务器上已经提供了诸多现成的服务。J2EE 1.3 为 EAServer 带来的好处包括：

Enterprise JavaBeans 2.0——消息驱动的 bean 以及对 Container Managed Persistence（容器-管理持续性）的增强。

JSP1.2/Servlet 2.3——利用 XML 及嵌入脚本编写 JSP，用 Java 动态生成 XML 以及 HTML 页面。

JAXP——用于 XML 语法解析的 Java API。

JAAS——基于 Java 的安全接口允许使用可插接的身份验证组件。

JMS 1.0.2——EAServer 将标准的 Java 消息接口作为现有服务之上的封装。

Connectors（JCA，连接器）——这是一种通用方法，让用户可以将连接连向 SAP 和 PeopleSoft 等传统系统。

JTS——Java 交易服务。

JDMK API——提供通过标准的 SNMP 监视系统连接 Java 应用的接口可实现多层应用开发。

能够有效地管理客户的会话、线索、数据库连接、Web 页、事务流和安全性；可以使用各种标准组件建立分布式业务应用。提供了完善的分布式应用体系结构，解除了开发者的后顾之忧；提供集成化的 4GL、Java 和 Web 工具，使开发者感到灵活和方便。

1.4.3 Java 类库

随着应用领域的不同，Java 有许多 API（Application Programming Interface），这些 API 分成三大类：

（1）Java Core API：由 Sun 制定的基本 API，任何 Java 平台都必须提供。（www.sun.com）

（2）Java Standard Extension API（javax）：由 Sun 制定的扩充 API，Java 平台可以选择性地提供或加装。（www.sun.com）

（3）厂商或组织所提供的 API：由各家公司或组织所提供。

例如：IBM HostAccessLibraryAPIForJava（www.ibm.com）

其中 Core API 和 Standard Extension API 已经逐渐涵盖了大部分的信息应用领域，例如多媒体、数据库、Web、企业运算、语音、实时系统、网络、电话、影像处理、加解密、GUI、分布式运算.....。

1.4.4 Java 中间件

1. SybaseEnterpriseApplicationServer（www.sybase.com）

Sybase 中间件产品

Open Client/Open Server：Open Client 和 Open Server 构成了 Sybase 开放式客户机/服务器互联的基础，为不同数据源以及几百种工具和应用提供了一致的开放的接口，简化了与异构系统的互连。

jConnect for JDBC：jConnect 是一个完全由 Java 实现的、符合 JDBC 标准的 Java 数据库接口。它在多层和复杂的环境中，为 Java 开发者提供一个专用的数据库访问接口。

Replication Server：Replication Server 可在网络上的数据库之间提供可靠、快速的复制功能。它可以在多个数据库之间维护一致性和连续性，可以在 Adaptive Server、非 Sybase、基于 LAN 和主机数据服务器之间复制数据。

2. WebSphereHostOn-Demandv4（www.ibm.com）

IBM WebSphere Studio 基于 Eclipse 开放资源平台，并支持 IBM 全线中间件产品。自此，软件开发人员可以通过采用一整套统一的、集成化工具降低开发成本，提高生产能力。

新工具可与 IBM 软件协同工作，进行包括 Web 服务在内的电子商务数据和应用的部署、保护、集成和管理。值得一提的是，该工具能够支持 WebSphere、Lotus Domino、CrossWorlds、DB2、Tivoli、MQ 和 IBM eServer。

这些工具扩充了 WebSphere Studio 应用开发产品系列，包括 WebSphere Studio Application Developer（WSAD）和 WebSphere Studio Enterprise Developer（WSED）等，并将 IBM 的全部中间件产品统一到单一的开发环境中，以便集成工具。并降低客户的开发费用。IBM 将接触到跨越协作化后端办公室应用、无线、中间市场业务开发以及企业现有应用与 Java/J2EE、XML 集成的业界最广泛的开发者社区。

随着 IBM 的中间件工具的功能被集成到 WebSphere Studio 中，开发者将可以使用单个统一“类似门户”的编程环境开发、测试、调试性能和软件。开发者可以为特定项目定制性能和灵活性稳定的应用开发环境。

这些工具建立在由领先的工具厂商所组成的大型协会所管理的 Eclipse 平台之上。使用基于 Eclipse 的工具，开发者可以由避免集成不兼容的工具所带来的烦琐任务，并消除在多个应用平台中开发应用的耗时过程，从而以更短的时间创建更高质量的应用。

3. ProgressSonicmq（www.sonicmq.com）

ProgressSonicMQ 是一种快速、灵活、可扩充的 Java 消息服务器（JMS）实现技术，其设计目标是简化当前高度分布式企业应用和基于 Internet 的解决方案的开发和集成过程。SonicMQ 完全符合 JMS、百分之百基于 Java，提供了全系列消息特性，包括点到点模型和发布预订模型、有保障的消息交付、全方位安全保证、交易支持及 XML 格式化消息扩展。具体将在后面介绍消息工具的时候提到。

4. Visiobroker

Borland 企业级服务器 VisiBroker 版本是一个用于分布式应用的、完全的 CORBA 环境。它包括 Java、C++和安全特性，它使 Web 客户机能够与 CORBA 对象通信，真正的外部连接。

VisiBroker 版本的所有产品特性都是符合 CORBA2.4 标准的，提供了业界证实的 ORB 运行时间并且支持用于建立、分发和管理开放、灵活和可互操作的分布式应用的开发环境。VisiBroker 技术以确认和开放的工业标准为基础，因此降低了开发商锁定风险。这种本地 IIOP 平台支持对高性能、可靠性和互操作性的要求；另外，它是惟一的能够使 Web 客户机与 CORBA 对象通信的商用产品，真正的外部连接（out of the box）。

1）高可伸缩性

由于它完善的线程和连接管理架构、Smart Agent™技术以及高性能 IIOP 引擎，因而以 VisiBroker 版本为基础的方案具有极佳的可伸缩性。通过集中式计算、企业级环境中的大量用户端和服务器测试得出的工业指标证明了这一点。

2）高可用性

VisiBroker 中的 Smart Agent 技术把由机器故障引起的操作中断降低到最低限度。加上它先进的 CORBA 命名服务特性-集群、负载均衡集群、容错和故障接管功能-VisiBroker 使得大型企业应用能够被安全而可靠地分发。

3）互操作性和灵活性

VisiBroker 支持用于开发和分发的多种平台。与其他符合