



Java Servlet 程序开发权威指南

沈兆阳 / 编著

- ◇ Java Servlet 是 Java 的一部分，是开发服务器端应用程序的重要组件。
- ◇ 本书就是介绍如何使用 Java Servlet 来取代 CGI 程序，以产生高安全性丰富的动态网页内容。
- ◇ 本书为读者设计了一个完整的范例——电子布告栏，该范例使用了许多 Java Servlet 的设计技巧，可以使您对设计一个 Java Servlet 来产生动态的网页，具有更加完整的概念，并掌握实际操作技巧。

 随书附赠光盘，内含所需使用到的软件与范例的程序代码



中国青年出版社



文魁资讯股份有限公司

e 时代网络学科推荐教程
E-time Network Discipline Recommend Textbooks

Java Servlet 程序开发权威指南

沈兆阳 / 编著



中国青年出版社
CHINA YOUTH PRESS

(京)新登字 083 号

本书简体中文版由文魁资讯股份有限公司授权中国青年出版社独家出版。未经出版者书面许可,任何单位和个人不得以任何形式复制或传播本书的部分或全部。

版权贸易合同登记号: 01-2001-2688

策 划: 胡守文

王修文

郭 光

责任编辑: 江 颖

朱新媛

责任校对: 肖新民

书 名: 《Java Servlet 程序开发权威指南》

ISBN 7-5006-4493-0/TP·209

编 著: 沈兆阳

出版发行: 中国青年出版社

地址: 北京市东四十条 21 号 邮政编码: 100708

电话: (010) 84015588 传真: (010) 64053266

印 刷: 北京机工印刷厂

开 本: 16 开

版 次: 2001 年 8 月北京第 1 版

印 次: 2001 年 8 月第 1 次印刷

印 数: 1-5000

定 价: 63.00 元 (附 1CD)

序

沈兆阳毕业于交通大学计算机工程系与研究所，曾经任职于 Informix 与 Sybase 两家著名的数据库厂商，目前是专职的作家。

Java 刚开始是 Sun Micro 公司为了小型的嵌入式系统而设计的，但是其最重要的功能却是在设计 Applet 上，以提供用户端安全而且丰富的网页内容。在 Java 程序语言之中，现在大家已经开始重视它所提供服务器端应用程序的功能。

Java 是一种高安全性的程序语言，它的特性非常适合用来开发客户/服务式的应用程序，因为它是跨平台的，所以也非常适合使用在大型的多主机系统架构中。Java 是一种面向对象的程序语言，这对熟悉面向对象语言的程序设计师来说，很快就可以上手而且不会造成学习上的困难。

Java Servlet 是一个 Java 的类别，它可以用来扩充服务器的功能。现在 Java Servlet 可以取代传统的 CGI 程序，以产生内容丰富的动态网页。Java Servlet 是在 Java 虚拟机器上执行的，所以不仅安全性高而且具有可携性。因为 Java Servlet 是在服务器上执行的，所以开启 Java Servlet 的浏览器并不需要支持 Java。

因为科技专用术语大多是以英文来描述的，所以作者最先面对的问题就是专有名词的翻译，我们的翻译着眼点为易于理解与保留原义，而且中文名词长度不宜过长。本书是专为动态网页设计师而撰写的，作者以深入浅出的方式、长期的程序设计经验以及循序渐进的内容安排，使得学习信息新知不再是一件苦差事。

我们的撰写方法是采用「面向操作」的方式，就算是读者一时间无法理解文句中的精神，也可以通过实际的操作来体会。如果读者对 Java Servlet 所支持的功能有任何的疑问，请寄电子邮件至沈兆阳(cyhsieh@mail.ht.net.tw)的电子邮件信箱，作者将尽速为您解答问题。

在信息科技的领域中，因为涵盖广阔与进步神速，没有人敢自称是专家。作者希望读者朋友们能不吝指教，多多赐予宝贵意见，只有通过不断的经验交流，才能够使大家一起进步。最后我要说，谢谢文魁信息公司洪锦魁先生的协助与支持，使得本书得以顺利出版。

沈兆阳 2001.06

本书导读

Java Servlet 是开发服务器端应用程序的重要组件。在过去，我们大都是使用 CGI(Common Gateway Interface)来产生动态的网页内容。当 WEB 服务器接收到用户端所提出的要求之后，系统会产生一个新的 CGI 程序，再利用标准输入与环境变量来传递参数，处理完毕之后 CGI 程序就会自动结束。每一个 CGI 程序只能服务一个用户端的要求，这样不仅浪费了系统资源，而且降低了整体的执行效能。

Java Servlet 是由 Sun Micro 的 Java 软件部门所制定的一个标准延伸，是 Java 程序语言的一部分，本书的重点是在介绍如何使用 Java Servlet 来取代 CGI 程序，以产生丰富的动态网页内容。

本书共分为以下的十四章：

- 第 1 章 简介与安装
- 第 2 章 HTML
- 第 3 章 Java 程序语言介绍
- 第 4 章 一般性介绍
- 第 5 章 计数器
- 第 6 章 执行环境
- 第 7 章 识别一用户
- 第 8 章 Cookie
- 第 9 章 会话
- 第 10 章 关系数据库
- 第 11 章 数据库存取
- 第 12 章 常用的设计技巧
- 第 13 章 SSI 与 JSP
- 第 14 章 电子布告栏

如果您已熟悉 HTML，请跳过第 2 章；如果您已精通 Java，请跳过第 3 章。我们于第 14 章为读者设计了一个完整的范例——电子布告栏，该范例使用了前面所介绍的设计技巧，可以使您对设计一个 Java Servlet 来产生动态的网页有完整的概念。

本书的撰写方法是采用「面向操作」的方式，就算是读者一时间无法理解文句中的精神，也可以通过实际的操作来体会。

光盘说明

本书所附的光盘是所需要使用到的软件与范例的程序代码，其组成结构是如下所示：

- /Chapter01——第 1 章所需使用的软件
- /Chapter02——第 2 章所使用的范例
- /Chapter03——第 3 章所使用的范例
- /Chapter04——第 4 章所使用的范例
- /Chapter05——第 5 章所使用的范例
- /Chapter06——第 6 章所使用的范例
- /Chapter07——第 7 章所使用的范例
- /Chapter08——第 8 章所使用的范例
- /Chapter09——第 9 章所使用的范例
- /Chapter10——第 10 章所使用的范例
- /Chapter11——第 11 章所使用的范例
- /Chapter12——第 12 章所使用的范例
- /Chapter13——第 13 章所需使用的软件与范例
- /Chapter14——第 14 章所使用的范例
- /App01——附录 1 所使用的范例
- /App02——附录 2 所使用的范例
- /App03——附录 3 所使用的范例
- /App04——附录 4 所使用的范例
- /App05——附录 5 所使用的范例
- /App06——附录 6 所使用的范例

目 录

Chapter 1 简介与安装

1.1 简介	2
1.2 安装系统	2
1.3 安装 MySQL 数据库	3
1.4 安装 Apache WEB 服务器	5
1.5 安装 PHP	7
1.6 安装 phpMyAdmin	9
1.7 安装 Java SDK	11
1.8 安装 Java Servlet SDK	13
1.9 章末总结	18

Chapter 2 HTML

2.1 标题字	22
2.2 对齐控制	23
2.3 注解	24
2.4 段落与换行	24
2.5 水平线	26
2.6 blockquote	27
2.7 所见所得	28
2.8 ol 与 li	29
2.9 ul 与 li	30
2.10 表格与外框	30
2.11 表格的背景颜色	32
2.12 表格字段的对齐方式	33
2.13 文件内的超链接	34
2.14 跨文件/网络的超链接	37

2.15 以框架水平分割网页	38
2.16 框架分界的宽度	39
2.17 框架的外框	40
2.18 框架视窗的滚动	41
2.19 超链接的 target	42
2.20 输入单选项目	44
2.21 输入复选项目	46
2.22 输入文字	47
2.23 输入密码	47
2.24 下拉菜单	48
2.25 输入大量文字	49
2.26 章末总结	51

Chapter 3 Java 程序语言介绍

3.1 Java 简介与数据类型	54
3.2 运算符	59
3.3 修饰符	62
3.4 流程控制与例外处理	71
3.5 对象与类别	77
3.6 线程	80
3.7 章末总结	83

Chapter 4 一般性介绍

4.1 要求与回应	92
4.2 GET 与 POST	95
4.3 Servlet API	95
4.4 第一个 Servlet	98

4.5 在 Servlet 中取得参数	100
4.6 取得目前的日期	103
4.7 章末总结	104

Chapter 5 计数器

5.1 一个简单的计数器	110
5.2 计数器的同步问题	114
5.3 Enumeration 接口	116
5.4 Hashtable 类别	117
5.5 多重计数器	119
5.6 一个可永久计数的计数器	122
5.7 一个可以被共用的计数器	128
5.8 一个定期重新执行的计数器	135
5.9 章末总结	137

Chapter 6 执行环境

6.1 启动参数	140
6.2 AUTH_TYPE 与 REQUEST _METHOD	141
6.3 要求的标头	143
6.4 服务器的相关信息	145
6.5 字符编码	147
6.6 用户端信息	148
6.7 PathInfo	150
6.8 要求资源的 MIME 类型	152
6.9 取得参数	154
6.10 通信协议	156
6.11 内容的类型与长度	158
6.12 要求的 URI	160
6.13 要求的 SCHEME	161
6.14 Servlet 的路径名称	163

6.15 错误处理	164
6.16 章末总结	170

Chapter 7 识别一用户

7.1 使用隐藏式表格字段	174
7.2 使用 Hashtable	185
7.3 使用额外的路径名称	196
7.4 章末总结	201

Chapter 8 Cookie

8.1 简介	204
8.2 新增一个 Cookie	204
8.3 重新设定 Cookie 的数值	206
8.4 Cookie 的版本	208
8.5 Cookie 的注解	209
8.6 Cookie 的最大年龄	211
8.7 Cookie 的传送协议	213
8.8 使用 Cookie 识别使者	214
8.9 章末总结	225

Chapter 9 会话

9.1 简介	228
9.2 计数器	228
9.3 要求的会话识别号码	231
9.4 会话的相关信息	232
9.5 删除一会话	237
9.6 删除所有的会话	239
9.7 使用 HttpSession 识别使者	242
9.8 章末总结	251

Chapter 10 关系数据库

10.1 关系数据库简介	254
--------------------	-----

10.2 MySQL 数据库服务器	254	10.26 常用的汇总函数	281
10.3 兆阳电脑用品大盘商	256	10.27 集合数据排序与 HAVING 子句	284
10.4 执行脚本文件新建示范数 据库	257	10.28 在一 SELECT 语句中使用 两个表格	285
10.5 新增一数据库	260	10.29 在一 SELECT 语句中使用 多个表格	287
10.6 使用 mysql 用户端程序执 行 SQL 语句	261	10.30 外部合并	288
10.7 选取特定的字段	262	10.31 一表格与自己做合并	289
10.8 选取唯一的数值	263	10.32 LIMIT 子句	290
10.9 选取子字符串	263	10.33 SELECT INTO OUTFILE 语句	291
10.10 选取特定的记录	264	10.34 LOAD DATA INFILE 语 句	292
10.11 排除特定的记录	265	10.35 LOCK/UNLOCK 语句	293
10.12 选取介于某一范围的数 据	265	10.36 新增记录	293
10.13 选取属于某一子集合的 数据	266	10.37 REPLACE 语句	294
10.14 识别 NULL 数值	267	10.38 删除记录	294
10.15 搜寻完全相等的字符串	267	10.39 修订记录的内容	295
10.16 搜寻子字符串	268	10.40 用户名称与通行密码	296
10.17 使用 ORDER BY 子句	269	10.41 增加新的 MySQL 用户	296
10.18 使用算数运算式	270	10.42 章末总结	299
10.19 为虚拟字段命名	270		
10.20 在 SQL 语句中使用注解	271	Chapter 11 数据库存取	
10.21 搜寻单一长度的任意字 符串	272	11.1 JDBC 驱动程序	302
10.22 常用的字符串函数	273	11.2 Statement 接口与 ResultSet 接口	303
10.23 常用的数学函数	276	11.3 查询 MySQL 的数据	305
10.24 常用的时间函数	278	11.4 以表格方式输出数据	308
10.25 GROUP BY 子句	280	11.5 新增与修订表格数据	314

11.6 一个完整的数据库存取	
范例	317
11.7 查询员工数据	319
11.8 新增员工数据	324
11.9 修订员工数据	328
11.10 删除员工数据	332
11.11 章末总结	336

Chapter 12 常用的设计技巧

12.1 背景线程	340
12.2 使用持续性连接	343
12.3 显示文件的内容	345
12.4 变更网址	348
12.5 分担服务器工作负担	350
12.6 将 Servlet 锁定于一服务器	352
12.7 共用一个类别	354
12.8 扩充一自订的 Servlet 类别	362
12.9 应用系统组态文件	367
12.10 HTML 样板	370
12.11 转换换行字符	372
12.12 章末总结	373

Chapter 13 SSI 与 JSP

13.1 安装 SSI 与 JSP	376
13.2 撰写 SSI 文件	382
13.3 一个简单的 JSP 文件	384
13.4 在 HTML 文件中调用 Servlet	386
13.5 在 JSP 中使用变量	388
13.6 使用 request 对象	389
13.7 if 语句与注解	390

13.8 声明自订的方法	392
13.9 使用 switch 语句	393
13.10 传递输入值至 JSP	395
13.11 显示服务器相关信息	398
13.12 以长条图来显示量值	400
13.13 登入系统	402
13.14 INCLUDE 文件	407
13.15 import 套件	408
13.16 在 JSP 中声明类型	410
13.17 章末总结	412

Chapter 14 电子布告栏

14.1 功能描述与系统安装	416
14.2 登入管理系统	417
14.3 新增一个布告	441
14.4 浏览布告栏	446
14.5 修改一个布告	454
14.6 删除一个布告	462
14.7 取消删除一个布告	463
14.8 永远删除布告	467
14.9 变更系统组态	471
14.10 变更管理者密码	476

附录 A

附录 1 javax.servlet 的 API	480
附录 2 javax.servlet.http 的 API	495
附录 3 phpMyAdmin 管理工具	522
附录 4 PHP 介绍	551
附录 5 电子布告栏的原始码	594
附录 6 兆阳公司示范数据库内 容	634

简介与安装

Chapter 1

- 1.1 简介
- 1.2 安装系统
- 1.3 安装 MySQL 数据库
- 1.4 安装 Apache WEB 服务器
- 1.5 安装 PHP
- 1.6 安装 phpMyAdmin
- 1.7 安装 Java SDK
- 1.8 安装 Java Servlet SDK
- 1.9 章末总结

Java 刚开始是 Sun Micro 公司为了小型的嵌入式系统（Embedded System）而设计的，但是其最重要的功能却是在于设计 Applet，以提供用户端安全而且丰富的网页内容。在 Java 程序语言中，大家已经开始重视它所提供的服务器端应用程序的功能。

1.1 简介

Java 是一个高安全性的程序语言，它的特性非常适合用来开发客户/服务模式（Client/Server）的应用程序，因为它是跨平台的，所以也非常适合使用于大型的多主机系统架构之中。它是一种面向对象的程序语言，这对熟悉面向对象语言的程序设计师来说，很快就可以上手而且不会造成学习上的困难。

Java Servlet 是开发服务器端应用程序的重要元件，在过去，我们大都是使用 CGI（Common Gateway Interface）来产生动态的网页内容，当 WEB 服务器接收到用户端所提出的要求之后，系统会产生一个新的 CGI 程序（Process），再利用标准输入与环境变量来传递参数，处理完毕之后 CGI 程序就会自动结束。每一个 CGI 程序只能服务一个用户端的要求，这样不仅浪费了系统资源，而且降低了整体的执行效能。

Java Servlet 是一个 Java 的类别，它可以用来扩充服务器的功能。现在 Java Servlet 可以取代传统的 CGI 程序，产生内容丰富的动态网页。Java Servlet 是在 Java 虚拟机器（Virtual Machine）上执行的，所以不仅安全性高而且具有可携性。因为 Java Servlet 是在服务器上执行的，所以开启 Java Servlet 的浏览器并不需要支持 Java。

Java Servlet 是由 Sun Micro 的 Java 软件部门所制定的一个标准延伸，是 Java 程序语言的一部分，本书的重点是在介绍如何使用 Java Servlet 来取代 CGI 程序，以产生丰富的动态网页内容。

1.2 安装系统

Java Servlet 是与平台无关的，为了使得本书的内容更生动有趣，我们选择了 Linux 为工作平台，使得读者们可以实际的执行一些 Java Servlet，以查看它们的执行结果。实际上大部分的开放系统都支持 Java，我们之所以选择 Linux 为工作平台，主要是因为 Linux 是免费的软件，这样不会造成版权上的问题。

Linux 是一个免费的操作系统，它是由一个非正式的组织在维护着，使用它的最大好处是没有版权上的问题，但是安装起来可能就比较困难（您可以选用其他的平台）。

作者花了九牛二虎之力（也许有些夸张），终于安装并测试成功。因为 Linux 的版本众多，而且从网络上下载的安装文件并不是说明得很完整，所以请您尽量按照本章所说的方式安装系统，当然您也可以使用自己熟悉的工作平台与 Java，例如：Sun 的 Java Web Server 或是 IBM 的 WebSphere。

我们先来谈一下 Linux 操作系统本身，作者安装的是 Mandrake 7.0，在安装操作系统时请您一并安装[development]开发系统，因为我们会使用到 GNU 编译程序。请您不要在安装 Linux 时安装 MySQL 与 Apache Web 服务器。当您安装好了 Linux 操作系统之后，请使用 'rpm -qa | grep MySQL' 与 'rpm -qa | grep Apache' 指令检查系统上是否已经安装了与 MySQL 或是 Apache 相关的软件，是则请您使用 'rpm -e xxxxx' 指令加以删除。请您在一个“干净的”Mandrake 7.0 之上安装系统，因为使用过一阵子或是安装了许多其他软件的 Linux 很可能造成安装上的问题。

我们在前面曾经提到过，Java Servlet 的最重要功能就是高效率地产生动态网页，因此整个系统会使用到 WEB 服务器、数据库以及一些管理工具。我们所使用的工作环境会比较复杂，将于 Linux 上安装下面的系统软件：

- MySQL 数据库服务器
- Apache WEB 服务器
- PHP
- phpMyAdmin 管理工具
- Java SDK
- Java Servlet SDK

作者在此再一次强调，请您使用本书所提供的软件并依照本章所介绍的方式与顺序来安装系统。

1.3 安装MySQL数据库

我们首先来安装 MySQL 数据库服务器，请您以 root 的身份登入系统，并且将 MySQL-3.22.32-1CLE.i386.rpm 文件由本书所附的光盘拷贝至 /usr/local 目录之下，执行下面的指令：

```
#cd /usr/local
#cp /mnt/cdrom/Chapter01/MySQL-3.22.32-1CLE.i386.rpm .
#rpm -ivh MySQL-3.22.32-1CLE.i386.rpm
```

这时候屏幕上会出现下面的画面, 请您参考图 1-3.1 (/mnt/cdrom 目录是光驱所对应的目录)。

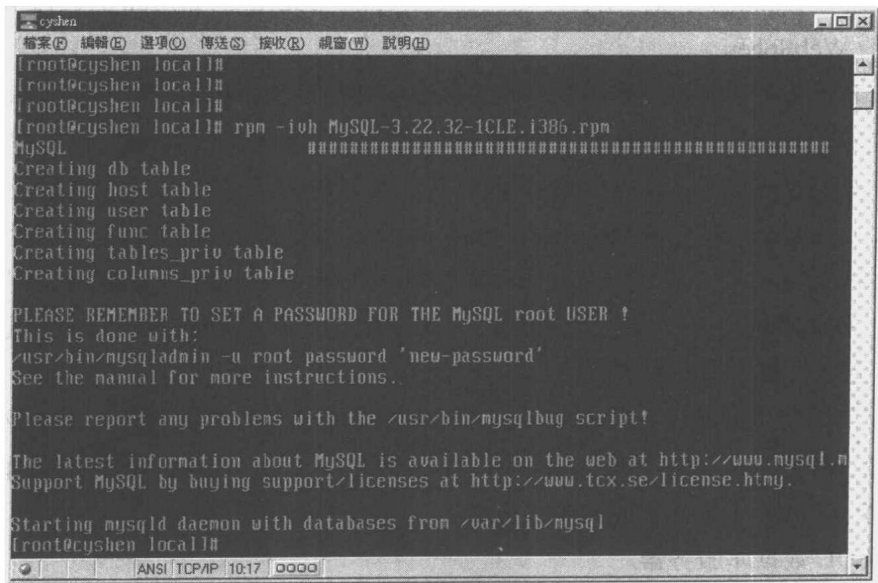


图 1-3.1 安装 MySQL 数据库服务器画面

紧接着我们来安装 MySQL 数据库的用户端软件, 请您将 MySQL-client-3.22.32-1CLE.i386.rpm 由本书所附的光盘拷贝至 /usr/local 目录之下, 请执行下面的指令:

```
#cp /mnt/cdrom/Chapter01/MySQL-client-3.22.32-1CLE.i386.rpm .
#rpm -ivh MySQL-client-3.22.32-1CLE.i386.rpm
```

这时候屏幕上会出现下面的画面, 请您参考图 1-3.2。

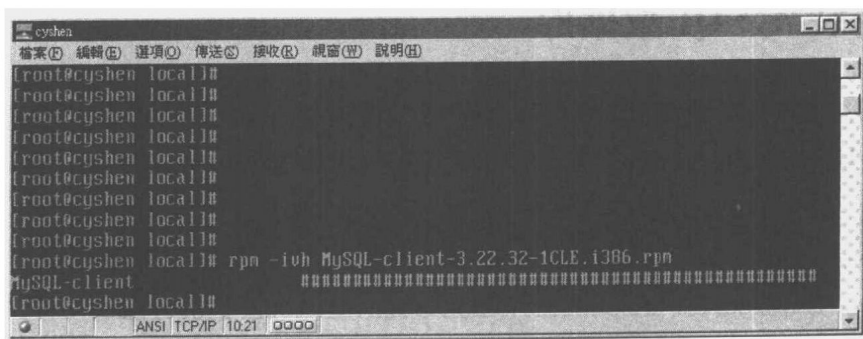


图 1-3.2 安装 MySQL 数据库客户端软件画面

现在我们来为读者介绍如何启动以及关闭 MySQL 数据库服务器，如果您要启动 MySQL 数据库服务器，请执行下面的指令：

```
#/etc/rc.d/init.d/mysql start
```

如果您想要关闭 MySQL 数据库服务器，请执行下面的指令：

```
#/etc/rc.d/init.d/mysql stop
```

当 MySQL 安装好了之后，系统会新增一个 root 用户，其通行密码为 NULL（不需任何的密码）。现在我们来测试 MySQL 服务器是否已经安装成功，请执行 mysql 用户端管理工具以连接上 MySQL 服务器：

```
#mysql
```

如果一切都没有问题，则屏幕上会出现下面的画面，请您参考图 1-3.3。

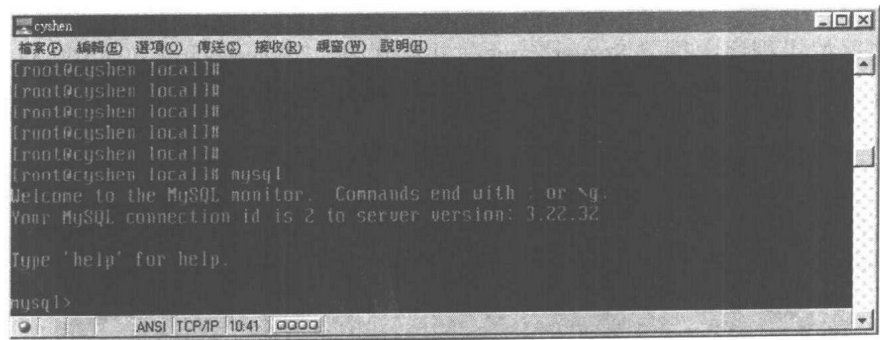


图 1-3.3 以 mysql 用户端管理工具连接上 MySQL 服务器画面

1.4 安装 Apache WEB 服务器

当我们安装好了 MySQL 数据库服务器之后，可以开始安装 Apache WEB 服务器，首先请您将 apache-1.3.9-17mdk.i586.rpm 文件由本书所附的光盘拷贝至 /usr/local 目录之下，执行下面的指令：

```
#cp /mnt/cdrom/Chapter01/apache-1.3.9-17mdk.i586.rpm .  
#rpm -ivh apache-1.3.9-17mdk.i586.rpm
```

这时候屏幕上会出现下面的画面，请您参考图 1-4.1。



图 1-4.1 安装 Apache WEB 服务器画面

当我们安装好了 Apache WEB 服务器之后, 应该要设定它的组态, 实际上 Apache 的组态是非常多的, 但是我们仅设定它的服务器名称, 这样它就可以开始工作了。请您以文字编辑程序开启/etc/httpd/conf 目录下的 httpd.conf 组态文件。

```
#vi /etc/httpd/conf/httpd.conf
```

请于文字编辑程序中搜寻 **ServerName**, 并将其改为"ServerName 202.160.87.72", 您可以将 **ServerName** 变更为您服务器的 IP 地址或是服务器的名称, 请您参考图 1-4.2。

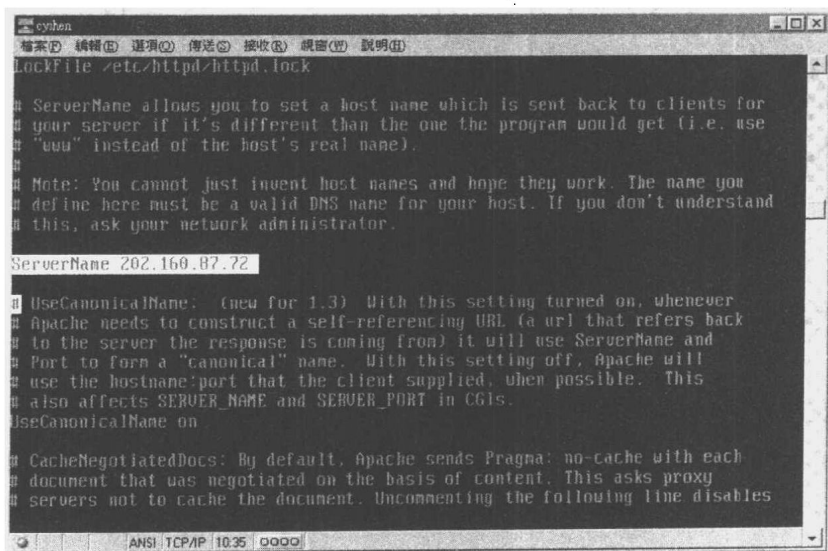


图 1-4.2 设定 Apache 的 ServerName 组态画面

现在我们来为读者介绍如何启动、关闭以及重新启动 Apache WEB 服务器, 如果您要启动 Apache WEB 服务器, 请执行下面的指令:


```
#/etc/rc.d/init.d/httpd start
```

如果您想要关闭 Apache WEB 服务器，请执行下面的指令：

```
#/etc/rc.d/init.d/httpd stop
```

如果您想要重新启动 Apache WEB 服务器，请执行下面的指令：

```
#/etc/rc.d/init.d/httpd restart
```

现在我们来测试 Apache WEB 服务器是否已经安装成功，请以浏览器开启下面的网址：

```
http://202.160.87.72/
```

如果一切都没有问题，则屏幕上会出现下面的画面，请您参考图 1-4.3。

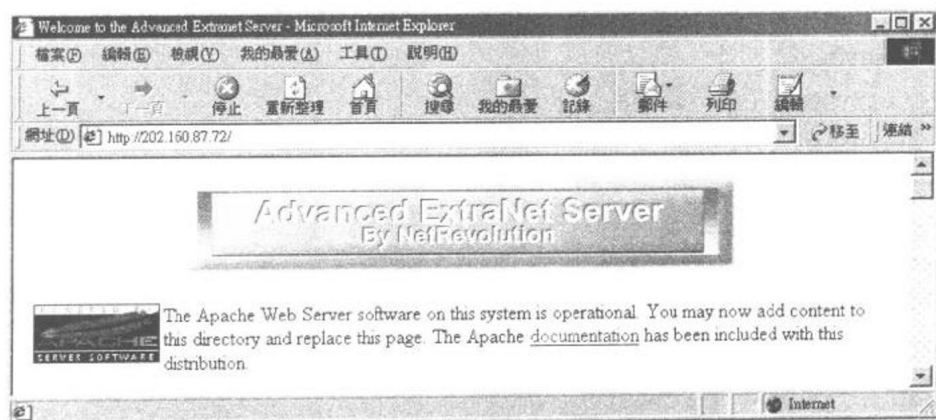


图 1-4.3 以浏览器连接 Apache WEB 服务器画面

1.5 安装PHP

当我们安装好了 Apache WEB 服务器之后，可以开始安装 PHP，首先请您将 mod_php3-3.0.13-6mdk.i586.rpm 文件由本书所附的光盘拷贝至/usr/local 目录之下，执行下面的指令：

```
#cp /mnt/cdrom/Chapter01/mod_php3-3.0.13-6mdk.i586.rpm .  
#rpm -ivh mod_php3-3.0.13-6mdk.i586.rpm
```

这时候屏幕上会出现下面的画面，参考图 1-5.1。