



Delphi7

Web 开发与应用

董立达 权重民 问鸿滨 编著



Delphi 7 Web 开发与应用

董立达 权重民 问鸿滨 编著



机械工业出版社

本书全面系统地介绍了在 Delphi 7 环境下开发 Web 应用程序的过程。上篇以真实案例讲解了基于 Delphi+Oracle+Apache 环境开发 Web 应用程序的具体方法及技巧。介绍了 Apache 及 Oracle 数据库的安装和简单应用,以及 WebBroker 技术的实现细节和注意事项。为具有 Delphi 开发基础并对 Web 开发感兴趣的程序员提供一个强有力的工具。下篇详细讨论了在 Delphi7 环境下 IntraWeb 控件包的功能。对 IntraWeb 的技术背景、基础应用、开发流程、基本控件、著名的第三方控件包 TMS IntraWeb Component Pack Pro、IntraWeb 中 CSS、JavaScript 等高级技术的开发与使用等,均做了深入的研究与探讨。读者通过学习,可以充分利用 IntraWeb 的强大功能,开发出高效率的 Web 应用系统。

本书适合 Delphi/Kylix/C++ Builder 程序员、网站开发与设计人员、高校学生及对 Web 开发有兴趣的读者。

图书在版编目(CIP)数据

Delphi 7 Web 开发与应用/董立达等编著. —北京:机械工业出版社, 2003.6
ISBN 7-111-12067-1

I. D... II. 董... III. 软件工具—程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字(2003)第 033477 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 刘 青

责任印制: 路 琳

北京机工印刷厂印刷·新华书店北京发行所发行

2003 年 6 月第 1 版·第 1 次印刷

787mm×1092mm 1/16·22.5 印张·555 千字

0 001—5 000 册

定价: 34.00 元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010) 68993821、88379646

封面无防伪标均为盗版

前 言

随着 Internet 的迅速发展, Web 开发与应用成了程序设计和开发人员学习的重要内容之一。作为开发工具类软件的领导者, Borland 公司在这方面自然是不甘落伍的。Delphi7 包括以下三种重要的 Web 开发技术:

(1) WebBroker。WebBroker 是 Delphi 里面最悠久的历史, 从 3.0 开始, Delphi 就开始支持 Web 开发, 利用的就是 WebBroker 技术。WebBroker 也是其他版本的 Delphi 开发 Web 技术的基础, 它提供了开发 Web 的所有底层功能, 同时其效率、灵活性、稳定性也是最高的。

(2) WebSnap。Borland 从 Delphi 6.0 开始为了增强 Delphi 开发 Web 的能力, 增加了 WebSnap 技术。WebSnap 内含有大量开发 Web 应用的组件, 以可视化的方式开发, 增加了 Session 管理、Servier-Side Script 等非常好的功能。但是由于 WebSnap 的开发步骤、开发思想与传统的 Delphi 程序设计差别太大, 开发人员要接触和学习大量的新概念、新办法, 因而不能快速地应用, 同时由于 WebSnap 过于庞大, 使得其稳定性大打折扣。

(3) IntraWeb。IntraWeb 是 Delphi 7.0 新增加的开发 Web 的控件, 它另辟蹊径, 允许程序员在 IDE 环境下像开发传统的 Delphi 程序一样开发 Web 应用, 还可以利用模板、CSS 文件自由地美化和设置网页的外观, 是完全的可视化设计。使用 IntraWeb 可以最大限度地保证 Delphi 程序员的知识积累, 而且其新的版本已经开始支持 Java 和 .NET 环境, 是 Web 开发应用的很好发展方向。

鉴于 Delphi 开发 Web 应用的技术重点, 本书对传统高效的 WebBroker 和新兴快捷的 IntraWeb 做了深入细致的探讨。

本书上篇(第 1~3 章)由权重民(E-mail:xalion@263.net)编著, 由他开发的 Delphi 窑洞(<http://www.51delphi.com>), 是国内第一个完全由 WebBroker 技术架构的技术类站点, 在广大 Delphi 开发人员当中享有极高的声誉。第 1 章重点介绍了 Apache 运行环境及 Oracle 数据库的安装过程, 第 2 章介绍了 WebBroker 开发的流程、技巧及注意事项, 第 3 章则公开了 Delphi 窑洞的全部源代码, 同时加注了详细解释, 操作性极强。

本书下篇(第 4~6 章)由董立达(E-mail:bnpyse@sina.com)编著。该部分首先介绍了 IntraWeb 环境几种不同的工作模式、开发流程及发布过程等; 接着详细介绍了 IntraWeb 控件及市场上著名的第三方控件 TMS IntraWeb Component Pack Pro 的各种功能, 重点剖析了 CSS、JavaScript 在 IntraWeb 中应用, 并附加了实例进行说明; 最后以一个完整的实例介绍了 IntraWeb 开发的全过程。

本书的实例均由问鸿滨进行全面测试。

三位作者均工作在开发一线, 因此本书的实用性较强, 适合 Delphi/Kylix/C++ Builder 程序员、网站开发与设计人员、高校学生及对 Web 开发有兴趣的读者阅读。

由于作者水平有限, 书中如有错误之处, 敬请广大读者批评指正。

作 者

目 录

前言

上篇 Web Broker 网站开发与案例

第 1 章 Apache 与 Oracle 简介 1

1.1 开发跨平台的 Web 服务器——

Apache 1

1.1.1 Apache 的发展历程 1

1.1.2 Apache 的主要特征 1

1.1.3 Apache 的安装过程 3

1.1.4 跨平台的 Web 服务器 7

1.2 Oracle 数据库简介 9

1.3 Oracle 服务器的安装 12

1.4 Oracle 客户端的安装 20

1.5 使用 PL/SQL Developer 开发和 管理 Oracle 30

1.5.1 PL/SQL 的基本功能 30

1.5.2 建立 Oracle 用户 35

1.5.3 建立数据表 38

1.6 Oracle 的 Net 功能及其 应用 42

第 2 章 WebBroker 开发 46

2.1 WebBroker 能做什么 46

2.2 WebBroker 中的重要对象 47

2.2.1 TWebModule 47

2.2.2 TWebRequest 和 TWebResponse 49

2.2.3 TPageProducer 51

2.3 WebBroker 的开发技巧 53

2.3.1 开发流程 53

2.3.2 上传文件 54

2.3.3 使用 Cookie 保存状态 57

2.3.4 保存状态的另一种方法 60

2.3.5 数据流 60

2.3.6 在网页中使用 TeeChart 62

2.3.7 提高网站的效率 65

第 3 章 用 WebBroker 开发的 实例 68

3.1 网站数据库 68

3.1.1 访问人次累计表 (ALLUSER) 68

3.1.2 技术资料表 (BOOK) 69

3.1.3 访问 IP 记录表 (IP) 71

3.1.4 技术讨论表 (JSTL) 73

3.1.5 技术文章库 (JSWZ) 75

3.1.6 友情链接表 (LINK) 77

3.1.7 下载软件表 (SOFT) 78

3.1.8 软件评论表 (SOFTTL) 80

3.1.9 用户库 (YHK) 82

3.2 网站源程序 83

3.2.1 主 WebModule 83

3.2.2 页面产生控件 84

3.2.3 WebModule 中的 Action 事件 91

3.2.4 WebModule 单元的源代码 92

下篇 IntraWeb 开发与技巧

第 4 章 IntraWeb 基础应用 155

4.1 IntraWeb 的版本介绍 155

4.2 IntraWeb 开发与传统 Web 开发 的不同 156

4.2.1 为 Weblications 定制的 IntraWeb 157

4.2.2 建立 Weblications 就像建立一个 普通的应用程序 157

4.3 我的第一个 IntraWeb 程序 158

4.3.1 建立一个小型的数据库 158

4.3.2 建立登录用的窗体 159

4.3.3 用 Template 来美化窗口 160

4.3.4 用图片来美化窗口	161	4.7 TMS 的表格控件	241
4.3.5 小结	162	4.7.1 TMS 表格的结构样式	241
4.4 IntraWeb 工作模式	163	4.7.2 页面控制 (Controller)	241
4.4.1 DLL 模式的开发流程	163	4.7.3 表头控制 (Header)	243
4.4.2 DLL 模式开发程序的 发布	164	4.7.4 列的控制 (Columns)	244
4.4.3 DLL 模式开发程序的 卸载	167	4.7.5 脚注控制 (Footer)	248
4.4.4 应用模式程序的安装	168	4.7.6 TMS 表格的高级应用	249
4.4.5 应用模式程序的卸载	169	第 5 章 IntraWeb 高级应用	253
4.4.6 页模式	170	5.1 IntraWeb 中 CSS 的应用	253
4.4.7 从应用模式到 DLL 模式	171	5.1.1 CSS 简介	253
4.4.8 从 DLL 模式到应用模式	172	5.1.2 CSS 技巧分类	256
4.4.9 小结	173	5.1.3 在 IntraWeb 中使用 CSS	258
4.5 IntraWeb 控件介绍与 应用	173	5.2 IntraWeb 中 JavaScript 的 应用	261
4.5.1 标准控件 IWStandard	173	5.2.1 JavaScript 简介	261
4.5.2 数据控件 IWData	201	5.2.2 IntraWeb 使用的 JavaScript 剖析	273
4.5.3 客户端 IWClientSide	206	5.2.3 IntraWeb 使用 JavaScript 的 要点	307
4.5.4 小结	208	5.3 HTML 标记的应用	318
4.5.5 控制控件 IWControls	208	第 6 章 IntraWeb 开发的实例	320
4.5.6 杂项	219	6.1 数据结构	320
4.6 TMS For IntraWeb 控件 简介	220	6.2 数据模块	326
4.6.1 TMS 内嵌 HTML 编辑器	220	6.3 ServerController 模块	333
4.6.2 TMS 的标签文字类控件	222	6.4 窗体与模板文件	335
4.6.3 TMS 的编辑类控件	227	6.4.1 CSS 样式表	335
4.6.4 TMS 的列表类控件	230	6.4.2 登录窗体	338
4.6.5 TMS 的日期类控件	233	6.4.3 主窗体	344
4.6.6 TMS 的菜单类控件	235	6.4.4 进料表	349
4.6.7 TMS 的图形类控件	237	6.4.5 其他	352
4.6.8 TMS 的面板类控件	239	6.4.6 小结	352

上篇 Web Broker 网站开发与案例

第 1 章 Apache 与 Oracle 简介

1.1 开发跨平台的 Web 服务器——Apache

Apache 是一个功能强大、非常灵活的 Web 服务器。到目前为止，大约有超过 700 万台的 Apache 服务器在 Internet 上运行，数量居各种运行在 Internet 上的服务器之首。

1.1.1 Apache 的发展历程

Apache 起初只是应用在 Unix 系统的 Web 服务器上，但随着开发者对其所作的修改越来越完善，其功能也日益强大。时至今日，Apache 已能够应用在几乎所有的计算机开发平台上，如 Unix、Linux、Netware5.x、Windows95/98/NT/2000 上，对于其他许多服务器的操作平台（如 OS/2、Amiga 等），也有 Apache 的用武之地。

在 1995 年以前，由于当时商品化的软件尚不能够满足应用的需要，大多数用户对 Web 服务器软件都不满意，于是美国国家计算机安全协会（NCSA）在 1995 年编写了一个 Web 服务器。该服务器当时看起来很好，但后来，NCSA Web 服务器的核心开发人员几乎都离开了 NCSA，致使这个开发项目一度陷入停顿。与此同时那些使用 NCSA Web 服务器的人们开始交换他们为该服务器编写的补丁，这使他们认识到成立一个管理这些补丁程序的组织很有必要。于是，Apache Group 应运而生，他们把该服务器软件称为 Apache。

Apache 源自 A Patchy Server 的读音，意思是充满补丁的服务器，这些软件精英们开发了这样一个项目，而最终创造了世界上一流的 Web 服务器软件。

Apache Group 只对那些提交到他们手中的新代码进行审查、完成测试、并进行质量控制检查。如果对提交的代码感到满意，那么这些代码将被集成到 Apache 的主要发行版中。

1995 年 4 月，Apache（0.6.2 版）公开发布，1.0 版则于 1995 年 12 月 1 日发布。到 2002 年底，Apache 的最稳定版本为 1.3.27，同时，在 2001 年底，Apache 2.0 的测试版本就已经出来了。Apache 2.0 采用全新的构架，在性能上完全超过了先前版本。通过初始化一种全新线程，它在某些情况下可以更快的对请求作出应答。同时，新版本还在与 Windows 的兼容性上作了改进，使这一原本为 Unix/Linux 系统设计的服务器能够更好的兼容 Windows 系统。到 2003 年 3 月，Apache 2.0 的正式稳定版为 2.0.44。

1.1.2 Apache 的主要特征

Apache 的最大特点是它所用的开发模式：开放源代码。开放源代码的优点有两个：一是软件免费；二是开发的产品功能非常强大。

由于 Apache 为自由软件,所以它可以吸引全世界很多的开发人员。今天,他们已为 Apache 编写出了许多优秀的模块,而这些模块能满足用户对功能的需要。如果这些模块都不是你想要的,那么你可以开发出自己想要的模块。

除了上述特征,Apache 还有许多重要的特征:

支持最新的 HTTP1.1 协议——Apache 是最先使用 HTTP1.1 协议的服务器之一。Apache 与 HTTP1.1 的完全兼容及其对 HTTP1.0 的向后兼容使它具有更加卓越的性能。例如,HTTP1.1 出现之前,Web 浏览器在发出另一个请求之前,必须等待来自 Web 服务器的响应。HTTP1.1 出现之后,就不会出现这种情况。现在,Web 浏览器可以并发的发送请求,而且不需要在每个请求中都传送 HTTP 头标,从而节省了带宽。这使最终用户端的性能得到提高,并使平行请求的文件在浏览器中显示得更迅速。

简捷强大的配置——Apache 服务器没有为管理员提供图形用户界面。但是,Apache 提供了三个简单的文本配置文件,通过这些文本可以对 Apache 进行定制。如果将这三个文件合并成一个文件,还可以大大的简化配置任务。

支持虚拟主机——Apache 是首批既支持基于 IP 的虚拟主机,又支持命名虚拟主机的 Web 服务器之一。

支持“服务器端包含”命令(SSL)——Apache 提供扩展的“服务器端包含”命令,这为 Web 站点开发人员提供了更大的灵活性。通过在 Apache 中使用经过扩展的 Server Side Includes 命令,可以完成任何可以通过 SSL 完成的事情。

支持“通用网关接口”(CGI)——Apache 用 mod_cgi 模块来支持 CGI,该模块遵守 CGI1.1 标准。此外,还提供一些扩展的特性,如定制环境变量和调试支持功能等。

支持 HTTP 认证——Apache 不但支持基于 Web 的基本认证,它还为支持消息摘要的认证做好了准备(后者在目前一些流行的服务器上尚未实现)。Apache 通过使用标准的口令文件、DBM、SQL 调用或对外部认证程序的调用实现基本认证。

服务器状态和可定制的日志——在记录日志和监视服务器本身状态方面,Apache 为我们提供了许多不同的选项。例如,可以通过 Web 浏览器来监视服务器的状态,也可以根据自身需要定制日志文件。与支持服务器包含命令相似,任何可以由 SSL 做的事情,都可以通过 Apache 中使用扩展的 Server Side Includes 命令来完成。

用户会话过程的跟踪功能——通过使用 HTTP Cookies(一个称为 mod_usertrack 的 Apache 模块),可以在用户浏览 Apache Web 站点时对用户进行跟踪。

集成的 Perl——Perl 早已成为 CGI 脚本编程的标准语言。Apache 是使 Perl 成为流行的 CGI 编程语言的重要因素之一。现在,Apache 能更好地支持 perl,通过使用其 mod_perl 模块,可以将基于 Perl 的 CGI 脚本装入内存,而且可以根据需要多次重复使用该脚本,从而节省了启动开销。

集成的代理服务器(Proxy)——可以将 Apache 作为正向代理服务器。但是,当前可选的代理模块尚未支持逆向代理或最新的 HTTP1.1 协议。幸运的是,这一模块能够很快得到更新。

支持 FastCGI——并不是所有人都会采用 Perl 编写 CGI 程序,那如何使 CGI 应用程序运行得更快呢?Apache 为此提供了一个完美的解决方案。在 Apache 中使用 mod_cgi 模块来实现 FastCGI 环境,并使 FastCGI 应用程序运行得更快。

支持安全 Socket 层 (SSL) —— 由于版权法和美国法律在进出口方面的限制, Apache 本身不支持 SSL。但是, 有两种方法可以满足 SSL 的需求, 即下载 Apache 的补丁程序和购买商业化 Apache 的衍生产品

支持 Java Servlets —— 一个称为 mod_jserv 的试验性 Java Servlet(alpha) 模块可用于 Apache 中。当模块正式发布的时候, 它将允许 Apache 运行服务器端的 Java 应用程序, 且应用程序能充分利用多线程优点和其他的 Java 特性。

与其他服务器相比, Apache 具有更高速、更稳定、功能更齐全的优点。尽管有些商业化服务器具有比 Apache 更高速的特点, 但获得那样的服务器, 可能要付出很大的费用。所以, Apache 仍然是最受人们欢迎的服务器之一。

1.1.3 Apache 的安装过程

从 <http://www.apache.org> 可以下载到最新版本的 Apache 服务器, 笔者用的是 apache_2.0.43-win32-x86-no_ssl.exe, 双击运行, 出现以下画面, 点击 Next, 如图 1-1 所示。

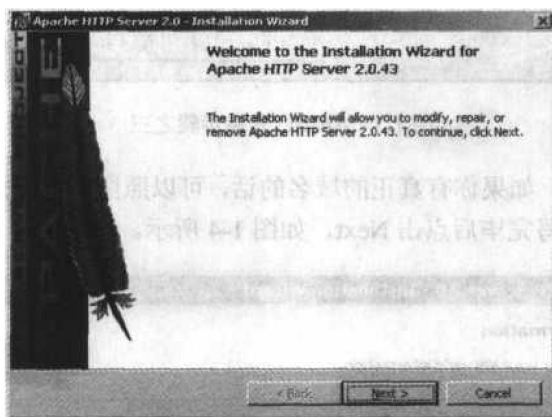


图 1-1 Apache 安装步骤之一

出现版权信息界面, 选择接受, 然后点击 Next, 如图 1-2 所示。

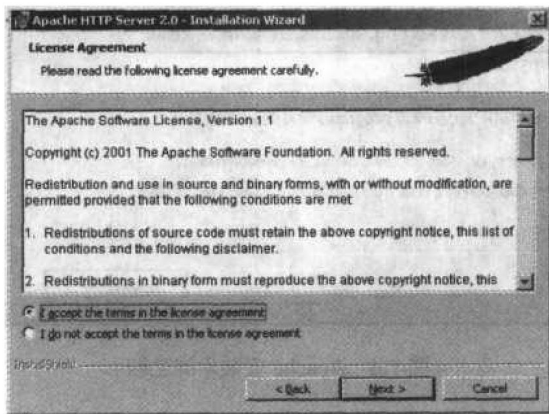


图 1-2 Apache 安装步骤之二

出现 Apache 基本信息介绍页面，点击 Next，如图 1-3 所示。

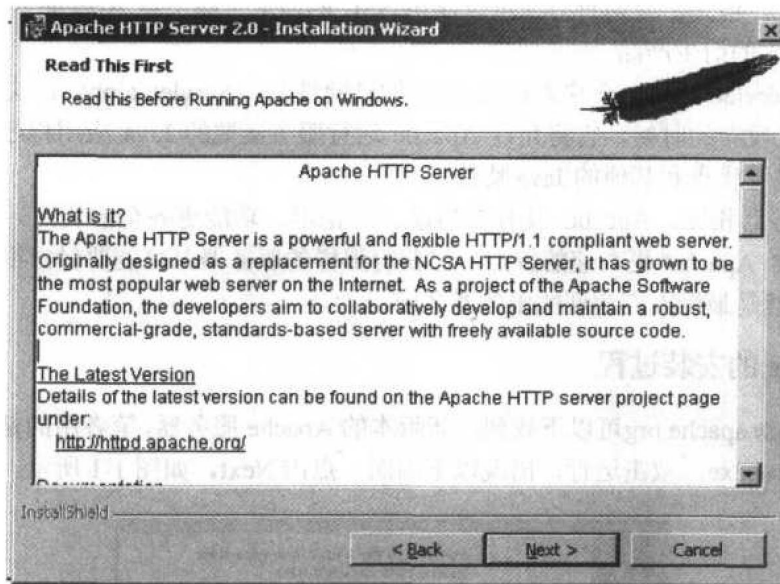


图 1-3 Apache 安装步骤之三

出现域名设置页面，如果你有真正的域名的话，可以照图示的方式填写，否则就随便写，不会影响正常使用。填写完毕后点击 Next，如图 1-4 所示。

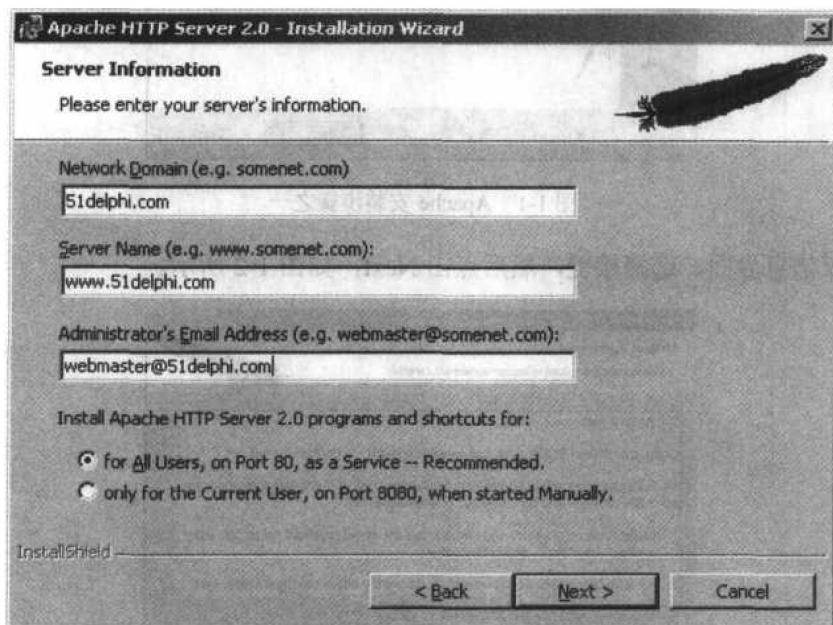


图 1-4 Apache 安装步骤之四

出现安装类型界面，在这里建议选定制方式，因为安装的默认路径为 C 盘这样对以后的调试，备份都不方便，因此选 Custom 方式，点击 Next，如图 1-5 所示。

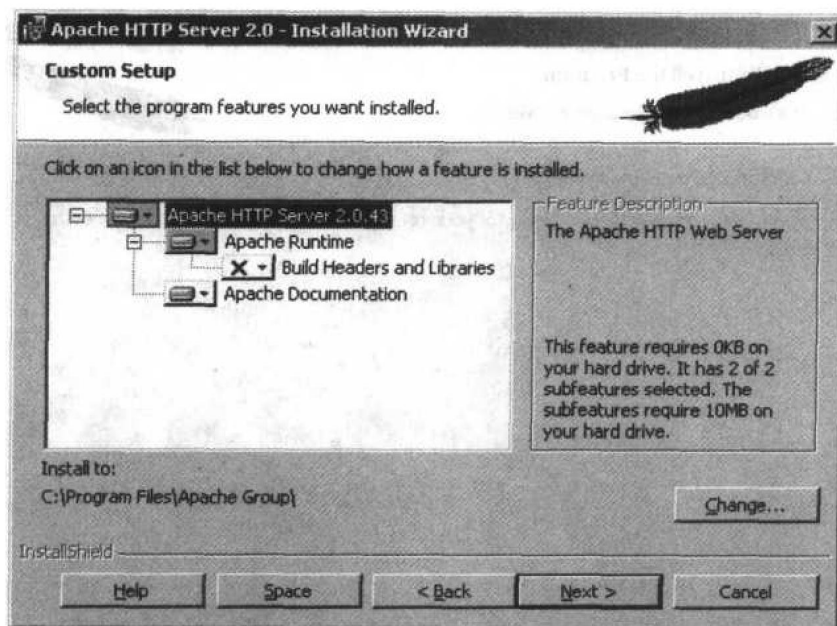


图 1-5 Apache 安装步骤之五

更改安装路径 c:\program Files\Apache Group\为 d:\ Apache Group\, 其他的不用管, 如图 1-6 所示。

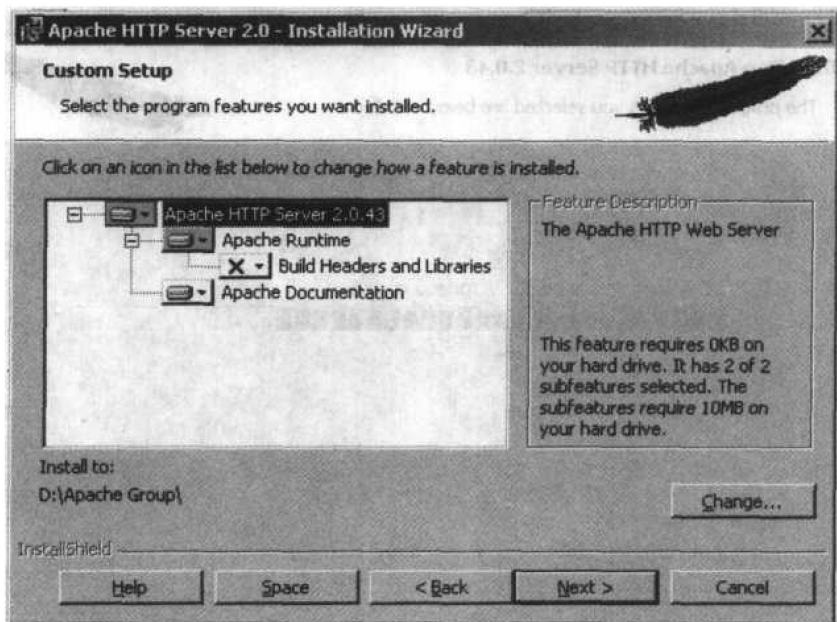


图 1-6 Apache 安装步骤之六

点击 Next, 如图 1-7 所示。

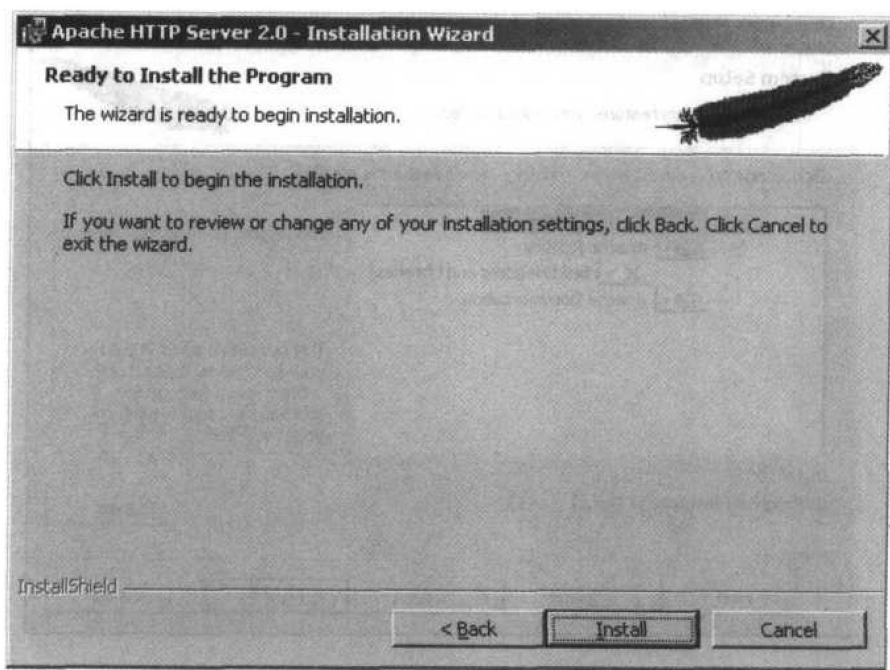


图 1-7 Apache 安装步骤之七

点击 **Install**，开始安装，如图 1-8 所示。

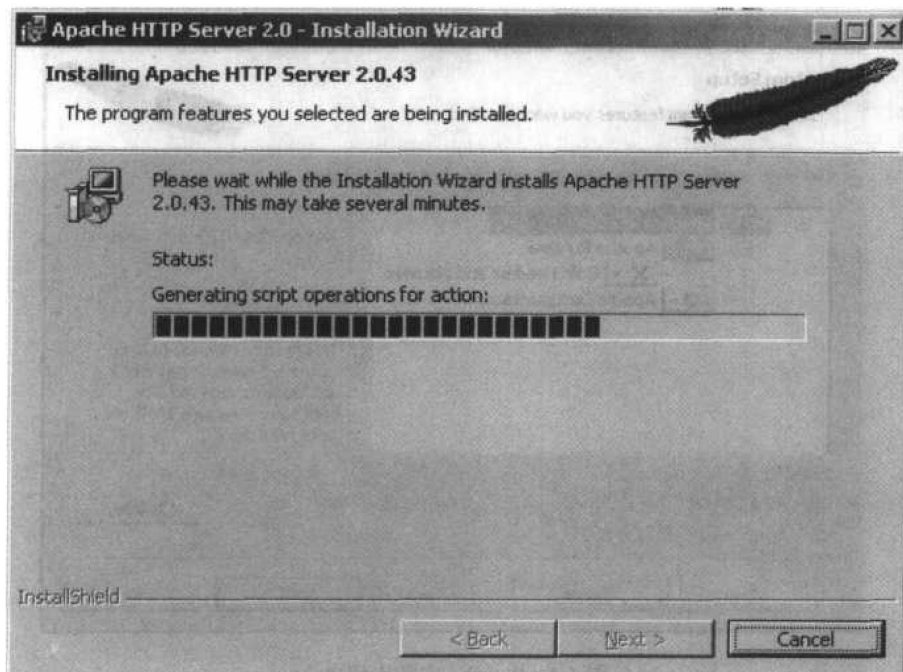


图 1-8 Apache 安装步骤之八

安装完成，如图 1-9 所示。

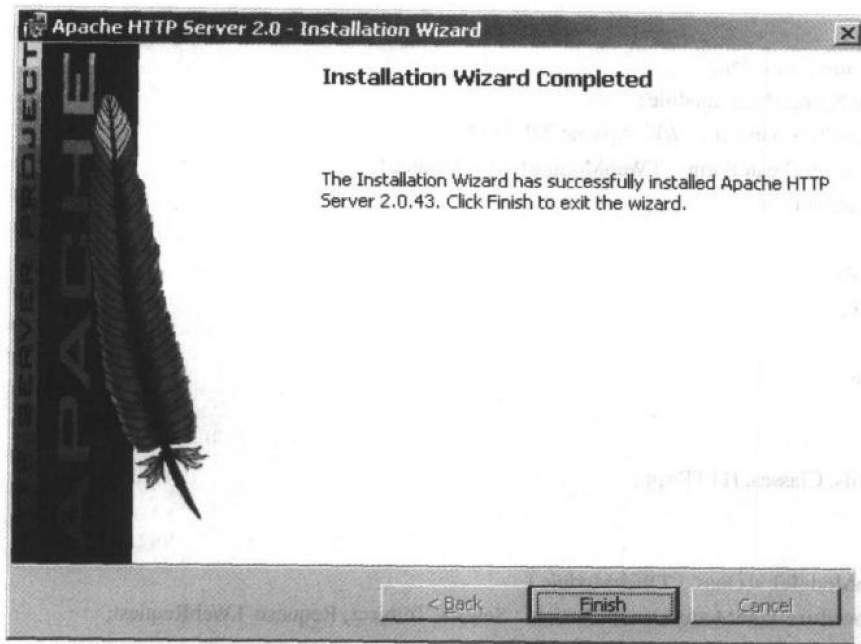


图 1-9 Apache 安装步骤之九

这样在任务栏就会显示如图 1-10 所示图标。



图 1-10 Apache 安装后之状态栏

以后就可以使用该图标启动、停止 Apache 服务器了。

1.1.4 跨平台的 Web 服务器

Delphi7 和 Kylix3 都支持 Apache Http Server，它们是通过支持 DSO（动态共享模块）来支持 Apache 的。我们可以建立一个 Web 服务器，只要分别用 Delphi7 和 Kylix3 编译一下，就可以分别在 Windows 和 Linux 下运行。程序代码如下：

```
test.dpr:
library test;
uses
  WebBroker,
  ApacheAppTWO, //声明支持 Apache 2.0
  main in 'main.pas' {WebModule1: TWebModule};
{$R *.res}
exports
  //动态模块的名称
  apache_module name 'test_module';
```

```

begin
    Application.Initialize;
    ModuleName:='test_module';
    handler:='test-handler'; //以 Apache 2.0 为例
    Application.CreateForm ( TWebModule1, WebModule1 );
    Application.Run;
end.
Mainp.pas:
unit main;

interface

uses
    SysUtils, Classes, HTTPApp;

type
    TWebModule1 = class ( TWebModule )
        procedure WebModule1indexAction ( Sender: TObject; Request: TWebRequest;
            Response: TWebResponse; var Handled: Boolean ) ;
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    WebModule1: TWebModule1;

implementation

{$R *.DFM}

procedure TWebModule1.WebModule1indexAction ( Sender: TObject;
    Request: TWebRequest; Response: TWebResponse; var Handled: Boolean ) ;
begin
    //显示内容到浏览器
    response.Content:='Hello,This is a apache Web server!';
end;
end.

```

在 Delphi 7 下编译的结果为 test.dll，在 Kylix 3 下编译的结果为 test.so。
在 Apache 的 httpd.conf 的配置为：

```

LoadModule test_module "d:/Apache Group/Apache/modules/test.dll"
//指向模块的相应位置
<Location /test>
    SetHandler test-handler

```

</Location>

关闭 Apache 服务，再重新启动 Apache 服务，以便 Apache 加载该模块，然后在浏览器里面输入：`http://localhost/test`。

浏览器就会输出以下界面，如图 1-11 所示。

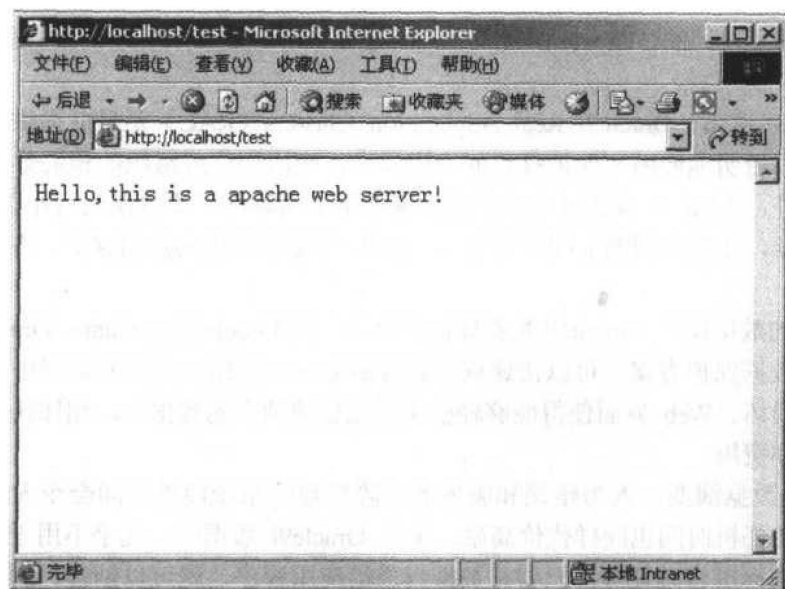


图 1-11 Apache 运行测试例程

这样就实现了跨平台的 Web 服务器，当然你可以扩充 `test.dpr`，使它来满足你的不同功能。

1.2 Oracle 数据库简介

Oracle9i 数据库是最先进的面向对象的数据库，获得《PC Magazine》杂志的编辑选择奖，同时还被 Linux Journal 评为 Linux 上的最佳数据库。Oracle9i Database 是目前业内伸缩性最好、功能最齐全的数据库。无论是用于驱动网站、打包应用程序、数据仓库或者是 OLTP 应用程序，Oracle9i Database 都是任何专业计算环境的技术基础。Oracle 9i 具有以下优点。

1. 事务处理

(1) 行级锁定。总的来说，绝大部分多用户数据库在处理冲突（包括不同用户的交互、应用程序同时访问或更新同一数据等操作）时通常需要锁定整个数据表，从而使所有其他的事务处于等待状态。对于 Oracle9i 数据库，只有包括了所述数据的单个行才被锁定。因此，多个用户可以从同一个表访问数据，并确保能得到快速、准确的结果。

(2) 实体化视图。实体化视图用于预计算和存储集合数据，例如每月销售数据总额。存储预计算结果使得数据库服务器可以显著提高普通查询的性能，因为它不需每次再重新计算这些结果。

(3) 数据分区。分区允许表和索引细分为更小的单位,从而能够更细粒度地管理和访问数据库对象。分区能够提高可管理性、性能和可用性,因此为应用程序带来巨大的好处。分区可以将某些查询和维护操作的性能提高好几个数量级。

(4) 查询优化。查询优化对关系数据库的性能,尤其对复杂 SQL 语句至关重要。查询优化程序不是在实验室中基于纯粹的理论推测和假想而开发出来的,它是通过不断适应实际需求客户的需求而精心设计的。Oracle 的优化程序可能是业界经过多次证明的最佳优化程序。

2. 可靠性

(1) 数据库群集。Oracle9i Real Application Clusters (RAC) 是目前提供的惟一的群集数据库,它可以透明地伸缩和保护打包的应用程序,应用程序或组织的相关数据几乎不需要任何修改。此外,RAC 可以通过不同价格点来伸缩应用程序,不管是运行小型的 Linux 或 Windows 服务器,还是大型的 UNIX 服务器。RAC 还提供出色的数据保护,在不到 30s 内实现热故障转移。

(2) 完整的数据保护。Oracle9i 数据库的特性之一是 Oracle Data Guard。Oracle Data Guard 是一种完整的数据保护方案,可以快速恢复各种故障——从站点故障和数据中心灾难,到人为错误和数据损坏。Web 界面使得能够轻松无误地创建独立的数据库,相比传统的方法,节省了大量时间和费用。

(3) 在线数据演变。人为错误和服务器故障导致的系统停机时间会令人非常沮丧,而由于维护造成的停机时间也同样代价高昂。采用 Oracle9i 数据库,几乎不用干预系统操作就可以执行所有的标准管理任务。只要将数据提供给应用程序,就可以创建、删除或移动表、改变系统参数和重组数据库。

(4) 自助式纠错。Oracle9i 数据库的闪回查询特性允许应用程序开发人员将纠错功能构建到他们的应用程序中,这样便可以纠正终端用户的错误,而无需对数据库执行恢复操作。使用闪回查询,应用程序可以查看昨天或指定的任何时间的数据。Oracle9i 数据库可以按照请求时间时的数据原样重构数据。

3. 数据仓库

(1) 集成的 OLAP。由于 Oracle9i OLAP 充分集成到关系数据库中,所有的数据和元数据都在 Oracle9i 中存储和管理,从而提供了卓越的可伸缩性、强健的管理环境,以及具有行业领先的可用性与安全性。

(2) 数据挖掘。Oracle9i Data Mining 允许公司构建高级的商务智能应用程序,挖掘公司数据库以发现新的见解,并且将见解集成到商务应用程序中。Oracle9i Data Mining 将数据挖掘功能嵌入到 Oracle9i 数据库中,用于分类、预测和关联。

(3) 数据仓库提取—转换—加载 (ETL)。对于数据仓库,通常从各种事务系统和数据流中收集数据。来自各种信息源的数据必须首先提取,然后转换成单一的、一致性的视图,最后加载到数据仓库中。通过让这些关键操作内置于 Oracle9i 数据库中,由于能够并行运行各种 ETL 操作,因而能够获得更大的灵活性和节省更多的时间。

4. 管理工具

(1) 一次性登录。Oracle Advanced Security 通过在安全套接层 (SSL, 即 Internet 认证标准) 上使用可互操作的 X.509 (版本 3) 证书用于认证,提供增强的基于 PKI 的一次性登录。除了强大的用户认证,SSL 还为多种类型的连接提供网络数据机密性和数据完

整性: LDAP(轻型目录访问协议)、IIOP(Internet Intra-ORB 协议)和 Net(以前称为 Net8)。

(2) 有选择性的数据加密。在 Oracle9i 数据库中能够本地加密数据,这使得应用程序即使在外来的入侵者访问服务器时,也能够保护他们的敏感数据。Oracle9i 在可导出的码长度和 Triple-DES(3DES)中使用业界标准的数据加密标准,提供基于服务器的加密和解密操作。

(3) 安全的数据共享。Oracle Label Security 是一个强大的行级安全解决方案,基于政府和国防机构使用的标签概念,用于保护敏感的信息和提供数据分离。传统的信息系统不允许数据在一个数据库中分成不同的敏感级别。Oracle9i Label Security 对分配给某一应用程序用户的标签授权,与分配给某一数据块的敏感标签进行比较,从而调解对数据的访问。

5. 数据集成和消息发送

(1) 智能自我管理和优化。Oracle9i 数据库中的一个关键核心领域是通过自动化日常 DBA 任务,增强数据库可管理性,降低管理的复杂性,使之能够更好地自我优化和自我管理。许多新特性已经增加到其中,以简化空间、内存、资源管理以及其他日常的数据库管理任务。

(2) 管理整个堆栈。Oracle Enterprise Manager 提供全面的管理框架,用于支持多个异种环境。它还提供实时的时间监控、无与伦比的可伸缩性、分布式的数据库管理、性能和可伸缩性数据的收集和分析、Oracle 环境的自动优化,以及充分集成的报表生成功能。

(3) 定位诊断。Oracle Diagnostics Pack 提供监控 Oracle 环境所需的工具,包括:数据库、主机系统、Web 服务器和 Oracle 应用程序。它还包括 Oracle 诊断的核心方面:指示灯关闭(lights-out)监控和通知、实时诊断、历史数据分析和基于 Web 的报表生成。

(4) 数据库资源管理。Oracle9i 数据库资源管理器按照数据库管理员定义的业务优先级应用系统资源,因而能够区分数据库操作的优先顺序。它还使得管理员能够在数据库用户和应用程序间分配 CPU 资源,以及限制任何操作的并行度。它还能够自动地检测运行时间长的操作,限制它们的资源使用,从而预防系统大范围的性能下降。

6. 内容管理

(1) 消息队列。Oracle Advanced Queuing 是 Oracle9i 数据库惟一集成的消息队列特性。消息队列功能允许应用程序通过 AQ 队列中的消息异步地交流。与数据库的集成带来空前的高级功能、操作简易性、可靠性,以及消息队列的安全性。它还为消息队列应用程序提供独特的消息管理功能,例如审计和追踪。

(2) 数据库复制。复制是在构成分布式数据库系统的多个数据库中,拷贝和维护数据库对象(例如表)的过程。在多个位置都可以获得同样的数据。Oracle Advanced Replication 设计用于支持各种应用程序。它可以支持大量的部署配置,以及集成在一个分布式计算平台中的服务器到服务器的应用程序。

(3) 原有数据库数据网关。Oracle 提供两个连通性解决方案来满足异类的数据库访问需求:一般的连通性(Generic Connectivity)和透明的网关(Transparent Gateways)。这些网关提供无缝地访问非 Oracle 数据源,例如 DB2、Sybase、Ingres 和许多其他源。一般的解决方案使用标准的 ODBC 机制访问这些数据源,而透明网关则对选定的原有数据库使用本地操作。