

Java 网络程序设计

TCP/IP 与 Internet 程序设计
企业网络程序设计

黄嘉辉 编著

清华大学出版社

前 言

Java 技术具有强大的网络功能和全新的理念。作为一种跨平台的程序设计语言,Java 具备简化的优点,并提供面向对象而又不依赖于机器的开放结构,具有卓越的可移植性、安全性特点,同时具有灵敏的响应和交互能力。

按照 Sun 的定义,Java 技术主要包括 3 个平台:从小型设备的 J2ME、适用于个人计算机以及工作组服务器的 J2SE,到支持企业级应用的 J2EE。Java 的三种平台目前在实际中应用非常广泛,这股趋势似乎会随着 Internet 的兴盛而更加风靡。

由于最初设计出发点就是面向网络,所以 Java 能够完全满足网络应用对便捷性、模块化和安全性的要求。Java 技术可以轻松地连接现有的计算机系统、降低计算成本和加快软件开发。同时,Java 还使企业能够通过 Internet 安全地连接客户、供应商和合作伙伴。这些优势使得越来越多的企业倾向于部署基于 Java 的应用。

本书就 Java 技术中的 TCP/IP 通信协议、企业网络技术、Java 编程思想等 Java 专题与新技术作了深入讨论,详细介绍了 Java 的网络应用以及 Java 在企业网络建设中所起的重要作用。

全书共分上下两篇。

上篇为 TCP/IP 与 Internet 程序设计篇,共 12 章,各章内容要点如下:Java 版本、应用范畴及系统结构(第 1 章);与网络相关的一些基本概论,如 OSI 七层模型、IP 地址、TCP/IP 协议、通信端口和网络结构等(第 2 章);网络应用的核心——Socket,及其应用程序流程(第 3 章);发送电子邮件的 SMTP 协议和接收电子邮件的 POP3 协议(第 4、5 章);客户端和服务端端的 FTP 通信协议及相关内容(第 6、7 章);HTTP 通信协议与 Web 服务器(第 8 章);Java Web 浏览器的有关内容(第 9 章);聊天程序的客户端与服务端端的程序设计(第 10 章);网络游戏的程序设计实例讲解(第 11 章);JSSE 与 HTTPS 服务器(第 12 章)。

下篇为企业网络技术篇,分 8 章内容介绍了企业级软件的 Java 技术解决方案,着重于 Java 专题与新技术的说明。第 13 章概述了企业级软件的构建模式及 Java 应用构架。其余各章详细介绍了 Java 技术实现,包括:Java 档案、Java Plug-in 与 HTML Converter、Java 安全性原则、三层体系结构 Socket 与 JDBC、JavaMail API、Java Web Start、J2EE 连接器架构。

其中值得一提的是 Java Web Start 及 Java 连接器架构,前者将包含于 JDK 1.4 版之中,以实现 Easy Deployment 的概念,后者则是 J2EE 新增的一项标准,以作为应用程序与企业信息系统之间的桥梁。

各种类型的企业级应用软件,从 B2B(企业对企业)、供应链管理、企业应用系统集成,到门户网站等,均有两个共同的特点,即 100% Java 开发及支持 XML。由企业级应用可以看到一个趋势,那就是 Java 并非纸上谈兵或昙花一现的语言、环境或平台,而是真正已实际运用在企业中,并提供企业完善的解决方案。

本书旨在帮助 Java 技术的专业人士全面提升技能，重点为读者打造面向对象的程序设计基础，从而熟练掌握 Java 的应用与开发。书中内容及范例均经笔者再三着墨及测试，为的是能力求完美，并希望读者有所收获，同样地，秉着“分享”的精神，希望读者能不吝指教，以使本书内容更加充实。

目 录

Java 网络程序设计——上篇	
TCP/IP 与 Internet 程序设计篇	
第 1 章 Java 简介	3
1.1 Java 的版本	3
1.2 Java 的应用范畴	6
1.2.1 两层的客户/服务器体系结构	6
1.2.2 基于 HTML 的两层体系结构	7
1.2.3 基于 HTML 的三层体系结构	8
1.2.4 基于 Applet 的三层体系结构	8
1.2.5 Java Web Start 技术	11
第 2 章 网络概论	13
2.1 历史与变革	13
2.2 OSI 七层模型	15
2.3 TCP/IP	17
2.4 TCP 与 UDP	18
2.5 IP 地址	19
2.6 域名系统	21
2.7 通信端口	23
2.8 RFC 文件	24
2.9 网络结构与类型	27
2.9.1 网络结构	27
2.9.2 网络类型	28
第 3 章 Socket	30
3.1 Berkeley Socket	30
3.2 Microsoft Windows Socket	33
3.3 Java Socket	39
3.4 服务器端与客户端 Socket 应用程序流程	43
3.4.1 服务器端 Socket 应用 程序流程	44
3.4.2 客户端 Socket 应用 程序流程	47
第 4 章 SMTP 通信协议	50
4.1 SMTP 简介	50
4.2 SMTP 客户端——Java 应用程序	56
4.3 SMTP 客户端——Java Applet	68
第 5 章 POP3 通信协议	77
5.1 POP3 介绍	77
5.1.1 用户身份验证	79
5.1.2 邮件事务	80
5.1.3 UPDATE	83
5.2 POP3 客户端	83
第 6 章 服务器端的 FTP 通信协议	100
6.1 FTP 介绍	100
6.1.1 FTP 体系结构	100
6.1.2 FTP 指令	101
6.1.3 FTP 服务器端响应信息	107
6.1.4 FTP 流程	108
6.2 FTP 服务器端程序设计	110
第 7 章 客户端的 FTP 通信协议	129
7.1 FTP API 介绍	130
7.2 FTP 客户端示例 1	135
7.3 FTP 客户端示例 2	152
第 8 章 HTTP 通信协议与 Web 服务器	156
8.1 HTTP 通信协议	156
8.1.1 HTTP 头	157
8.1.2 HTTP 方法	159
8.1.3 HTTP 响应信息	162
8.1.4 URL	163
8.1.5 HTTP 流程	164
8.2 Web 服务器示例	165

第 9 章 Java Web 浏览器	180	12.4.2 取得客户端 Secure Socket 信息	270
9.1 Java Web 浏览器核心	181	12.4.3 客户端发送与接收	271
9.2 HyperlinkListener	189	12.4.4 客户端关闭	271
9.3 动画显示网页下载状态	193	12.5 Secure Socket 示例—— HTTPS 服务器	271
9.4 查看及保存网页源代码	198	Java 网络程序设计——下篇 企业网络技术篇	
9.5 网页浏览功能与发送 E-mail	207	第 13 章 企业解决方案	281
9.6 设置 HTTP 代理服务器	210	13.1 B2C	281
第 10 章 聊天程序的客户端/ 服务器端	212	13.2 B2B 整合	284
10.1 聊天程序的客户端—— Java Applet	213	13.3 供应链管理	286
10.2 聊天程序的客户端—— Java 应用程序	223	13.4 企业应用系统整合	287
10.3 聊天服务器程序设计	231	13.5 门户网站	288
第 11 章 网络游戏	238	第 14 章 Java 档案	290
11.1 游戏体系结构与设置	238	14.1 建立 JAR 文件	291
11.2 游戏服务器	242	14.2 查看 JAR 文件内容	293
11.3 游戏客户端界面	249	14.3 读取和更新 JAR 文件内容	295
第 12 章 JSSE 与 HTTPS 服务器	253	14.4 执行 JAR 文件	295
12.1 SSL	253	14.5 电子签名	296
12.2 JSSE API 介绍	256	第 15 章 Java Plug-in 与 HTML Converter	298
12.2.1 安装 JSSE API	257	15.1 Java Plug-in 体系结构	299
12.2.2 设置 Java 安全性原则	257	15.2 Java Plug-in 规范	300
12.2.3 公钥	258	15.3 Java Plug-in HTML Converter 介绍	302
12.2.4 JSSE API	259	15.4 Java Plug-in HTML Converter 的使用方法	306
12.3 服务器端 Secure Socket 程序流程	262	第 16 章 Java 安全性原则	308
12.3.1 建立服务器端 Secure Socket	263	16.1 定义安全性原则	308
12.3.2 取得服务器端 Secure Socket 信息	264	16.2 policytool 工具	312
12.3.3 接收客户端连接	265	16.3 安全性原则的类	316
12.3.4 服务器端接收与发送	266	第 17 章 三层体系结构 Socket 与 JDBC	324
12.3.5 服务器端关闭链接	268	17.1 服务器端 Socket 程序流程	325
12.4 客户端 Secure Socket 程序流程	268	17.1.1 建立服务器端 Socket	326
12.4.1 建立客户端 Secure Socket	269		

17.1.2 取得服务器端 Socket 信息	327	18.10 JavaMail Applet 与安全性原则	441
17.1.3 接收客户端连接	329	第 19 章 Java Web Start	445
17.1.4 接收与发送——服务器端.....	330	19.1 Java Web Start 介绍.....	445
17.1.5 关闭连接——服务器端.....	332	19.2 Java Web Start 体系结构及系统需求	446
17.2 客户端 Socket 应用程序流程.....	332	19.3 Java Web Start 安装与设置.....	447
17.2.1 建立客户端 Socket.....	333	19.4 设置 Web 服务器	450
17.2.2 取得客户端 Socket 信息.....	334	19.4.1 设置	451
17.2.3 接收与发送——客户端.....	335	19.4.2 设置 Netscape 企业服务器	451
17.2.4 关闭连接——客户端	337	19.4.3 设置 Tomcat 服务器.....	453
17.3 JDBC 简介	337	19.5 Java Web Start 应用程序管理器	453
17.3.1 JDBC 版本介绍.....	337	19.5.1 菜单	454
17.3.2 JDBC 驱动程序.....	340	19.5.2 图标	457
17.3.3 JDBC 流程	341	19.6 Java 网络加载协议	458
17.4 三层体系结构	350	19.6.1 jnlp 元素	460
17.4.1 表示层	351	19.6.2 information 元素.....	460
17.4.2 商业逻辑层	359	19.6.3 Security 元素	462
17.4.3 数据源层	372	19.6.4 resources 元素	463
第 18 章 JavaMail API	375	19.6.5 application-desc 元素	467
18.1 JavaMail API 介绍	375	19.6.6 applet-desc 元素	467
18.1.1 设置 JavaMail 1.1.3.....	376	19.7 Java 应用程序的设计考虑与限制	468
18.1.2 设置 JavaMail 1.2.....	377	19.8 HTML 网页设计	470
18.1.3 设置 JAF.....	377	19.9 范例解说	471
18.2 JavaMail 体系结构.....	379	第 20 章 J2EE 连接器架构	475
18.3 JavaMail 核心成员与流程处理.....	379	20.1 J2EE 连接器架构	477
18.3.1 Session.....	380	20.2 通用客户界面	481
18.3.2 Message	382	20.3 JCA 1.0 的缺点	484
18.3.3 Address	388	附 录	486
18.3.4 Authenticator	390	附录 A HTML Converter.....	486
18.3.5 Transport.....	391	附录 B Java 安全性原则.....	489
18.3.6 Store 与 Folder.....	393	附录 C 通信端口.....	493
18.4 传送邮件	395	附录 D 多用途因特网邮件扩展协议 (MIME).....	498
18.5 传送附件	409	附录 E java.util.Locale——ISO 语言代码	508
18.6 接收邮件	413		
18.7 删除邮件	427		
18.7.1 暂时删除	428		
18.7.2 永久删除	429		
18.8 处理附件	432		
18.9 检测邮件处理状态	438		

附录 F	java.util.Locale – ISO 国家(或地区)代码	512
附录 G	Sun 与 Microsoft 的战争 ——从 Java 谈起	519
附录 H	参考资料	522

前 言

Java 技术具有强大的网络功能和全新的理念。作为一种跨平台的程序设计语言，Java 具备简化的优点，并提供面向对象而又不依赖于机器的开放结构，具有卓越的可移植性、安全性特点，同时具有灵敏的响应和交互能力。

按照 Sun 的定义，Java 技术主要包括 3 个平台：从小型设备的 J2ME、适用于个人计算机以及工作组服务器的 J2SE，到支持企业级应用的 J2EE。Java 的三种平台目前在实际中应用非常广泛，这股趋势似乎会随着 Internet 的兴盛而更加风靡。

由于最初设计出发点就是面向网络，所以 Java 能够完全满足网络应用对便捷性、模块化和安全性的要求。Java 技术可以轻松地连接现有的计算机系统、降低计算成本和加快软件开发。同时，Java 还使企业能够通过 Internet 安全地连接客户、供应商和合作伙伴。这些优势使得越来越多的企业倾向于部署基于 Java 的应用。

本书就 Java 技术中的 TCP/IP 通信协议、企业网络技术、Java 编程思想等 Java 专题与新技术作了深入讨论，详细介绍了 Java 的网络应用以及 Java 在企业网络建设中所起的重要作用。

全书共分上下两篇。

上篇为 TCP/IP 与 Internet 程序设计篇，共 12 章，各章内容要点如下：Java 版本、应用范畴及系统结构(第 1 章)；与网络相关的一些基本概论，如 OSI 七层模型、IP 地址、TCP/IP 协议、通信端口和网络结构等(第 2 章)；网络应用的核心——Socket，及其应用程序流程(第 3 章)；发送电子邮件的 SMTP 协议和接收电子邮件的 POP3 协议(第 4、5 章)；客户端和服务端端的 FTP 通信协议及相关内容(第 6、7 章)；HTTP 通信协议与 Web 服务器(第 8 章)；Java Web 浏览器的有关内容(第 9 章)；聊天程序的客户端与服务端端的程序设计(第 10 章)；网络游戏的程序设计实例讲解(第 11 章)；JSSE 与 HTTPS 服务器(第 12 章)。

下篇为企业网络技术篇，分 8 章内容介绍了企业级软件的 Java 技术解决方案，着重于 Java 专题与新技术的说明。第 13 章概述了企业级软件的构建模式及 Java 应用构架。其余各章详细介绍了 Java 技术实现，包括：Java 档案、Java Plug-in 与 HTML Converter、Java 安全性原则、三层体系结构 Socket 与 JDBC、JavaMail API、Java Web Start、J2EE 连接器架构。

其中值得一提的是 Java Web Start 及 Java 连接器架构，前者将包含于 JDK 1.4 版之中，以实现 Easy Deployment 的概念，后者则是 J2EE 新增的一项标准，以作为应用程序与企业信息系统之间的桥梁。

各种类型的企业级应用软件，从 B2B(企业对企业)、供应链管理、企业应用系统集成，到门户网站等，均有两个共同的特点，即 100% Java 开发及支持 XML。由企业级应用可以看到一个趋势，那就是 Java 并非纸上谈兵或昙花一现的语言、环境或平台，而是真正已实际运用在企业中，并提供企业完善的解决方案。

本书旨在帮助 Java 技术的专业人士全面提升技能，重点为读者打造面向对象的程序设计基础，从而熟练掌握 Java 的应用与开发。书中内容及范例均经笔者再三着墨及测试，为的是能力求完美，并希望读者有所收获，同样地，秉着“分享”的精神，希望读者能不吝指教，以使本书内容更加充实。

Java 网络程序设计——上篇



TCP/IP 与 Internet 程序设计篇

- Java 简介
- 网络概论
- Socket
- SMTP 通信协议
- POP3 通信协议
- 服务器端的 FTP 通信协议
- 客户端的 FTP 通信协议
- HTTP 通信协议与 Web 服务器
- Java Web 浏览器
- 聊天程序的客户端/服务器端
- 网络游戏
- JSSE 与 HTTPS 服务器

本篇就 TCP/IP 通信协议、Java 专题与新技术作了深入讨论，包括各类体系结构、编程方法等，涵盖了 Java 网络程序设计中的各类应用。

第 1 章 Java 简介

1.1 Java 的版本

Java 自推出以来，随着不断的改版以及新的 API(Application Programming Interface, 应用程序接口)和功能的增加，其应用越来越广。早期的 Java 主要用于网页上特殊效果的处理，如 Java Applet，当通过浏览器向 Web 服务器传达 HTTP 请求(Request)时，Web 服务器会随着此请求，将在网页中定义的 Java Applet 下载到客户端并加以执行，由于 Java Applet 早期的执行效率不是很理想，因此 Java 当时并不为人们接受。

随着 Internet 的普及，Java 日益显出其重要性，Sun 公司则不断增加 Java 包(Package)及其附加功能，版本更是在极短的时间内更新，到 Java 2(JDK 1.2+)之后，为满足不同的需求，将 Java 2 分为 3 种版本，分别为：

- J2ME(Java 2 Platform, Micro Edition, Java2 平台袖珍版)：主要用于消费性电子产品，如手机、掌上电脑、个人数字助理、汽车的卫星导航系统及家电产品等。
- J2SE(Java 2 Platform, Standard Edition, Java2 平台标准版)：标准 Java 版本，包括 Java 2 SDK(Software Development Kit, 软件开发工具包)、开发所需的附属工具(如调试工具、Java 文件管理器、Java Applet 档案、Java Applet 浏览器等)、Java 应用程序接口包 JRE(Java Runtime Environment, Java 运行环境)。此版本一般适用于个人计算机以及一般工作组服务器的 Internet 应用程序开发上。
- J2EE(Java 2 Platform, Enterprise Edition, Java2 平台企业版)：除了标准的 J2SE 所包括的软件包之外，J2EE 还包括了EJB(Enterprise JavaBeans, 企业 JavaBeans)的制作及维护工具、将 Tomcat Servlet Container 集成于 Java Servlet 2.2 和 JSP 1.1 的应用、支持 CORBA 技术和支持 XML(eXtensible Markup Language, 可扩展标记语言)标准等。Sun 公司将 J2EE 定义为企业级应用程序。

图 1.1 为 Java 各版本在各种平台系统中应用的定义。
以内容区分，Java 各版本分别包括下列技术和产品。

1. J2ME

- 技术：
 - ◆ Java 2 平台，袖珍版
 - ◆ Connected Device Configuration (CDC, 连接设备配置)
 - ◆ Connected Limited Device Configuration (CLDC, 连接有限设备配置)
 - ◆ C Virtual Machine (CVM, C 虚拟机)
 - ◆ K Virtual Machine (KVM, K 虚拟机)

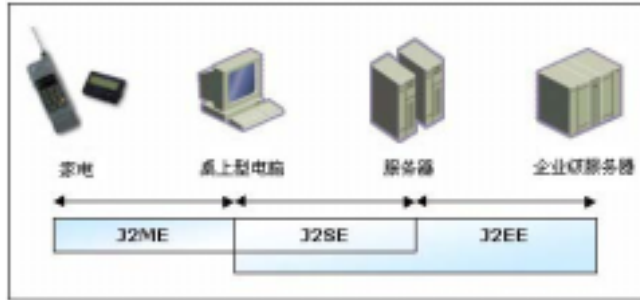


图 1.1 Java 的版本

- ◆ Personal Java Application Environment (个人 Java 应用环境)
- ◆ Personal Java Technology (个人 Java 技术)
- ◆ Embedded Java Application Environment (Java 嵌入应用环境)
- ◆ Embedded Java Technology (Java 嵌入技术)
- ◆ Java Card
- ◆ Java Phone API
- ◆ Java TV API
- ◆ Jini Network Technology (Jini 网络技术)
- ◆ Mobile Information Device Profile (MIDP, 移动信息设备)
- 产品：
 - ◆ Personal Applications Suite (个人应用程序组)
 - ◆ Java Dynamic Management Kit (Java 动态管理工具)
 - ◆ Java Embedded Server Software (Java 嵌入服务器软件)

2. J2SE

- 产品：
 - ◆ Java 2 SDK, Standard Edition (Java 2 软件开发工具包, 标准版)
 - ◆ Java 2 Runtime Environment, Standard Edition (Java 2 运行环境, 标准版)
 - ◆ Java Plug-in (Java 插件)
 - ◆ Java Web Start
 - ◆ JavaBeans Development Kit (JavaBeans 开发工具包)
 - ◆ Java HotSpot Server Virtual Machine (热点服务器虚拟机)
- 主要 API：
 - ◆ Collections Framework (集合架构)
 - ◆ Java Foundation Classes (JFC, Java 基类)
 - ◆ Swing Components
 - ◆ Pluggable Look & Feel
 - ◆ Accessibility
 - ◆ Drag and Drop

- ◆ Security
- ◆ Java Interface Definition Language (Java IDL , Java 接口定义语言)
- ◆ Java Database Connectivity (JDBC , Java 数据库连接)
- ◆ JavaBeans
- ◆ Remote Method Invocation (RMI , 远程方法激活协议)
- ◆ Java 2D

3. J2EE

- 技术：
 - ◆ EJB
 - ◆ JavaServer Pages (JSP , Java 服务器网页)
 - ◆ Java Servlet
 - ◆ Java Naming and Directory Interface (JNDI , Java 命名及目录接口)
 - ◆ Java IDL
 - ◆ JDBC
 - ◆ Java Message Service (JMS , Java 消息服务)
 - ◆ Java Transaction Architecture (JTA , Java 事务处理)
 - ◆ Java Transaction Service (JTS , Java 事务服务)
 - ◆ JavaMail
 - ◆ RMI-IIOP
- 产品：Java 2 SDK , Enterprise Edition (Java 2 软件开发工具包 , 企业版)

根据 Java 所提供的 API 来进行分类，大致有以下重要功能(除标准的 Java API 之外)：

- Accessibility
- Drag and Drop
- Enterprise JavaBeans
- JDBC
- Java 2D
- Java 3D
- Java Communications (COMM)
- Java Foundation Classes
- Java Internationalization and Localization
- Java Media Framework
- Java Message Service (JMS)
- Java Message Queue
- Java Naming and Directory Interface (JNDI)
- Java Phone API
- Java Speech API
- Java TV API
- Java Telephony API

- Java Transaction (JTA)
- JavaBeans Activation Framework (JAF)
- JavaHelp
- JavaMail
- Jini Network Technology
- Pluggable Look & Feel
- RMI-IIOP
- Remote Method Invocation (RMI)
- Java Swing
- Java XML API

1.2 Java 的应用范畴

根据 Gartner Group 的预测,Java 软件开发工程师在今后 4 年中,将增长 5%~20%;至 2003 年,Java 将提供优越的平台以处理服务器端 XML 文件,并且将 Java 与 XML 集成应用到电子商务(e-Business)应用上;至 2004 年,将有 60% 以上的企业使用 Java 开发企业应用程序。

Java 从应用范畴(Scope)的角度可分为下列 5 类:

- 两层的客户/服务器体系结构(2-Tier Client-and-Server Architecture)
- 基于 HTML 的两层体系结构(2-Tier HTML-based Architecture)
- 基于 HTML 的三层体系结构(3-Tier HTML-based Architecture)
- 基于 Applet 的三层体系结构(3-Tier Applet-based Architecture)
- Java Web Start 技术

1.2.1 两层的客户/服务器体系结构

两层的客户/服务器体系结构如图 1.2 所示。

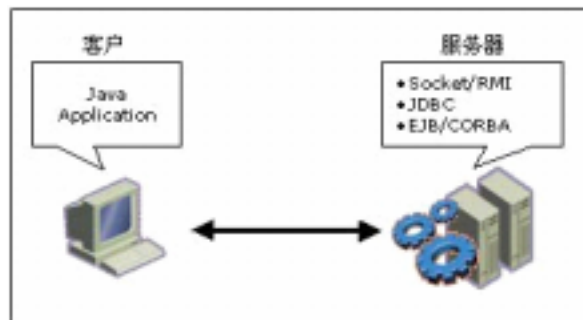


图 1.2 两层的客户/服务器体系结构

如同以往的客户/服务器应用，在客户端(前台)提供了用户界面(User Interface)，以使用户通过图形化界面的操作，访问服务器端(后台)的信息。此图形化用户界面渐渐用 javax.swing(Swing API)取代 java.awt(AWT API)来开发。服务器端主要提供数据源(Data Source)。

说明：笔者习惯使用“信息”一词取代“数据”一词，因为在访问时，不一定是针对数据库的数据，也有可能来自不同类型的应用系统，如 e-mail。

在 Java 中，客户端与服务器端的沟通主要通过 Socket(java.net)和 RMI(java.rmi)；而服务器端与数据源的沟通，主要通过 JDBC(java.sql)和 CORBA(org.omg.CORBA, Common Object Request Broker Architecture, 公共对象请求代理体系结构)。

1.2.2 基于 HTML 的两层体系结构

基于 HTML 的两层体系结构的应用是比较常见和较易开发的，如图 1.3 所示。

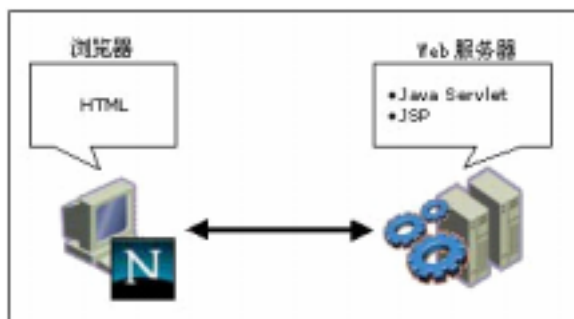


图 1.3 基于 HTML 的两层体系结构

客户端仅使用 Web 浏览器与后端服务器沟通，客户端通过浏览器将 HTTP(HyperText Transfer Protocol, 超文本传输协议)请求下达到服务器端，而服务器端经过处理之后，会将 HTML 响应返回到客户端浏览器以显示结果。

说明：HTTP 通信协议定义的方法包括 CONNECT、DELETE、GET、HEAD、OPETOINS、POST、PUT 和 TRACE。在 HTML 中较常使用的是 GET 和 POST 方法，如：

```
GET /login.html HTTP/1.1
Accept: */*
Accept-Language: zh
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows NT; DigExt)
Host: localhost
Connection: Keep-Alive
```

这些是 HTML 的文件头(Header)部分。

网页的处理主要以 HTML 为主，此类应用的重点在服务器端。由于客户端通常会下达

查询和更新数据等请求到服务器端，服务器端必须能处理与后端数据源的沟通，目前较常见的方法(不论是否为 Java)有：

- Active Server Page(ASP)及 ASP.NET
- JSP 及 Java Servlet
- PHP
- Perl

同样的，在 Java 中，服务器端与数据源间的沟通方法主要有：JDBC(java.sql)和 CORBA(org.omg.CORBA)。

1.2.3 基于 HTML 的三层体系结构

基于 HTML 的三层体系结构如图 1.4 所示。

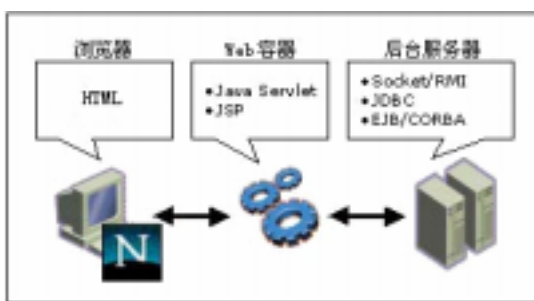


图 1.4 基于 HTML 的三层体系结构

基本上，基于 HTML 的三层体系结构与基于 HTML 的两层体系结构类似，只是前者更进一步将商业逻辑层(Business Logic Tier)与数据源层(Data Source Tier)分开，各司其职。

其中商业逻辑层专门定义如何访问信息的逻辑，即如何定义 SQL 指令访问信息，而此层所使用的方法如下：

- ASP 及 ASP.NET
- JSP 及 Java Servlet
- PHP
- Perl

数据源层则负责信息的保存，如数据库、ERP 系统和邮件服务器等。商业逻辑层与数据源层之间的沟通方式包括：Socket(java.net)、RMI(java.rmi)、JDBC(java.sql)和 CORBA(org.omg.CORBA)。

1.2.4 基于 Applet 的三层体系结构

基于 Applet 的三层体系结构如图 1.5 所示。