

Oracle

Web 应用 培训教程

Oracle
Web
Applications 101

Oracle Press™ 授权出版

精彩绝伦的
Oracle Web应用,
从这一刻开始

(丹) Sten E. Vesterli 著

蒋蕊 王磊 王巍 等译

266

TP372.492
w136

Oracle 技术系列丛书

Oracle Web 应用培训教程

(丹) Sten E. Vesterli 著

蒋蕊 王磊 王巍 等译

王焱 审校



A0989471



机械工业出版社
China Machine Press

本书讨论 Oracle9i 应用服务器的体系结构, 解释如何使用 5 种不同的方法建立 Web 应用: 使用 Java servlet 和 Java 服务器页面、PL/SQL 和 PL/SQL 服务器页面、Oracle Designer、Oracle9i Application Server Portal 以及 Oracle Forms 和 Reports。

本书内容丰富、由浅入深、条理清晰, 通过本书的学习, 读者将学会何时及如何使用基于 HTML 的应用、基于 Java applet 的应用, 或使用代码生成工具建立强大的 Web 应用。

本书是 Web 应用开发人员的学习指南。

Copyright © 2001 by The McGraw-Hill Companies, Inc.

Original language published by The McGraw-Hill Companies, Inc. All Rights reserved. No part of this publication may be reproduced or distributed in any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Simplified Chinese translation edition jointly published by McGraw-Hill Education (Asia) Co. and China Machine Press.

本书中文简体字版由美国麦格劳-希尔教育出版公司授权机械工业出版社出版, 未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有 McGraw-Hill 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2001-4777

图书在版编目 (CIP) 数据

Oracle Web 应用培训教程 / (丹) 瓦斯特利 (Vesterli, S. E.) 著; 蒋蕊等译. - 北京: 机械工业出版社, 2002.2

(Oracle 技术系列丛书)

书名原文: Oracle Web Applications 101

ISBN 7-111-09753-X

I. O… II. ①瓦… ②蒋… III. 关系数据库 - 数据库管理系统, Oracle - 应用 - 因特网
IV. TP393.492

中国版本图书馆 CIP 数据核字 (2001) 第 097305 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 何秀丽 张鸿斌

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2002 年 2 月第 1 版第 1 次印刷

787mm × 1092mm 1/16·22.25 印张

印数: 0 001 - 5 000 册

定价: 36.00 元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

译者序

首先,非常高兴能够与机械工业出版社的朋友再次合作,翻译出版此书。

21 世纪是一个开放的世纪,在全球商务、科学和军事领域,Web 革命正在悄悄地继续着。Web 应用奇迹般地提高了我们的工作效率、效益与服务,我们看到一批像 Yahoo 这样的信息产业正在崛起。随着交互式 Web 站点的建立,Web 朝着灵活、人性化和互动式方向发展。

今天,我们正处于 Web 生命周期的起始位置——换句话说,现在确实到了参与 Web 应用研发的时间,你等待的时间越长,你进入 Web 世界的难度也就越大。

本书收集了使用 Oracle 工具和服务器建立 Web 应用过程中所收集到的丰富经验,含有丰富的提示和技巧,能够帮助经验丰富的开发者充分利用 Oracle 产品提供的多种特性。同时,书中包括一个循序渐进的说明,手把手地帮助你安装全部服务器和工具,并讲述了建立 Oracle Web 应用的 5 种途径,让你经历整个开发过程,允许你自己找出哪种方法更满足你的需求。

本书面向 Web 应用初学开发者。对于系统设计人员和希望对 Oracle Web 技术概括了解的人来说,本书也很有帮助。

此外,还要说明的是本书翻译过程中,对一些众所周知的英文单词给予了原样保留,例如 applet、form、cookie 等等。

最后,要感谢我们这个充满协作精神的工作小组,正是全体同仁的不懈努力与执著追求,保证了本书保质保量地呈现在广大读者朋友面前。这些朋友是王磊、王焱、王鑫、蒋天仪、刘敏、单翼、杨明、隗伟、陈杰、马静波、周易、王倩、张兵、周同、吴俊、李小亮、薛海潮等等。

虽然我们尽了百分之百的努力,也难免会出现一些不尽人意的地方,我们诚恳地请读者朋友批评指正。

蒋 蕊

2001 年 10 月 20 日于北京

序 言

在 1995 年，当要求我讨论客户/服务器是否已灭亡时，我做了大量的研究。我很快意识到客户/服务器实际上不久就会灭亡。在 1996 年初，我开始建立正式的 Oracle Web 应用。我知道这门技术将会变得十分重要，但是无法想像到 Web 应用仅仅 5 年之后就变得无所不在，我也无法预言 Web 应用对我的生活及身边的一切会产生什么影响。

我们已经走过了很长的一段征途，但是今天，正在建立的每个令人激动的交互式 Web 站点似乎指出了 Web 应用更美好的未来之路：灵活、人性化和互动式。我们正处于 Web 生命周期的起始位置——换句话说，你还什么也看不到。但是现在确实是我们加入的时间，你等待的时间越长，进入 Web 世界的难度也就越大。

现在库存有什么？更好的开发工具和环境（正如我们已经开始看到的，如 Portal、JSP 和 Cherokee）、快捷的访问速度（到家庭的连接速度可达到 T3——传输速率可达 44.736 Mb/s 的通信线路。——译者注）、更高的可靠性和性能（iAS 已经清楚地提供了这些）、软件公司变成服务公司（正如 Oracle 的 Larry Ellison 所允诺的）、改进的商务对商务的通信（通过 XML），等等！

许多“.com”的倒闭确实成为报纸的头版标题，但是在这些感觉论者的故事背后，Web 革命正在悄悄地继续着。大量的组织正在建立并扩展它们的 Web 应用，在工作效率、销售额、效益和客户服务上进行着奇迹般的提高。Web 并不只是简单的 Internet 和无处不在的连接，而是一个功能强大的开发体系。

在我的书里，特别是在最近出版的《Oracle8i Web Development》一书中，我与其他在 TUSC 工作的同事分享了在使用 Oracle 工具和服务器建立 Web 应用过程中所收集到的丰富经验。我的前一本书，《Oracle Application Server Web Toolkit Reference》，对 Web 技术进行了非常好的介绍，但是这本书更好（并且是最新的）。这些非常成功的书籍（以及即将由 Oracle 出版社出版发行的《Oracle9i Web Development》）含有丰富的提示和技巧，能够帮助经验丰富的开发者充分利用 Oracle 产品提供的多种特性。

本书采用了一种不同的方法，专门用于帮助你开始建立 Oracle Web 应用事业。它包括一个循序渐进的说明，介绍如何安装你所需要的全部服务器和工具，并包括了建立 Oracle Web 应用的 5 种途径：使用 Java 或 PL/SQL 手工建立，使用 Oracle Designer、Oracle Portal 和 Oracle Developer。

当我收到 Sten 送来的本书时，易于学习和清晰的写作风格给我留下了深深的印象。我确实喜欢他这种将事情进行类比的写作方式，例如将客户/服务器结构比作有趣的电视装置。我发现他的写作方式阅读起来棒极了。显然，Sten 非常乐于分享这些知识。

要进入 Oracle Web 应用的精彩世界，本书是一条非常棒的途径。我将本书推荐给 TUSC

中的新开发者，同时，也要将它推荐给您。

——Bradley D. Brown

TUSC (The Ultimate Software Consultants, 终点软件顾问公司) 董事长和总设计师;
《Oracle8i Web Development》与《Oracle Application Server Web Toolkit Reference》两书的
作者

前 言

本书是建立 Oracle Web 应用的初学者指南——Web 应用访问一个使用 Oracle 工具建立的 Oracle 数据库。本书解释了在建立 Oracle Web 应用时你所面临的选择，并讲述了建立它们的 5 种基本方式：

- 使用 Java servlet
- 使用手写 PL/SQL
- 使用 Oracle Designer
- 使用 Oracle Portal
- 使用 Oracle Developer: Forms 与 Reports

本书手把手地带领你安装全部必要的软件，并且帮助你使用每种方法让你的第一个 Web 应用得以运行。你将看到如何使用 5 种不同的方法建立相同的 4 个示例应用，让你跟随整个开发过程，并可以找出哪种方法更满足你的需求。

本书所涉及的软件包括：Oracle8i 版本 3 数据库、Oracle9i 应用服务器、Oracle JDeveloper 3.2、Oracle Designer 6i、Oracle9iAS Portal 和 Oracle Developer 6i。

本书适用读者

本书面向首次建立 Web 应用，使用 Oracle 工具连接到 Oracle 数据库的开发者。对于系统设计人员和希望对 Oracle Web 技术概括了解的人来说，本书也很有帮助。你不必知道有关 Web 技术的任何知识，但是你要熟悉 SQL 和 Oracle 数据库的基础知识。

5 种方法中的任何一种方法都包含在一章中，能够独立于其他章节阅读——因此，如果你只想了解 Java servlet 或 Oracle Designer，你可以仅仅阅读相关章节即可。

两个章节说明了“手工”建立 Web 应用的方法：第 6 章介绍 Java servlet，第 7 章介绍 PL/SQL Web 应用。这些章节不可避免地要求读者对所使用的语言有一定的了解，因此第 6 章假定读者具有一定的 Java 知识，第 7 章要求读者具有一些基础的 PL/SQL 知识。

如何阅读本书

第 1 章、第 2 章和第 3 章概要地介绍 Web 应用，比较了 Oracle 提供的可用工具，并解释了它们是如何工作的。

第 4 章含有一个循序渐进的指导，帮助用户在单机上安装进行开发并运行 Oracle Web 应用所需的全部软件——数据库、应用服务器和开发工具。理想情况下，你应该拥有一个功能适中的 PC 机，用于安装软件及进行试验。机器应该安装了 Windows 2000 或 Windows NT，并且至少有 256 兆的内存（虽然你使用 128 兆内存加上一个大的交换文件也可以做到，但是性能却

大打折扣)。安装指南领着你在 Windows/2000 机器上完成安装，但是大部分安装指南也适用于 Linux 和其他 UNIX 类型的机器。

第 5 章讲述了一个通用应用开发方法，适用于全部手写 Web 应用类型，而不用考虑你所选择的是 Java、PL/SQL 还是一个完全不同的编程语言。

第 6 章说明如何使用这个方法实际开发 4 个小型示例应用，如使用 Oracle JDeveloper 开发 Java servlets；第 7 章显示了使用 PL/SQL 开发相同的示例应用。为了便于你在需要时参考当前或其他章节，一些内容在两章中均有出现——毕竟，即使使用其他的编程语言，方法总是一样的。

第 8 章和第 9 章说明如何使用两个 Oracle 代码生成工具：Oracle Designer 和 Oracle Portal。在这两章中，相同的 4 个小型示例应用被再次开发。

第 10 章说明如何使用众所周知的 Oracle 客户/服务器开发工具——Oracle Forms 和 Oracle Reports——来开发相同的应用，并将它们应用到 Web 上。

第 11 章列出了一些工具和应用程序，在开发 Oracle Web 应用时，可能需要用到它们；第 12 章解释了在例子中使用的代码库（本书中的程序代码和全部例子可以从 www.vesterli.com 上下载得到）。

最后，第 13 章为你提供了一些指向书籍和 Web 站点的连接，从中可以继续你的征程，直到变成一个 Oracle Web 应用的高手。

约定

很多代码清单显示了完整的 HTML、PL/SQL 或 Java 代码，以便让你看到它们是如何组合在一起的。以粗体字书写的部分是新增加的材料，在附随的文字说明中解释得更详细：

```
//this is trivial
//but this is explained in detail
//this isn't hard either
```

当某一部分代码已经做过说明时，可能会在后面的程序清单中删略掉。这种情况使用粗斜体字 *[snip]* 指出，如下所示：

```
//we already know all this stuff
[snip]
//here is where it gets interesting again
```

大小写

Java 语言是区分大小写的。所有的 PL/SQL 开发者请跟我说：“JAVA IS CASE SENSITIVE”。如果你从 PL/SQL 转向使用 Java，开始时一定会产生许多编译错误——但是，你能很快掌握大小写的用法。

为了易于转换，并且更加易于比较 Java 和 PL/SQL 代码，许多 PL/SQL 代码使用了 Java 语言中常用的大小写符号，而没有使用 PL/SQL 中正常使用的全部大写符号。因此，在调用

PL/SQL 函数 `START_PAGE` 时，本书可能使用类似 `startPage` 这样的函数名。

从本书获得最大收益

无论你只想学习本书所讲述的 5 种方法中的一种方法，还是希望变成一个全能的专家，我都鼓励你准备一个功能合适的 Windows 2000 或 NT 计算机，便于用来学习这些材料。然后，安装软件，并按照例子所显示的你感兴趣的方法去做。每种方法均通过相同的 4 个小例子加以演示说明，并且一步步的指令会帮助你完成这 4 个小例子以及整个开发过程。如果你遇到问题，可以从 <http://www.vesterli.com> 站点下载大部分完整应用。准备好了吗？让我们开始吧！

本书英文原书书名：Oracle Web Application 101

英文版原书书号：ISBN 0-07-213221-3

英文版原书出版社网址：www.osborn.com

目 录

译者序

序言

前言

第一部分 Web 应用介绍

第 1 章 为什么使用 Web 应用 1

1.1 什么是 Web 应用 1

1.2 Internet 标准 2

1.2.1 TCP/IP 3

1.2.2 HTTP 4

1.2.3 HTML 4

1.2.4 XML 6

1.3 Intranet、Extranet 和 Internet 6

1.4 理想客户机 7

1.4.1 交易 7

1.4.2 重客户机 8

1.4.3 基于浏览器的 Java 应用 8

1.4.4 基于 HTML 的应用 9

1.4.5 可以全部拥有吗 10

1.5 Web 应用优点 10

1.5.1 自由的底层结构 10

1.5.2 自由升级 10

1.5.3 可互换的组件 11

1.5.4 聚集于可用性 11

1.6 小结 12

第 2 章 如何建立 Web 应用 13

2.1 应用必须是 Java 吗 13

2.2 选择你的武器 14

2.2.1 你的用户在哪里 15

2.2.2 你的用户需要什么 16

2.3 建立基于 Applet 的应用 16

2.3.1 使用 Web Forms 17

2.3.2 使用 Java Applet 17

2.4 建立基于 HTML 的应用 18

2.4.1 服务器页面或 Servlet 18

2.4.2 使用 Java 21

2.4.3 使用 PL/SQL 21

2.4.4 使用其他编程语言 22

2.5 生成应用 22

2.5.1 Oracle Portal 22

2.5.2 Oracle Designer 23

2.6 可以打印出它们吗 24

2.6.1 打印问题 24

2.6.2 解决方案 25

2.7 小结 25

第 3 章 Oracle Web 体系结构 26

3.1 Oracle 产品名称 26

3.2 数据库 26

3.3 Web/应用服务器 28

3.3.1 Web 服务器 28

3.3.2 应用服务器 29

3.3.3 Internet 应用服务器版本 30

3.4 开发工具 31

3.4.1 Java 工具 31

3.4.2 PL/SQL 工具 31

3.4.3 Oracle Designer 32

3.4.4 Oracle Portal 32

3.4.5 Oracle Forms 32

3.4.6 Oracle Reports 32

3.5 管理工具 33

3.6 运行 Oracle Web 应用 33

3.6.1 静态内容 33

3.6.2 Java Servlet 33

3.6.3 Java 服务器页面 34

3.6.4 PL/SQL Web 应用 35

3.6.5 PL/SQL 服务器页面 35

3.6.6 Oracle Forms 35

3.6.7 Oracle Reports 37

3.7 小结	38
第4章 开始前的准备工作	39
4.1 如何安装 Oracle 软件	39
4.2 准备工作	40
4.2.1 硬件	40
4.2.2 网络	41
4.2.3 软件	41
4.2.4 文档	42
4.3 数据库	43
4.3.1 准备安装数据库	43
4.3.2 分步安装数据库软件	44
4.3.3 创建数据库	49
4.3.4 检查安装	51
4.4 应用服务器	52
4.4.1 准备安装应用服务器	52
4.4.2 逐步安装 Internet 应用服务器	55
4.4.3 配置应用服务器	64
4.4.4 验证安装	68
4.5 Web PL/SQL 生成器包库	71
4.6 Developer 服务	71
4.6.1 准备安装	72
4.6.2 安装 Forms 服务	74
4.6.3 安装 Reports 服务	75
4.6.4 建立虚拟路径	77
4.6.5 更改环境参数	79
4.7 开发工具	80
4.7.1 JDeveloper	81
4.7.2 PL/SQL 开发工具	82
4.7.3 Oracle Designer 6i 版本 2	82
4.7.4 Oracle Portal	88
4.7.5 Oracle Developer	89
4.8 如果出现故障	90
4.8.1 清除机器	90
4.8.2 磁盘映像	91
4.9 小结	91

第二部分 建立 Web 应用

第5章 设计 HTML 应用	93
5.1 概述	93
5.2 应用设计	94

5.2.1 Web 应用流程	94
5.2.2 创建情节串连图板	100
5.2.3 应用流程图	101
5.2.4 建立框架	103
5.3 页面设计	104
5.3.1 HTML 快速课程	104
5.3.2 JavaScript	119
5.3.3 Web 设计的一些提示	119
5.3.4 选择一个工具	120
5.4 转换	120
5.4.1 将 HTML 转换为 Java	122
5.4.2 将 HTML 转换为 PL/SQL	124
5.4.3 模块化	129
5.5 商业逻辑	129
5.5.1 向服务器页面添加商业逻辑	129
5.5.2 向 Servlets 添加商业逻辑	134
5.6 小结	138
第6章 使用 Java	139
6.1 开始学习	139
6.1.1 编译 Java 代码	139
6.1.2 查找 Java 代码	139
6.1.3 启动 JDeveloper	143
6.2 连接到数据库	145
6.2.1 JDBC 驱动器	145
6.2.2 映射 SQL 数据类型到 Java	146
6.2.3 创建来自 JDeveloper 的连接	146
6.2.4 使用 JDBC	148
6.2.5 使用 SQLJ	151
6.2.6 JDBC 还是 SQLJ	154
6.3 Java Servlets	154
6.4 Hello World	154
6.4.1 建立项目	155
6.4.2 建立 Servlet	156
6.4.3 在 JDeveloper 中察看	158
6.4.4 运行你的第一个 Servlet	159
6.4.5 JDeveloper 为你建立了什么	160
6.4.6 JDeveloper 所建立的剩余部分	162
6.4.7 通过 Web 服务器测试代码	163
6.5 列出全部雇员	164
6.5.1 应用设计	164

6.5.2 页面设计	165	8.1.1 Designer 工具	230
6.5.3 转换	167	8.1.2 使用 Designer	231
6.5.4 商业逻辑	173	8.2 使用 Designer 建立 Web 应用	232
6.6 带有约束条件的清单	175	8.2.1 建立数据库	232
6.6.1 应用设计	175	8.2.2 捕获一个设计	232
6.6.2 页面设计	177	8.2.3 建立应用	235
6.6.3 转换	180	8.3 Hello World	240
6.6.4 商业逻辑	181	8.3.1 创建模块	241
6.7 维护管理雇员	184	8.3.2 生成模块	243
6.7.1 应用设计	185	8.3.3 查看结果	245
6.7.2 页面设计	188	8.3.4 设置 Generator Preferences (生成 器优先项)	246
6.7.3 转换	189	8.4 雇员清单	247
6.7.4 商业逻辑	190	8.4.1 创建模块	247
6.8 小结	196	8.4.2 创建模块组件	248
第 7 章 编写 PL/SQL	197	8.4.3 定义模块组件外观	251
7.1 开始学习	197	8.4.4 生成模块	253
7.1.1 PL/SQL 编程工具	197	8.4.5 改进模块	254
7.1.2 PL/SQL Web 工具箱	197	8.4.6 重新生成模块	256
7.1.3 创建数据库访问描述符	198	8.5 查找雇员	257
7.2 Hello World	201	8.5.1 拷贝模块	257
7.2.1 编写应用	201	8.5.2 编辑模块	257
7.2.2 运行应用	202	8.5.3 编辑模块组件	257
7.3 列出全部雇员	202	8.5.4 生成模块	258
7.3.1 应用设计	202	8.6 维护管理雇员	259
7.3.2 页面设计	203	8.6.1 拷贝模块	259
7.3.3 转换	205	8.6.2 编辑模块	259
7.3.4 商业逻辑	210	8.6.3 生成表应用程序接口	260
7.4 具有约束条件的清单	211	8.6.4 生成模块	261
7.4.1 应用设计	212	8.6.5 选择布局风格	261
7.4.2 页面设计	212	8.7 小结	264
7.4.3 转换	215	第 9 章 Oracle Portal	265
7.4.4 商业逻辑	216	9.1 什么是 Oracle Portal	265
7.5 维护管理雇员	218	9.1.1 建立 Portal	265
7.5.1 应用设计	218	9.1.2 启用数据库的 Web	265
7.5.2 页面设计	221	9.2 开始学习	266
7.5.3 转换	223	9.2.1 Oracle Portal 中的用户	266
7.5.4 商业逻辑	224	9.2.2 启动 Oracle Portal	267
7.6 小结	229	9.2.3 建立事物	268
第 8 章 Designer 应用	230	9.2.4 锁定组件	268
8.1 Designer 方法	230		

9.3 Hello World	269	11.1 常用工具	317
9.3.1 建立应用	269	11.1.1 文本编辑器	318
9.3.2 添加一个应用组件	270	11.1.2 符号链接	318
9.3.3 在 Oracle Portal 以外运行应用	273	11.1.3 磁盘映像	318
9.4 雇员清单	274	11.1.4 磁盘分区	319
9.4.1 控制格式化	274	11.2 PL/SQL 编辑器	319
9.4.2 建立报表组件	277	11.3 转换工具	319
9.5 查找雇员	280	11.3.1 HTML 到 Java	319
9.6 维护管理雇员	284	11.3.2 HTML 到 PL/SQL	320
9.7 小结	288	11.4 HTML 工具	320
第 10 章 Oracle Developer	289	11.4.1 HTML 编辑器	321
10.1 Forms 和 Reports 在 Web 上 如何工作	289	11.4.2 样式表单编辑器	322
10.1.1 Web Forms	289	11.4.3 图像编辑器	322
10.1.2 Web Reports	291	11.5 Javadoc	322
10.2 Hello World Form	292	11.6 PL/SQL Server Pages	324
10.2.1 建立 Form	292	第 12 章 Web 应用工具箱	325
10.2.2 从 Form Builder 运行 Form	295	12.1 HTML 模板	325
10.2.3 在 Web 浏览器中运行 Form	296	12.2 JavaScript	325
10.3 雇员清单报表	298	12.3 Java 工具	326
10.3.1 建立报表	298	12.3.1 示例 Format 类	327
10.3.2 添加格式	301	12.3.2 HtmlForm 类	327
10.3.3 从 Report Builder 运行报表	302	12.3.3 DatabaseUtility 类	330
10.3.4 在 Web 浏览器中运行报表	304	12.4 PL/SQL 工具	331
10.4 带有参数的报表: 查找雇员	306	12.4.1 示例 Format 包	332
10.4.1 建立报表	306	12.4.2 HtmlForm 包	332
10.4.2 运行报表	308	12.4.3 DatabaseUtility 包	335
10.5 维护 Employees 表中的数据	309	第 13 章 有用的书籍和 Web 站点	337
10.5.1 建立 Form	309	13.1 发现你所需要的	337
10.5.2 从 Form Builder 运行 Form	312	13.2 Oracle 技术支持网络	337
10.5.3 在 Web 浏览器中运行 Form	314	13.2.1 软件	337
10.6 小结	315	13.2.2 Oracle 文档	338
第三部分 参 考		13.3 Oracle 技术支持	338
第 11 章 商务工具	317	13.4 书籍	338
		13.5 Web 站点	340

第一部分 Web 应用介绍

第 1 章 为什么使用 Web 应用

想像这样一个世界，在那里每台电视机只能接收一个频道。没有频道选择器，如果想观看其他电视台的节目，你需要另外的电视机。你必须与电视台签约使用某个频道，它们会派一个专业的电视机安装工人到家里安装该频道的专用电视机，当你决定观看其他频道时，其他的专业安装工人也必须来架设另外的电视装置。

非常可笑，是吧？当然，这非常荒唐！

但是考虑一下客户/服务器应用所建立的方式：对于每个应用，人们四处奔走在每个客户工作站安装专用的客户程序；对于每个新应用，IT 专家必须再次来到用户那里安装并设置应用。

Web 应用是基于标准的方法的回应，标准的方法能够让电视很好地工作：如果你希望开始收看一个新的频道，只需要简单地调到这个频道即可。现在，当你想要使用一个新的 Web 应用时，只需简单地在浏览器中输入地址便可访问这个应用。

本章定义 Web 应用并解释了 intranet（企业内部互联网）、extranet（企业外部互联网）和 Internet（国际互联网）之间的区别。接着，本章说明了客户的可能范围，并以编写 Web 应用的优点的讨论结束本章。

1.1 什么是 Web 应用

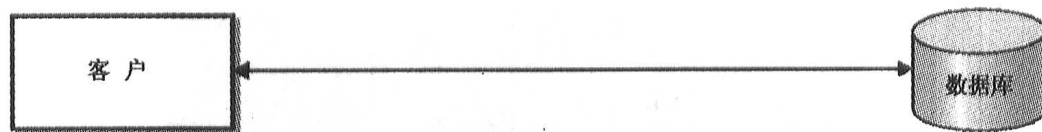
Web 应用是在万维网（World Wide Web，稍后详细说明）上使用的基于公共标准的应用。Internet，特别是万维网，已经取得巨大成功，即使你的应用从不会通过 Internet 让全世界访问，仿效这里所使用的原则仍然非常有意义。

今天，全部多用户应用都是采用某种形式的客户/服务器应用——每个用户在一台客户工作站工作，一些中央服务器存储通用数据。无论使用诸如 Oracle Forms 之类工具开发的“老式”客户/服务器应用，还是新式的 Web 应用，都是实际的客户/服务器应用。

要区分传统客户/服务器应用与 Web 应用之间的差别，我们将使用另一种方法：整个体系结构所具有的层次（或级）的数量。

在传统的客户/服务器应用（使用类似 Oracle Forms 的工具）中，一台客户计算机处理用户界面，一个数据库服务器存储数据。应用的实际功能（商业逻辑）可以驻留在客户端（在 Oracle Forms 中作为触发器）和/或数据库中（作为存储过程）。这是一个两层结构。

回到 1990 年，当在位于瑞士日内瓦的欧洲粒子物理研究所（European Laboratory for Par-



tle Physics) 最初构思出万维网时, 目的在于服务静态文件, 如学术论文。因而, Web 最初具有两层: 仅用于在屏幕上显示文件的浏览器, 以及按照请求分发文件的文件服务器 (或 Web 服务器)。



这简直棒极了; 但经过一段时间后, Web 开发者希望得到更多的功能。他们希望通过运行一个快速创建自定义网页的程序, 获得创建动态内容的能力。因此, Web 服务器功能得以扩展, 能够通过公共网关接口 (Common Gateway Interface, CGI) 调用程序。

这非常好用, 但不久之后, Web 开发者希望这些程序能够访问数据库。这在不同的编程语言中, 有许多不同的方法实现, 但是由于数据库通常驻留在 Web 服务器以外的其他服务器上, 因而现在在客户端有一个 Web 浏览器, 在服务器端有两个或多个服务器层。这导致术语“三层”和“多层”应用的出现。



Web 应用因而被定义为基于 Internet 标准, 使用 Web 浏览器作为客户的多层应用。

1.2 Internet 标准

由于 Web 网页的通信和显示基于许多公共标准, Web 应用解决了许多困扰客户/服务器领域的难题。最为重要的标准在表 1-1 中列出。

表 1-1 用于 Internet 的公共标准

标 准	全 称	应用于
TCP/IP	Transmission Control Protocol/Internet Protocol (传输控制协议/网际协议)	客户与服务器硬件之间的网络通信
HTTP	Hypertext Transport Protocol (超文本传输协议)	Web 服务器与 Web 浏览器之间的通信
HTML	Hypertext Markup Language (超文本链接标记语言)	网页定义
XML	Extensible Markup Language (扩展链接标记语言)	用户专用内容的定义
XHTML	Extensible Hypertext Markup Language (扩展超文本链接标记语言)	网页定义 (HTML 的最新版本)

1.2.1 TCP/IP

最基础的标准是 TCP/IP 协议。该协议定义了计算机之间建立连接的规则，换句话说，规定了客户与服务器硬件之间建立连接的规则。

1. IP 号

在 TCP/IP 网络上的每台计算机必须拥有一个惟一的 IP 号，写作 4 组数字，每组数字介于 0~255 之间，例如：192.168.27.43。

这些数字可以是固定的（换句话说，在计算机配置中定义），也可以是动态分配的。在动态分配号码的情形下，一台计算机被给予大量号码，以便分发给要求号码的其他计算机。请求和得到一个号码的协议被称为动态主机配置协议（Dynamic Host Configuration Protocol, DHCP），分发号码的服务器被称为 DHCP 服务器。

2. 服务器名

IP 号允许计算机发现另外一台计算机并与之通信，但是我们乐于使用名字，而不仅仅是号码，因此，每台计算机还可以拥有一个名字。名字可以写作 3 个或更多个词，使用句点分隔，如图 1-1 所示。

最右边的词（在图 1-1 中为 com）是顶级域（TLD），右边第 2 个词为二级域（SLD）。

顶级域由互联网名称与号码分配组织（Internet Corporation for Assigned Names and Numbers, ICANN）管理，以几种形式存在，如表 1-2 中所示。除此之外，一些新的顶级域也在 ICANN 考虑之中。



图 1-1 服务器名组成要素

表 1-2 顶级域

类 型	域	用 于
通用顶级域	.com	商业组织
	.net	网络组织
	.org	其他组织，包括非盈利组织
专用顶级域	.edu	高等教育
	.gov	美国政府
	.int	国际组织
	.mil	美国国防部
	.uk	（大不列颠）联合王国
国家顶级域	.jp	日本
	.de	德国
	.dk	丹麦
	等等	

在通用的顶级域下，公司和个人可以使用 ICANN 认可的注册机构注册它们自己的二级域（例如，Oracle 公司具有二级域 oracle.com）。你的 Internet 服务提供商可以为你做此项工作。

要在一个特定的顶级域中注册二级域，你的组织必须符合以下的类别之一，并且必须与相关的授权机构取得联系。

国家顶级域由本地注册机构办理。每个国家都有不同的规则——一些国家允许将国家名注册为二级域（如德国，其中 Oracle 为 oracle.de）；其他一些国家使用二级域以进一步分组国家域（如英国，其中 Oracle 为 oracle.co.uk）。另外，一些国家要求你必须是该国的居民或在该国有办公场所，而其他国家并不这样要求。

3. 从名称到号

为了让计算机能够与服务器通信，它需要知道服务器的 IP 号。在一个小型、封闭网络环境中，你可以创建一个含有名称与对应号码清单的文件（称为主机文件），然而，在 Internet 上，这肯定是不可能的。取而代之的是，需要知道其他计算机 IP 号码的计算机将联系一个特殊的信息服务器——域名系统（Domain Name System，DNS）服务器——DNS 知道这些名称和号码。DNS 服务器含有它所负责的服务器的有关信息，并且知道联系哪台服务器得到其他服务器的信息。

对于一个 Web 开发者来说，整个机制是透明的——如果你可以使用名字连接到一个服务器，那么网络设置就是正确的。

1.2.2 HTTP

TCP/IP 协议是两台计算机之间底层通信的标准，而 HTTP 协议是控制 Web 浏览器与 Web 服务器之间通信的高层协议。你可以将它看作整个 Web 地址的前缀（参见图 1-1）。

当你在浏览器中输入一个 Web 地址时，浏览器向服务器发送一条 HTTP 消息，请求 Web 网页。如果能够找到该网页，Web 服务器使用另外一条 HTTP 消息响应，HTTP 消息含有正在返回数据类型的信息，当然还有网页本身。如果无法发现网页或者发生其他错误，Web 服务器返回一条 HTTP 错误消息。

注意，HTTP 错误消息的出现暗示着浏览器的确已经连通 Web 服务器，但是 Web 服务器出于某些原因不能交付你所请求的网页。

1.2.3 HTML

由 Web 服务器返回的实际 Web 网页必须符合超文本标记语言（Hypertext Markup Language，HTML）标准。这个标准定义了所允许的 Web 网页格式。

HTML 基于早期结构化通用标记语言（Structured General Markup Language，SGML）所做的工作，SGML 是一个用于指明文档结构的标准。因此，HTML 最初仅仅打算规范文档的结构，而不是页面的布局，这就解释了为什么最初 HTML 仅仅含有类似 this is a level 2 sub-heading（这里是第二级副标题）的信息，而不含有类似 this is Times Roman 24 point bold（这里是 Times Roman 24 点阵加重字体）的信息。

在客户/服务器环境中，屏幕使用像素精度被仔细地定义，如图 1-2 所示。屏幕定义与全部商业逻辑一起存储在一个专用文件格式的文件中，类似于 Oracle Forms 所使用的 .fmb 文件。