

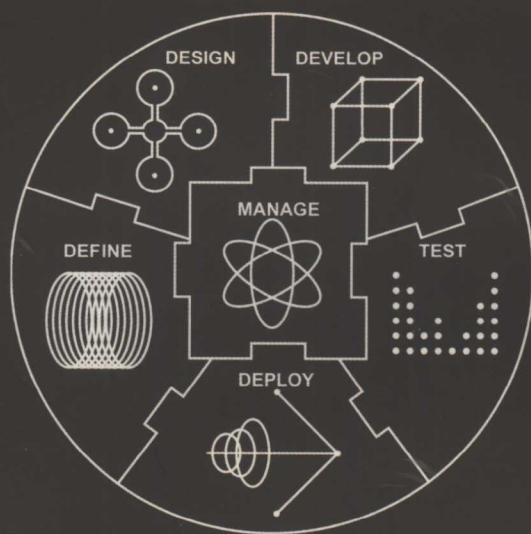
高等职业技术教育IT类双证教材

Borland 软件认证课程推荐教材

# Delphi 7

## 快速网络开发教程

ATA教育公司 总策划  
牛汉民 编 著



Delphi 快速网络开发工程师认证



科学出版社

[www.sciencep.com](http://www.sciencep.com)

高等职业技术教育 IT 类双证教材  
Borland 软件认证课程推荐教材

# Delphi 7 快速网络开发教程

(Delphi 快速网络开发工程师认证)

ATA 教育公司 总策划

牛汉民 编著

科学出版社

北京

## 内 容 简 介

本书是高等职业技术教育中 IT 类专业学生获取学历证书和国际著名软件厂商 Borland 软件认证证书的学习用教材。本教材依据课程教学大纲并结合 Delphi 7 网络开发工程师的教学特点编写而成。

全书共分 8 章, 主要内容包括创建 Web 服务器应用程序、Web Broker、WebSnap、IntraWeb、XML 与 Delphi、使用 Web Services、InternetExpress、Sockets 编程。为便于教学使用, 本教材编入了大量教学范例, 并拟制了大量习题。本教材结构清晰, 应用实例丰富, 实现了理论学习和具体应用的充分结合。

本书可作为各大专院校、高等职业技术学院计算机软件开发专业课程和相关双证教学课程的教材, 也可供从事 Delphi 7 网络开发的程序设计人员学习、参考。

### 图书在版编目 (CIP) 数据

Delphi 7 快速网络开发教程/牛汉民编著, -北京: 科学出版社, 2005  
(高等职业技术教育 IT 类双证教材. Borland 软件认证)  
ISBN 7-03-016513-6

I. D… II. ①牛… III. 软件工具-程序设计-高等学校: 技术学校-教材  
IV. TP393

中国版本图书馆 CIP 数据核字 (2005) 第 139331 号

责任编辑: 杨 凯 刘晓融 / 责任制作: 魏 谨

责任印制: 刘士平 / 封面设计: 李 力

北京东方科龙图文有限公司制作

<http://www.okbook.com.cn>

科学出版社 出版

北京东黄城根北街16号

邮政编码: 100717

<http://www.sciencep.com>

源海印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

\*

2006年1月第一版 开本: B5 (720×1000)

2006年1月第一次印刷 印张: 22 1/4

印数: 1~3 000 字数: 428 000

定价: 45.00 元

(如有印装质量问题, 我社负责调换〈新欣〉)

高等职业技术教育IT类双证教材  
Borland 软件认证课程推荐教材

## 编审委员会成员名单

编 委 马肖风 王建国 罗晓中

组织实施 王 健 刘兴国

技术编审 尤克滨 李晓辉 王 健

# 前 言

Borland 公司自 1983 年成立以来,秉承“持久领先、卓越不凡”的服务宗旨,一直致力于加速整个应用程序的开发周期,不断推出其编译速度快、代码质量高而且易于调试的应用程序开发工具产品,逐步奠定了自己在世界 IT 行业应用程序开发工具领域的领先地位。当前广泛流行的 Borland 公司的软件产品,为我们提供了涵盖定义(Define)、设计(Design)、开发(Develop)、测试(Test)和发布(Deploy)等各个环节在内的整个软件应用周期的解决方案。Delphi 7 就是 Borland 公司强力推出的重要的应用开发工具之一。

Delphi 7 为我们提供了一个高效率的应用开发环境,其应用范围十分广泛,不仅能够用来编写一般的控制台和 GUI(Graphic User Interface, 图形用户界面)应用程序,也可以用来开发多媒体应用程序、数据库应用程序和 Web 服务器应用程序。使用 Delphi 7 所提供的各种最新技术,我们还可以非常方便地实现支持所有工业标准的企业级多层分布式应用架构。

Delphi 7 是专门为新一代电子商务领域的应用开发所提供的从应用设计到应用发布的完整解决方案,并引导开发人员开始迈向 .NET 这一全新的平台。基于 Delphi 6 的强大功能,Delphi 7 进一步加强和完善了高性能的数据访问中间件技术 DataSnap、基于组件的 Web 应用程序架构 WebSnap 以及 Web Services 开发平台 BizSnap,进一步改进了集成开发环境、编译器和调试工具,并提供了诸如 IntraWeb 组件、Rave 报表组件等在内的新的特性。为了加强对应用开发周期更加全面的支持,在 Delphi 7 中,还包含了建模工具——ModelMaker 和 Bold for Delphi。

Delphi 7 突破了基于 Windows 平台这一局限,它不仅仍然包含一直为广大开发人员所喜爱的 Windows 环境下的编译器,同时也包含了 Kylix 3 的 Delphi 语言版本以及一个全新的 .NET 编译器预览版本。可以看出, Borland 公司正在其开发工具产品中逐步实现完全平台无关性这一承诺。因此,对于广大开发人员来说, Delphi 仍将继续是我们今后从事应用开发的正确选择。

本书是用于 Delphi 7 快速网络开发工程师培训的基本教材。全书共分为 8 章。第 1 章“创建 Web 服务器应用程序”讨论了 Web 服务器应用程序的一些基本概念,重点讨论了 HTTP 服务器的操作、Web 服务器 IIS 的配置步骤和各种类型 Web 服务器应用程序的特点、创建、使用和调试;第 2 章“Web Broker”在介绍 Web Broker 应用程序的结构和其中各个部分之间的工作关系的基础上,重点讨论

Web 调度器的使用, 动作项的使用, 如何访问客户端请求信息, 在 Web 服务器应用程序中如何创建 HTTP 响应消息; 第 3 章“WebSnap”讨论了 WebSnap 体系结构的内容, 使用 WebSnap 创建 Web 服务器应用程序的步骤, 适配器和适配器页面生成器的使用; 第 4 章“IntraWeb”讨论了 IntraWeb 的开发模式的特点和创建步骤, IntraWeb 页面模式与 Web Broker、WebSnap 相结合的开发实现; 第 5 章“XML 与 Delphi”介绍了 XML 的语法和相关概念, 在 Delphi 7 中管理 XML 文档的方法, 在数据库应用中对 XML 文档的使用方式; 第 6 章“使用 Web Services”讨论了可调用接口的使用, 如何编写支持 Web Services 的服务器, 为 Web Services 编写客户端应用程序的步骤、方法; 第 7 章“InternetExpress”介绍了 InternetExpress 组件的作用和使用, 重点讨论了创建 InternetExpress 应用程序的步骤; 第 8 章“Sockets 编程”讨论了 Sockets 基本的概念、Socket 连接的三种基本类型、关于 Sockets 的一些描述信息、Socket 组件的使用、Socket 事件的响应、在 Socket 连接上的读、写。

在本书的编写过程中, 我们得到了 Borland 公司的授权和大力支持, 尤克滨、李晓辉、王健等技术专家就本书的技术审核提供了热情帮助, 在此一并表示衷心感谢。

ATA 公司为本书配套提供有教学 PPT 软件、多媒体演示光盘、教学大纲、考试大纲、模拟试题, 以及认证考试和证书授予, 供合作院校教师和学生使用、参与。普通读者可以通过 ATA 教育公司网站 (<http://www.atlearning.com/>) 下载获得教学大纲、考试大纲等资源, 还可与 ATA 公司电话联系相关事宜: 010-65181122。

藉此特别说明, 本书是由科学出版社出版的高等职业技术双证教育 IT 类专业 Borland 软件认证课程系列教材之一, 用于 Borland《Delphi 快速网络开发工程师认证》证书考试的课程教学。本系列教材还包括针对 Borland 软件认证《Java 初级程序员认证》、《Java Web 应用开发工程师认证》、《J2ME 应用开发工程师认证》、《Delphi 初级程序员认证》、《Delphi 程序员认证》、《Delphi 高级程序员认证》等证书考试的课程教材。读者可以通过 <http://www.sciencep.com> 了解其他教材的出版情况, 并通过 <http://www.atlearning.com> 网站获得相关考试的信息。


北京全美教育技术服务有限公司


(ATA 教育公司)

[www.atlearning.com](http://www.atlearning.com)


2005 年 10 月



# 目 录


<b>第 1 章 创建 Web 服务器应用程序</b> .....	<b>1</b>
1.1 Web 服务器应用程序概述.....	1
1.1.1 Web 服务器应用程序的工作方式 .....	2
1.1.2 Web 服务器应用程序所遵循的 Internet 标准.....	2
1.1.3 统一资源定位符及其组成.....	2
1.1.4 HTTP 请求信息的内容.....	4
1.2 HTTP 服务器的操作.....	6
1.2.1 服务客户端请求.....	6
1.2.2 响应客户端请求.....	6
1.2.3 一个简单的 Web 服务器范例 .....	7
1.3 Web 服务器应用程序的类型.....	8
1.3.1 ISAPI / NSAPI.....	9
1.3.2 CGI.....	9
1.3.3 Apache .....	9
1.3.4 调试类型.....	9
1.3.5 Web 服务器应用程序目标类型的转换 .....	10
1.4 Web 服务器 IIS 的配置.....	11
1.5 调试 Web 服务器应用程序.....	15
1.5.1 使用 Web 应用程序调试器 .....	15
1.5.2 调试 DLL 类型的 Web 应用程序.....	17
1.6 Web Broker 和 WebSnap .....	19
1.7 HTML 页面内容生成器.....	20
1.7.1 使用页面生成器.....	21
1.7.2 使用数据集页面生成器.....	25
1.7.3 使用表格生成器.....	29
 练习 题 .....	37
<b>第 2 章 Web Broker</b> .....	<b>40</b>
2.1 使用 Web Broker 创建 Web 服务器应用程序 .....	40
2.1.1 工程文件和 Web 模块单元文件 .....	41



2.1.2	Web 模块 .....	44
2.1.3	Web 应用类对象 .....	45
2.2	Web Broker 应用程序的结构 .....	48
2.3	Web 调度器 .....	49
2.3.1	在调度器中添加动作项 .....	50
2.3.2	调度请求消息 .....	50
2.4	动作项 .....	51
2.4.1	动作项的控制 .....	52
2.4.2	动作项的匹配 .....	52
2.4.3	使用动作项响应请求消息 .....	53
2.4.4	使用多个动作项的 Web 服务器应用程序范例 .....	54
2.5	访问客户端请求信息 .....	57
2.5.1	包含请求头信息的属性 .....	58
2.5.2	HTTP 请求消息的内容 .....	59
2.6	创建 HTTP 响应消息 .....	59
2.6.1	填写响应头 .....	60
2.6.2	设定响应内容 .....	61
2.6.3	发送响应 .....	62
2.7	在响应中使用数据库信息 .....	67
2.7.1	在 Web 模块中添加会话 .....	67
2.7.2	使用数据库信息的应用范例 .....	68
	练习题 .....	79
<b>第 3 章</b>	<b>WebSnap .....</b>	<b>81</b>
3.1	WebSnap 概述 .....	81
3.1.1	WebSnap 体系结构 .....	81
3.1.2	WebSnap 中的 Web 模块 .....	83
3.2	使用 WebSnap 创建 Web 服务器应用程序 .....	87
3.2.1	选择所创建的 Web 服务器应用程序的类型 .....	87
3.2.2	定制 Web 应用模块及其所包含的组件 .....	88
3.2.3	设置应用模块选项 .....	89
3.2.4	在工程中添加数据模块和页面模块 .....	92
3.3	适配器和适配器页面生成器 .....	95
3.3.1	适配器 .....	95
3.3.2	适配器页面生成器 .....	97
3.3.3	在工程中添加数据集显示 .....	103



3.3.4 以主、从表的形式显示数据.....	105
3.4 适配器页面生成器高级应用技术.....	107
3.4.1 数据编辑表单.....	107
3.4.2 使用定制的动作.....	116
3.5 数据表达和页面格式化.....	120
3.5.1 应用程序数据的表达.....	120
3.5.2 适配器网格元素的格式化处理.....	122
3.5.3 OnGetDisplayText 事件.....	123
3.5.4 Custom 属性.....	124
3.5.5 CSS 格式化.....	126
3.6 服务器端脚本.....	130
3.6.1 显示动态数据.....	130
3.6.2 显示图像和备注数据.....	132
3.6.3 引用适配器动作.....	134
3.7 会话和用户列表服务.....	138
3.7.1 支持会话和用户列表服务的相关组件.....	138
3.7.2 使用会话服务统计点击次数的范例.....	140
3.7.3 使用用户列表服务.....	142
 练习题.....	149
<b>第 4 章 IntraWeb.....</b>	<b>151</b>
4.1 IntraWeb 概述.....	151
4.1.1 IntraWeb 的主要特性.....	152
4.1.2 IntraWeb 组件.....	152
4.1.3 IntraWeb 的开发模式.....	155
4.1.4 一个简单的 IntraWeb 程序.....	157
4.1.5 IntraWeb 服务器控制器.....	160
4.2 IntraWeb 应用模式开发.....	160
4.2.1 窗口管理.....	160
4.2.2 布局管理.....	161
4.2.3 状态管理.....	163
4.2.4 会话管理.....	165
4.2.5 应用模式开发范例.....	166
4.3 IntraWeb 页面模式开发.....	169
4.3.1 IntraWeb 页面模式下的管理.....	169
4.3.2 Web Broker + IntraWeb 页面模式的开发范例.....	170
4.3.3 WebSnap + IntraWeb 页面模式的开发范例.....	174

4.4 IntraWeb 数据库应用开发举例 .....	183
 练习题 .....	191
<b>第 5 章 XML 与 Delphi .....</b>	<b>193</b>
5.1 XML 概述 .....	194
5.1.1 XML 文档 .....	194
5.1.2 XML 文档和 HTML 文档的比较 .....	196
5.1.3 XML 文档和 HTML 文档 .....	196
5.1.4 良构的 XML 文档 .....	198
5.1.5 文档类型定义 .....	200
5.1.6 有效的 XML 文档 .....	204
5.1.7 可扩展样式表语言 .....	204
5.2 在 Delphi 7 中管理 XML 文档 .....	215
5.2.1 使用 TXMLDocument 组件编程 .....	216
5.2.2 利用 XML 数据绑定接口处理 XML 文档 .....	226
5.3 在数据库应用中使用 XML .....	233
5.3.1 XML 文档与数据包之间转换的定义 .....	234
5.3.2 将 XML 文档转换为数据包 .....	240
5.3.3 将 XML 文档用作提供器的源文档 .....	244
5.3.4 将 XML 文档用作提供器的客户 .....	248
5.4 XSL 页面生成器 .....	250
 练习题 .....	257
<b>第 6 章 使用 Web Services .....</b>	<b>260</b>
6.1 可调用接口 .....	261
6.1.1 在可调用接口中使用非数量类型 .....	263
6.1.2 注册非数量类型 .....	263
6.1.3 使用远程化对象 .....	265
6.1.4 表示附件 .....	266
6.1.5 管理远程化对象的生命期 .....	266
6.1.6 远程化对象举例 .....	266
6.2 Delphi 中的 Web Service 组件 .....	268
6.2.1 THTTTPRIO 组件 .....	268
6.2.2 TOPToSoapDomConvert 组件 .....	269
6.2.3 THTTTPReqResp 组件 .....	269
6.2.4 THTTTPSoapDispatcher 组件 .....	270

6.2.5	THTTSPsoapPascalInvoker 组件 .....	270
6.2.6	TWSDLHTMLPublish 组件 .....	270
6.2.7	TSoapConnection 组件 .....	271
6.3	编写支持 Web Services 的服务器 .....	271
6.3.1	创建 Web Services 服务器 .....	271
6.3.2	使用 SOAP 应用程序向导 .....	272
6.3.3	添加新的 Web Services .....	273
6.3.4	使用 WSDL 导入器 .....	274
6.3.5	为 Web Services 应用程序生成 WSDL 文档 .....	275
6.4	为 Web Services 编写客户端应用程序 .....	275
6.4.1	导入 WSDL 文档 .....	275
6.4.2	调用可调用接口 .....	276
6.5	Web Service 使用举例 .....	278
6.5.1	一个简单的 Web Services 及其使用 .....	279
6.5.2	使用一个处理数据集的 Web Service .....	280
6.5.3	在 Web Service 中处理复杂数据类型 .....	283
	练习题 .....	286
<b>第 7 章</b>	<b>InternetExpress .....</b>	<b>289</b>
7.1	InternetExpress 组件 .....	289
7.2	创建 InternetExpress 应用程序 .....	290
7.2.1	使用 Java 脚本库 .....	291
7.2.2	访问应用程序服务器的授权许可 .....	292
7.3	使用 TXMLBroker 组件 .....	292
7.3.1	获取 XML 数据包 .....	292
7.3.2	使用 XML Delta 包对数据进行更新 .....	293
7.4	使用 TInetXPageProducer 组件创建 Web 页面 .....	294
7.4.1	使用 Web 页面编辑器 .....	295
7.4.2	设置 Web 项的属性 .....	296
7.4.3	定制页面生成器模板 .....	296
7.5	InternetExpress 技术应用范例 .....	298
7.5.1	准备工作 .....	298
7.5.2	单数据集维护范例 .....	299
7.5.3	主从数据集维护范例 .....	301
7.5.4	数据集查询范例 .....	304
7.5.5	利用表单提交的范例 .....	308

 练习题 .....	312
<b>第 8 章 Sockets 编程 .....</b>	<b>314</b>
8.1 Sockets 基本的概念 .....	314
8.1.1 基于 Sockets 的通信模型 .....	315
8.1.2 实现服务 .....	315
8.1.3 Socket 通信方式 .....	316
8.1.4 基于 Sockets 通信的基本应用模式 .....	316
8.2 Socket 连接的类型 .....	316
8.2.1 客户端连接 .....	317
8.2.2 监听连接 .....	317
8.2.3 服务器连接 .....	318
8.3 Sockets 的描述 .....	318
8.3.1 主机的描述 .....	318
8.3.2 使用端口 .....	319
8.4 使用 Socket 组件 .....	319
8.4.1 获取连接信息 .....	320
8.4.2 使用客户端 Sockets .....	320
8.4.3 使用服务器 Sockets .....	321
8.5 响应 Socket 事件 .....	326
8.5.1 错误事件 .....	326
8.5.2 客户端事件 .....	327
8.5.3 服务器事件 .....	327
8.6 在 Socket 连接上的读、写 .....	327
8.6.1 非阻塞连接 .....	327
8.6.2 阻塞连接 .....	328
8.7 一个使用定制协议的网络聊天程序 .....	328
8.7.1 开发服务器 .....	329
8.7.2 开发客户端 .....	334
 练习题 .....	339
<b>Borland 认证课程介绍 .....</b>	<b>341</b>

# 第 1 章

## 创建 Web 服务器应用程序

---

### 本章教学目标

- ➔ 了解 Web 服务器应用程序的有关概念、五种标准类型的 Web 服务器应用程序的特点和使用
- ➔ 掌握 HTTP 服务器的操作，Web 服务器 IIS 的配置步骤，调试 Web 服务器应用程序的步骤，各种 HTML 页面内容生成器的使用

推荐课时安排 6 学时

Web 服务器应用程序可以扩展现有 Web 服务器的功能。Web 服务器应用程序从 Web 服务器接收 HTTP 请求消息，继而执行这些消息所请求的动作，最终形成响应返回给 Web 服务器。许多可以由一般的应用程序所执行的操作都可以由 Web 服务器应用程序来完成。

Delphi 7 为我们提供了两种不同的体系结构来开发 Web 服务器应用程序，它们分别是 Web Broker 和 WebSnap。虽然这两种体系结构有所不同，但它们仍然包含了许多共同的元素，一般来说，可以将 WebSnap 视为 Web Broker 的超集。除 Web Broker 和 WebSnap 之外，在 Delphi 7 中，还包含有由 AtoZed 软件公司所开发的 IntraWeb。IntraWeb 是以真正的 RAD 方式来开发 Web 应用程序的组件体系结构。

在本章中，我们讨论 Web Broker 和 WebSnap 的一般特性，并给出基于 Internet 的 Web 服务器应用程序的一般说明。

### 1.1 Web 服务器应用程序概述

在许多常见的应用程序中，需要为用户提供访问数据的界面。通常，我们通过创建传统的前端元素（比如对话框和滚动窗口）来完成这样的任务，并且使用我们所熟悉的窗体设计工具来精确指定这些对象的布局。但是，Web 服务器应用程序则必须按不同的办法来进行设计。所有传递给用户的信息必须采用通过 HTTP 协议进行传送的 HTML 页面的格式，这些页面通常由客户机器上的 Web 浏览器进行解释并以适合于用户特定系统的方式显示出来。

### 1.1.1 Web 服务器应用程序的工作方式

Web 服务器应用程序是对 Web 服务器功能的扩展，它是运行在 Web 服务器机器上的一个独立应用程序（.EXE）或动态连接库（.DLL），Web 服务器应用程序从 Web 服务器接收 HTTP 请求消息，根据请求执行各种操作并生成响应信息返回给 Web 服务器。如图 1-1 所示。

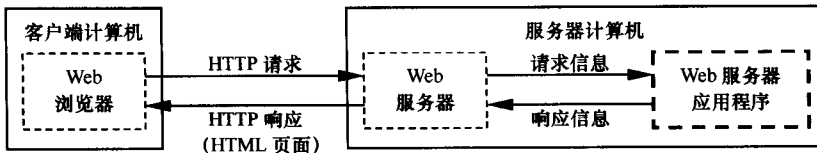


图 1-1 Web 服务器应用程序的工作方式

其中，客户端常用的 Web 浏览器包括微软的 IE（Internet Explorer）和网景的 Netscape 等。Web 服务器是服务于 HTTP 请求的专门软件，它们和 Web 服务器应用程序共同存在于服务器计算机上，常用的 Web 服务器包括 IIS（Internet Information Services）和 Apache 等。

### 1.1.2 Web 服务器应用程序所遵循的 Internet 标准

控制 Internet 活动的许多协议，都是由 Internet 工程任务组（IETF: Internet Engineering Task Force）创建、修改和维护的 Internet 标准——RFC（Request for Comment）文件所定义的。在我们编写 Internet 应用程序的时候，应当遵循下列重要的 RFC 文件所描述的 Internet 标准。

- RFC822: “Standard for the Format of ARPA Internet Text Messages”，描述了消息头的结构和内容。

- RFC1521: “MIME（Multipurpose Internet Mail Extensions）Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies”，描述了用于封装和传递多类型、多格式消息的方法。

- RFC1945: “Hypertext Transfer Protocol—HTTP/1.0”，描述了用于发布超媒体文档的传输机制。

 注 要了解由 IETF 所维护的 RFC 文件，可以登录 [www.ietf.cnri.reston.va.us](http://www.ietf.cnri.reston.va.us) 站点。


### 1.1.3 统一资源定位符及其组成

统一资源定位符（URL: Uniform Resource Locator）是对网络资源位置的一

个完整描述，它由主机、脚本名称、路径信息和查询等多个部分组成并可以在应用程序中进行访问。URL 的组成如图 1-2 所示。



图 1-2 URL 的组成

 **注** “http://”部分用于标识访问协议为超文本传输协议，从技术上来说，它不属于 URL 的组成部分。该部分也可以用来指定诸如 https、ftp 等其他协议。

- “主机”部分用于标识运行 Web 服务器和 Web 服务器应用程序的计算机，通常由主机域名或主机的 IP 地址表示。在“主机”部分还可以指定主机上接收消息的端口号，端口号由冒号引出，如：

www.SomeSite.com: 80

或

172.19.102.153: 80

由于标准协议所使用的端口号是固定的，因此通常可以省略。一些标准的高级协议所使用的端口号如表 1-1 所示。

表 1-1 标准高级协议及其端口号

协 议	端 口
HTTP（超文本传输协议，Hypertext Transfer Protocol）	80
FTP（文件传输协议，File Transfer Protocol）	21
SMTP（简单邮件传输协议，Simple Mail Transfer Protocol）	25
POP3（邮局协议，版本 3，Post Office Protocol）	110
Telnet（远程登录）	23

- “脚本名称”部分用于指定 Web 服务器应用程序。对于微软的 Web 服务器 IIS 来说，脚本名称通常包含 Web 服务器虚拟目录和 Web 服务器应用程序名称。

- “路径信息”部分标识消息在 Web 服务器应用程序中进行处理的目的地，路径信息的值可能引用的是主机上的目录、响应指定消息的组件名称或者 Web 服务器应用程序用来划分对输入消息进行处理的其他设施。

- “查询”部分由一个或多个命名的值所组成，用于对请求信息的细节进行进一步说明。多个“名-值”对之间由“&”符号隔开，“名-值”对的名称和值是在 Web 服务器应用程序中定义的。

 **注** URL 是定义于 HTTP 标准 RFC1945 中的 URI（Uniform Resource Identifier，统一资源标识符）的一个子集。Web 服务器应用程序通常可以从多个

来源生成响应内容，最终结果也可以不驻留在一个特定的位置，而是在需要的时候创建。URI 可以描述那些并非指定位置的资源。

### 1.1.4 HTTP 请求信息的内容


HTTP 请求消息包含了许多描述有关客户、请求目标、请求被处理的方式以及随请求发送的内容等标题 (Header) 信息。每一个标题信息由一个诸如 “Host” 这样的名称进行标识，并后随一个字符串值。

HTTP 请求消息是由 Web 浏览器发出的。当客户端用户在 Web 浏览器中指定一个 URL 之后，浏览器会收集有关协议、特定域、信息的路径、日期和时间、操作环境、浏览器本身以及其他内容信息，最后组织成一个由若干标题组成的请求消息，在连接到指定的 Web 服务器之后发送给服务器。

下面，我们通过一个实验来观察由浏览器发出的 HTTP 请求消息。

**例 1-1** 观察由浏览器发出的 HTTP 请求消息。

- 编译并运行 Delphi 7 的演示程序 NetChat，该程序位于 Delphi 7 安装路径中的 “\Delphi7\Demos\Internet\NetChat” 文件夹。

 **注** NetChat 程序是一个网络聊天程序，可以用来模拟一个接收 HTTP 消息的服务器，其工作原理是使用我们在本书后续内容中将要介绍的 Socket 技术。在为 Socket 服务器组件指定监听端口之后，它就可以在指定的端口上接收 HTTP 消息。

- 将标示为 “Local Port” 的编辑框中的本地端口号修改为 “6000”，然后，选择 “Change” 按钮使设置生效。如图 1-3 所示。

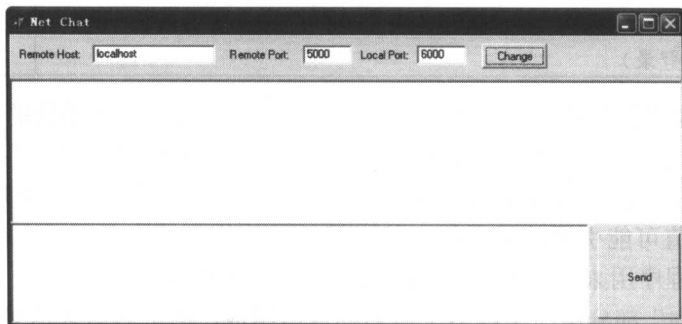


图 1-3 设置 NetChat 程序中 Socket 服务器组件的监听端口

- 打开浏览器，在 “地址” 栏中键入下列 URL：

<http://127.0.0.1:6000/art/gallery.dll/mammals?animal=dog&color=black>

如图 1-4 所示。





图 1-4 在 IE 浏览器中键入 URL

• 在浏览器的“地址”栏中按下回车键后，我们即可在 NetChat 的窗口中观察到浏览器所发出的 HTTP 请求消息。如图 1-5 所示。

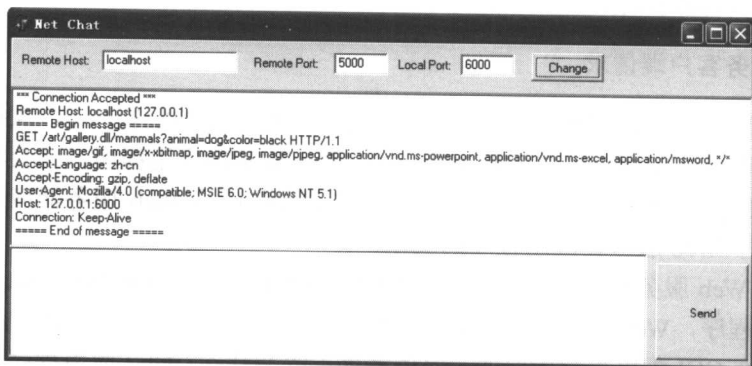


图 1-5 NetChat 程序收到的 HTTP 请求消息

在给出上述 URL 之后，浏览器所发出的 HTTP 请求消息为：

```
GET /art/gallery.dll/mammals?animal=dog&color=black HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/vnd.ms-powerpoint, application/vnd.ms-excel,
application/msword, */*
Accept-Language: zh-cn
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Host: 127.0.0.1:6000
Connection: Keep-Alive
```

• 第一行说明请求是一个 GET 动作，即要求 Web 服务器返回与 GET 后面的 URL 关联的内容。在该行的最后，指明客户端使用的是 HTTP 1.1 标准。

- Accept 标题列出客户端可以当作有效响应接收的媒体类型。
- Accept-Language 标题给出客户端使用的语言信息。
- Accept-Encoding 标题给出客户端使用的编码信息。