

源于实战、剖析实战、面向实战

来自一线开发作者的经验结晶



掌握 SQL Server 数据库技术的捷径

SQL Server 2005 Web 应用开发

陈永强 张志强 ◎ 主编
段林涛 苏长明 叶安胜 ◎ 副主编



清华大学出版社

TP311.138/546D

2008

SQL Server 2005 Web 应用开发

主 编 陈永强 张志强

副主编 叶安胜 段林涛 苏长明

清华大学出版社

北 京

内 容 简 介

本书讲解了利用 SQL Server 2005 进行 Web 数据库系统开发需要使用的相关技术, 以及在 Web 数据库应用系统中利用 ADO.NET 2.0 对数据库进行访问的各种方法。全书共有 14 章。第 1 章主要介绍 Web 数据库系统的基础知识; 第 2 章主要对 SQL Server 2005 进行简介; 第 3 章主要讲解了 SQL Server 2005 的数据库与数据表的操作; 第 4 章讲解了 SQL Server 2005 数据操纵技术的实现; 第 5 章讲解了 SQL Server 2005 视图的使用; 第 6 章讲解了 SQL Server 2005 存储过程的使用; 第 7 章讲解了 SQL Server 2005 触发器的使用; 第 8 章讲解了 Web 编程的基础知识; 第 9 章简介了 ADO.NET 2.0 的体系结构; 第 10 章讲解了利用 ADO.NET 2.0 编程连接 SQL Server 2005 数据库方法的实现; 第 11 章讲解了 ADO.NET 2.0 操作的基础知识; 第 12 章讲解了利用 Visual Studio 2005 中的 DataSource 控件完成对数据库的操作; 第 13 章主要通过大量实例讲解了利用 ADO.NET 2.0 编程完成对数据库的访问; 第 14 章以“高校基金项目申报系统”为例阐述了一个实际的 Web 应用系统的分析与开发过程。本书配套光盘中提供了所有的源代码。

本书可作为各类大专院校、各种成人教育学校和培训班数据库类课程的教材, 也可作为 Web 程序开发人员的参考书籍。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

SQL Sever 2005 Web 应用开发/陈永强, 张志强主编; 叶安胜, 段林涛, 苏长明副主编. —北京: 清华大学出版社, 2008.2

ISBN 978-7-302-16803-4

I. S… II. ①陈… ②张… ③叶… ④段… ⑤苏… III. ①关系数据库—数据库管理系统, SQL Sever 2005②主页制作—程序设计 IV. TP311.138 TP393.092

中国版本图书馆 CIP 数据核字(2008)第 004802 号

责任编辑: 彭 欣 桑任松

封面设计: 柏拉图+创意机构

版式设计: 杨玉兰

责任校对: 李玉萍

责任印制: 孟凡玉

出版发行: 清华大学出版社 地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编: 100084

c-service@tup.tsinghua.edu.cn

社 总 机: 010-62770175 邮购热线: 010-62786544

投稿咨询: 010-62772015 客户服务: 010-62776969

印 刷 者: 北京密云胶印厂

装 订 者: 三河市溧源装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 26.5 字 数: 638 千字

版 次: 2008 年 2 月第 1 版 印 次: 2008 年 2 月第 1 次印刷

印 数: 1~4000

定 价: 45.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 024162-01

前 言

SQL Server 2005 是一种功能强大的数据库管理系统, Visual Studio 2005 是 Microsoft 推出的基于 .NET Framework 2.0 架构运行的应用软件的开发工具。后台数据库采用 SQL Server 2005, 利用 Visual Studio 2005 的 ADO.NET 2.0 数据库访问技术开发 Web 数据库应用系统的模式目前正得到广泛应用。

本书特色

本书由浅入深, 从实用的角度出发, 结合大量的实例, 操作性强, 并且描述了实例的具体设计步骤并提供详细代码, 特别在第 14 章完整阐述了一个 Web 数据库应用系统的分析和开发全过程, 读者可以掌握利用 .NET 技术开发 Web 数据库应用系统的方法。

本书通俗易懂, 图文并茂, 实用性强, 适应读者边看书边操作的需要, 便于读者自学。本书可作为大专院校数据库类课程的教材, 也可作为计算机技术培训班教材, 也适合愿意学习利用 .NET 技术完成 Web 数据库应用系统开发的读者, 同时可作为各类管理人员和计算机爱好者的自学参考书。

本书内容

本书共分 14 章, 具体内容如下。

第 1 章主要介绍了 Web 数据库应用系统的基础知识。

第 2 章至第 7 章主要讲解了编程访问 SQL Server 2005 的各种技术, 包括数据库的操作、数据操纵技术的实现、视图的使用、存储过程的使用、触发器的使用。

第 8 章主要讲解了 Web 系统开发的相关基础知识, 包括 Web 服务器的配置、Visual Studio 2005 环境的安装、Visual Studio 2005 工具的使用。

第 9、10、11、13 章主要讲解了 ADO.NET 2.0 数据库访问技术的使用, 包括编程完成数据库的连接, 数据库的读、写等操作。

第 12 章主要讲解了 Visual Studio 2005 提供的新控件 DataSource 的使用, 包括利用该控件直接完成数据库的读、写操作。

第 14 章主要通过一个实例讲解了利用 .NET 技术完成 Web 数据库应用系统的开发全过程。包括数据库设计、功能模块的设计、页面的设计、详细代码的设计、代码优化的设计等。

本书导读

对于开发 Web 应用系统的初学者应该从本书前面章节开始学习 SQL Server 2005 的使用。通过前面章节的学习以了解数据库的基本知识和 SQL Server 2005 的使用方法。掌握 SQL Server 2005 数据管理的具体使用方法后, 再学习 Visual Studio 2005 开发工具的使用, 最后学习 ADO.NET 数据库访问技术的使用。

对于具有一定数据库基础,了解 SQL Server 2005 和 Visual Studio 2005 的读者,也可以直接学习后面具体编程的部分。

对于熟悉 SQL Server 2005 和 Visual Studio 2005 的读者,可将实例编程的章节作为学习重点,着重掌握 SQL Server 2005 和 Visual Studio 2005 开发 Web 数据库应用系统的过程。

我们的理念

根据我们的教学和开发经验,学习计算机软件技术的关键是让学习者迅速找到适合自己学习的方法,必须在第一时间使学习者看到自己的学习成绩,排除“畏惧编程”的心理因素,有必要走“依葫芦画瓢”的路,然后逐渐由浅入深,逐步进入软件开发领域。

因此本书采用大量的实例并结合具体操作进行编写,书中包含有丰富的操作内容,我们希望读者通过大量的实际操作加深对相关内容的认识和理解,尽快把理论知识转换为解决实际问题的能力。另一方面,为方便读者快速阅读本书,书中的大量实例在配套光盘中都给出了完整的源代码,读者可以根据自己的实际情况进行选择使用。

本书由西华大学陈永强和成都大学张志强担任主编,段林涛、苏长明、叶安胜担任副主编,同时参与编写工作的其他老师都是成都大学信息科学与技术学院的教师。编著者都长期从事计算机软件方面的教学工作,一直从事数据库及应用系统的开发工作,有着丰富的实践经验和教学科研经验。希望本书的出版,对广大数据库爱好者,特别是想学习 Web 数据库应用系统开发的设计和管理者,起到一定的帮助作用。

本书第 8 章由陈永强编写,第 1、9、14 章由张志强编写,第 10 章由段林涛编写,第 4 章由苏长明编写,第 11、12、13 章由叶安胜编写,第 2、3 章由黄爱民编写,第 5、6 章由李平勇编写,第 7 章由刘永红编写,成都理工大学信息工程学院硕士研究生周晓清完成了第 12 章部分内容编写及实例程序调试。最后由陈永强和张志强统稿完成。张志强、叶安胜、段林涛、苏长明和成都大学信息科学与技术学院学生胡明华完成了应用实例中程序的编写和修改工作。

在此感谢所有参与编写和提供帮助的同事和学生。

限于自身水平,我们全体参编人员虽然已尽心尽力,但由于时间仓促,书中难免出现遗漏和错误之处,希望广大读者指正。

作者

目 录

第 1 章 Web 数据库应用系统的概述.....1

1.1 数据源.....2

1.2 Web 页面概述.....2

1.2.1 Web 的工作方式.....2

1.2.2 Web 页面的特点.....3

1.2.3 Web 页面的功能.....3

1.2.4 Web 页的分类.....4

1.2.5 动态页面技术概述.....5

1.3 Web 数据库系统的实现技术.....6

1.3.1 Web 与数据库技术的结合.....6

1.3.2 Web 数据库系统的体系结构.....6

1.3.3 常用数据库访问技术.....8

1.4 Microsoft.NET 概述.....13

1.4.1 Microsoft.NET 战略.....13

1.4.2 Microsoft.NET 新特性.....13

1.4.3 Microsoft.NET 平台.....15

1.4.4 Microsoft.NET 框架 (.NET Framework).....15

本章小结.....20

第 2 章 SQL Server 2005 数据库

简介.....21

2.1 SQL Server 2005 数据平台.....22

2.2 SQL Server 2005 的新特性与功能.....23

2.3 SQL Server 2005 的各种版本.....27

2.4 SQL Server 2005 管理及 开发工具.....28

2.4.1 SQL Server 2005 联机丛书.....29

2.4.2 SQL Server Management Studio.....29

2.4.3 SQL Server Configuration Manager.....34

2.4.4 SQL Server Surface Area Configuration.....35

2.4.5 Database Engine Tuning Advisor.....36

2.4.6 SQL Server Profiler.....37

2.4.7 SqlCmd 命令行工具程序.....39

本章小结.....42

第 3 章 SQL Server 2005 数据库与表 的实现.....43

3.1 SQL Server 2005 数据库基本知识.....44

3.1.1 数据库文件.....44

3.1.2 文件组.....44

3.1.3 数据库状态.....45

3.1.4 数据库文件状态.....45

3.2 系统数据库.....46

3.2.1 master 数据库.....46

3.2.2 model 数据库.....47

3.2.3 msdb 数据库.....47

3.2.4 resource 数据库.....47

3.2.5 tempdb 数据库.....48

3.3 数据库索引.....49

3.3.1 索引的概念.....49

3.3.2 索引类型.....49

3.3.3 设计索引.....50

3.4 实现数据库.....51

3.4.1 创建数据库前的准备工作.....51

3.4.2 创建数据库.....52

3.4.3 修改数据库.....59

3.4.4 删除数据库.....65

3.5 实现数据表.....67

3.5.1 数据类型.....67

3.5.2 约束.....68

3.5.3 创建表.....	70	4.9 数据添加.....	120
3.5.4 修改表.....	83	4.9.1 INSERT 语句.....	120
3.5.5 删除表.....	86	4.9.2 插入一行数据.....	120
本章小结.....	87	4.9.3 插入多行数据.....	121
第 4 章 数据操纵技术	89	4.10 数据更新.....	122
4.1 SELECT 语句的基本语法.....	90	4.10.1 UPDATE 语句.....	122
4.2 基本查询.....	91	4.10.2 使用简单的 UPDATE 语句.....	123
4.2.1 查询全部列信息.....	91	4.10.3 带 WHERE 子句使用的 UPDATE 语句.....	123
4.2.2 查询部分列信息.....	92	4.10.4 将 UPDATE 语句用于来 自其他表的信息.....	123
4.2.3 输出别名修饰.....	92	4.10.5 使用包含 TOP 子句的 UPDATE 语句.....	124
4.2.4 DISTINCT 关键字的使用.....	93	4.10.6 修改多列数据.....	124
4.2.5 输出计算列.....	93	4.11 数据删除.....	125
4.2.6 ORDER BY 子句的使用.....	93	4.12 SQL Server 2005 数据操纵语句 新增功能.....	125
4.2.7 TOP 关键字的使用.....	95	4.12.1 WITH 公用表表达式.....	125
4.3 条件查询.....	95	4.12.2 TOP 表达式.....	126
4.3.1 使用关系表达式查询.....	95	4.12.3 OUTPUT 子句.....	127
4.3.2 使用逻辑表达式查询.....	96	4.12.4 WRITE 子句.....	129
4.3.3 设置取值范围的查询.....	97	本章小结.....	130
4.3.4 空值查询.....	98	第 5 章 视图	131
4.3.5 模糊查询.....	99	5.1 视图概述.....	132
4.4 分组查询.....	101	5.1.1 认识视图.....	132
4.4.1 GROUP BY 子句的使用.....	101	5.1.2 视图的典型应用.....	133
4.4.2 聚合函数的使用.....	101	5.1.3 视图的优缺点.....	135
4.4.3 HAVING 子句的使用.....	105	5.1.4 视图类型.....	136
4.5 连接查询.....	106	5.2 创建视图.....	136
4.5.1 常规连接方式.....	106	5.2.1 使用 SQL Server Management Studio 创建视图.....	138
4.5.2 使用 JOIN-ON 连接方式.....	108	5.2.2 使用 Transact-SQL 语句 创建视图.....	140
4.5.3 自连接方式.....	110	5.3 查询视图.....	143
4.6 子查询.....	111	5.3.1 查询由视图定义的数据.....	143
4.6.1 子查询概念.....	111		
4.6.2 无关子查询.....	112		
4.6.3 相关子查询.....	114		
4.7 集合操作.....	115		
4.7.1 集合合并.....	115		
4.7.2 半连接操作.....	117		
4.8 保存查询结果.....	119		

5.3.2 获取视图信息.....	144	本章小结.....	185
5.4 修改视图.....	147	第 7 章 触发器	187
5.4.1 修改视图的两种方法.....	147	7.1 触发器概述.....	188
5.4.2 重命名视图.....	150	7.1.1 DML 触发器.....	188
5.5 删除视图.....	151	7.1.2 DDL 触发器.....	188
5.6 通过视图修改基本表数据.....	152	7.2 DML 触发器的使用.....	189
5.6.1 通过视图向基本表中 插入数据.....	153	7.2.1 DML 触发器的设计.....	189
5.6.2 通过视图删除基本表中 的数据.....	155	7.2.2 插入的表和删除的表.....	189
5.6.3 通过视图更新基本表中 的数据.....	157	7.2.3 使用 SQL Server 管理平台 创建 DML 触发器.....	190
5.6.4 通过视图修改数据的限制.....	158	7.2.4 使用 CREATE TRIGGER 命令 创建 DML 触发器.....	191
本章小结.....	159	7.2.5 DML 触发器的应用示例.....	195
第 6 章 存储过程	161	7.3 DDL 触发器的使用.....	197
6.1 存储过程概述.....	162	7.3.1 DDL 触发器的设计.....	197
6.1.1 存储过程的定义.....	162	7.3.2 使用 CREATE TRIGGER 命令 创建 DDL 触发器.....	197
6.1.2 存储过程的优点.....	162	7.3.3 DDL 触发器的应用示例.....	201
6.1.3 存储过程的类型.....	163	7.4 触发器的维护.....	202
6.2 创建存储过程.....	164	7.4.1 触发器的修改.....	202
6.2.1 设计存储过程.....	164	7.4.2 触发器的禁用和删除.....	204
6.2.2 创建存储过程.....	165	本章小结.....	206
6.2.3 执行存储过程.....	167	第 8 章 Web 应用开发基础	207
6.2.4 创建存储过程实例.....	168	8.1 初识 ASP.NET 2.0.....	208
6.3 重新编译存储过程.....	175	8.1.1 ASP.NET 的特点.....	208
6.3.1 使用 WITH RECOMPILE 选项.....	176	8.1.2 ASP.NET 与 ASP.....	209
6.3.2 EXECUTE 语句中的 RECOMPILE 选项.....	176	8.1.3 ASP.NET 运行模型.....	209
6.3.3 使用 sp_recompile 系统 存储过程.....	177	8.2 Web 服务器的安装与配置.....	210
6.4 维护存储过程.....	177	8.2.1 安装 IIS.....	210
6.4.1 查看存储过程.....	177	8.2.2 配置 IIS.....	212
6.4.2 修改存储过程.....	179	8.2.3 创建虚拟目录.....	214
6.4.3 加密存储过程.....	182	8.3 ASP.NET 2.0 运行环境的安装.....	217
6.4.4 删除存储过程.....	183	8.3.1 Visual Studio 2005 的 安装需求.....	217
		8.3.2 Visual Studio 2005 的安装.....	218

8.3.3	检测 .NET Framework 2.0.....	220	本章小结	249
8.3.4	Visual Studio 2005 常见问题解决.....	221	第 10 章 SQL Server 数据库的连接	251
8.4	Visual Studio 2005 IDE 简介	222	10.1 .NET 数据提供程序	252
8.4.1	Visual Studio 2005 的起始页	222	10.1.1 SQL Server .NET Framework 数据提供程序.....	252
8.4.2	新建项目和打开项目对话框.....	224	10.1.2 OLE DB .NET Framework 数据提供程序.....	253
8.4.3	客户区	224	10.1.3 ODBC .NET Framework 数据提供程序.....	254
8.4.4	控制浮动面板	225	10.1.4 选择 .NET Framework 数据提供程序.....	254
8.4.5	解决方案资源管理器	225	10.2 Connection 对象(连接对象)	255
8.4.6	属性窗口	226	10.3 Connection 对象的属性	255
8.4.7	工具箱	226	10.3.1 ConnectionString 属性	256
8.5	Web 应用系统	226	10.3.2 ConnectionString 属性的设置与读取	257
8.5.1	Web 应用系统的创建	227	10.4 创建 Connection 对象	259
8.5.2	向 Web 应用系统中添加文件	228	10.5 Connection 对象的方法	260
8.5.3	Web 应用系统的发布	229	10.5.1 打开和关闭连接	260
8.6	Web 窗体	230	10.5.2 连接池	260
8.6.1	Web 窗体概述	230	10.6 Connection 对象应用	261
8.6.2	Web 窗体的创建	231	本章小结	262
8.6.3	Web 窗体文件的构成与关联	234	第 11 章 ADO.NET 操作基础知识	263
8.7	Web 应用系统配置	238	11.1 Command 对象(命令对象)	264
8.7.1	Application 对象	238	11.1.1 Command 对象属性	264
8.7.2	Global.asax 文件的创建与配置	239	11.1.2 创建 Command 对象	265
8.7.3	Web.config 文件的创建与配置	240	11.1.3 Command 的参数集合	266
8.7.4	添加引用的实现	243	11.1.4 Command 的方法	269
本章小结	243	11.2 DataReader 对象(数据阅读器对象)	273	
第 9 章 ADO.NET 简介	245	11.3 DataAdapter 对象(数据适配器对象)	275	
9.1	ADO.NET 概述	246	11.3.1 DataAdapter 对象的属性	276
9.2	ADO.NET 组件对象模型	247	11.3.2 DataAdapter 对象的创建	278
9.2.1	.NET 数据提供程序	247	11.3.3 DataAdapter 对象的方法	279
9.2.2	断开连接对象(DataSet)	248		

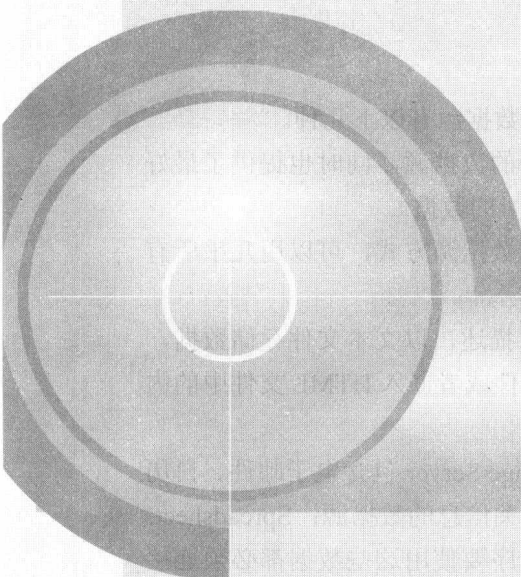
11.4 DataSet 对象	281	12.6 DataSource 控件对存储过程 的访问	326
11.4.1 DataSet 对象的属性 和方法	282	本章小结	329
11.4.2 DataTable 对象 (数据表对象)	283	第 13 章 利用 ADO.NET 对 SQL Server 数据库的操作	331
11.4.3 如何访问 DataSet 对象 中表的行和列	291	13.1 使用 SqlDataReader 数据源控件 完成数据库的读操作	332
11.4.4 DataView 对象 (数据视图对象)	292	13.1.1 不带参数的简单 绑定范例	332
11.5 事务	293	13.1.2 带参数的简单绑定范例	333
11.5.1 什么是事务	293	13.1.3 带参数的存储过程 的读操作	333
11.5.2 事务编程流程	293	13.2 使用 SqlDataReader 完成数据库 的读操作	333
11.5.3 事务对象编程	294	13.2.1 不带参数的简单绑定 范例	333
本章小结	296	13.2.2 带参数的简单绑定范例	335
第 12 章 利用 DataSource 控件对 SQL Server 数据库的操作	297	13.2.3 带参数的代码读取范例	337
12.1 DataSource 数据源控件简介	298	13.3 使用 SqlDataAdapter 对象完成 数据库的读操作	341
12.1.1 概述	298	13.3.1 DataSet 作为绑定数据源	341
12.1.2 DataSource 控件架构	299	13.3.2 带参数的数据读取	342
12.2 SqlDataReader 数据源控件 主要属性	299	13.4 使用 SqlCommand 对象完成 数据库的写操作	344
12.3 SqlDataReader 的控件 参数类型	306	13.4.1 不带参数的数据库 写操作	344
12.3.1 ControlParameter 参数类型	307	13.4.2 带参数的数据库写操作	347
12.3.2 CookieParameter 参数类型	310	13.4.3 使用存储过程完成 数据库写操作	349
12.3.3 FormParameter 参数类型	312	本章小结	352
12.3.4 QueryStringParameter 参数类型	314	第 14 章 综合实例	353
12.3.5 SessionParameter 参数类型	316	14.1 系统总体设计	354
12.4 参数方向	319	14.1.1 系统模型的建立	354
12.5 DataSource 控件使用 数据库事务	320	14.1.2 系统功能描述	354
		14.1.3 系统功能模块设计	355

14.2 数据库设计.....	356
14.2.1 数据表的设计.....	357
14.2.2 数据表之间的联系.....	363
14.3 系统性能优化措施.....	365
14.3.1 封装逻辑业务层功能 的优点.....	365
14.3.2 封装的实现.....	365
14.3.3 数据库连接配置设计.....	371

14.4 系统的整体页面风格设计.....	371
14.5 登录模块的设计.....	374
14.6 权限管理模块的设计.....	375
14.7 机构维护模块的设计.....	382
14.8 基础数据维护模块的设计.....	390
14.9 校基金项目模块的设计.....	394
本章小结.....	408

参考文献.....	409
-----------	-----

第12章 利用DataSource控件对 SQL Server数据库的操作.....	297
12.1 DataSource数据库控件简介.....	298
12.1.1 概述.....	298
12.1.2 DataSource控件架构.....	299
12.2 SqlDataAdapter数据库控件.....	299
12.2.1 主要属性.....	299
12.2.2 SqlDataAdapter的控件 参数类型.....	306
12.2.1.1 CommandParameter.....	307
12.2.2.1 CookieParameter.....	310
12.2.3.1 FormParameter.....	312
12.2.4.1 QueryStringParameter.....	314
12.2.5.1 SessionParameter.....	316
12.2.2 参数类型.....	319
12.2.3 参数方向.....	319
12.2.4 DataSource控件使用.....	320
12.2.5 数据库事务.....	320



第 1 章

Web 数据库应用系统 的概述

本章要点:

本章主要介绍了 Web 数据库系统相关的基础知识,阐述了 Web 数据库系统实现的原理和当前主要的实现方法,其中包括 Web 静态页面、Web 动态页、Web 与数据库结合技术的介绍等,重点阐述了 .NET 的体系架构和相关技术。通过该章的讲述,使读者对 Web 数据库应用系统有了一个初步的认识,为本书后面章节的应用程序开发打下了基础。

1.1 数 据 源

Web 数据库系统的应用程序涉及大量的数据源，大致数据源有以下几种。

(1) 数据库。通常提供最为可靠、伸缩性最大和最安全的数据源，同时也提供了最好的性能，它用于存储大量数据并可以任意的顺序检索任意数量的数据。

(2) 文本文件。这是对于存储数据来说相对节省空间的数据源方式，可以用几乎所有的格式存储信息，通常用于存储一组数据。

(3) XML。它是使用分层的关联格式，而且可扩展和自描述，以文本文件存储数据，利用 XML 数据源对象 XML-DSO 可以将一个外部的 XML 文件或者嵌入 HTML 文件中的内容提取到 HTML 页面中。

(4) 其他适合于特定领域的数据库源。如 Microsoft Exchange Server 包含电子邮件、日历和联系方式的数据源；Active Directory 存储网络上对象的相关信息的数据源；Spreadsheets 以表形式存储的数据源。不管数据源是哪一种，每个应用程序要使用这些数据都必须连接到这些数据源上，读取和转换，显示这些数据。

1.2 Web 页面概述

WWW 是一个基于 Internet 的、全球连接的、分布的、动态的、多平台的交互式图形超文本信息系统，它利用多种协议去传输和显示驻留在世界各地计算机上的多媒体信息源。同时 WWW 又极大地推动了 Internet 的发展，成为 Internet 最流行、最主要的信息服务方式，而 WWW 普遍采用 Web 页进行信息存储与传递，它提供了在 Internet 上的一种十分有效的浏览、检索及查询信息的方式，这些信息有多种类型，包括文字、图形、图像、声音和动画等。

1.2.1 Web 的工作方式

Web 的信息保存在 Web 站点中，用户通过 Web 浏览器来访问。体系架构上它是一种 B/S(Browser/Server, 浏览器/服务器)模式。用户使用浏览器从网上查阅 Web 信息，把需要的信息从网上下载到本机。由于信息的分布地方不同，用户需求信息不同，表现在 Web 上的是一种链接地址的不断变化。

Web 浏览器的主要功能是解释并显示由 Web 服务器传送来的，由 HTML 写成的文档页面，包括嵌入在 HTML 文档中的多媒体信息(如图像等)。因此浏览器中还可以根据用户的需要配置相应的辅助应用程序，用来处理嵌入在 HTML 文档中的声音、视频等多媒体信息。我们将显示在浏览器中的 HTML 文本称为 Web 页面。

整个 Web 系统的实现过程如下。

(1) 用户在浏览器的地址栏中输入一个 URL(Uniform Resource Locator, 统一资源定位

器),即通常的网址,随后浏览器向 Web 服务器发出连接请求。

(2) Web 服务器接到浏览器的请求后,将 URL 转换成对应的服务器上的文件路径名。(在 Web 服务器中有一个监听程序,随时监听网络中是否有连接的请求)

(3) 若 URL 指向的是 HTML 文档,到 Web 服务器直接传送到浏览器中解释执行 HTML 代码并显示相应的结果。如果 HTML 代码中有用 Java、JavaScript、ActiveX、VBScript 等编写的脚本代码时(它们一般是嵌入在 HTML 代码中的),Web 服务器也一并传送到浏览器中执行。

1.2.2 Web 页面的特点

一般 Web 页具有以下几个特点。

(1) Web 页的可读性强。Web 是一种超文本信息系统,利用超链接可以使用户快速地获得不同的信息,用户通过点击相关的关键词的超链接,可以迅速获得相关信息资料,提高阅读效率。

(2) Web 页具有导航能力。Web 页可以将图形、声音、视频信息集为一体,这都归结于它的导航能力。一个页面具有同时显示图像、声音、视频、Flash 动画等超媒体信息的能力。网站与网站之间、网页与网页之间的超链接,提供了多样化的用户浏览方式,以及迅速快捷的信息下载。

(3) Web 页面与系统平台无关。用户可以在不同的硬件平台和操作系统平台上访问 Web 页。

(4) Web 信息的分布特点。基于 Internet 的信息分布广的特点,一个网站所能容纳的信息有限,通过站点的信息交互驱动,使得物理位置不同的信息在逻辑上联系起来,对用户来说,只需利用超链接就能够方便地进行不同站点的访问。

(5) Web 页面的动态交互性。利用动态交互技术(ASP、ASP.NET、Java 等)克服了静态 Web 页面的缺点,使 Web 页具有动态性、交互性,这样用户能与服务器进行交互操作,从而使用户从单纯浏览信息过渡到 Web 应用程序的使用。

(6) Web 页面的信息更新性。Web 页面信息是可以动态变化的,信息的提供者可以很方便地对站点上的信息进行更新,用户可以得到实时的新信息。

1.2.3 Web 页面的功能

Web 页面实现的功能主要体现在以下几个方面。

1. 超链接功能

通过超链接功能,利用导航机制可以使用户进入不同的网站或页面,并通过链接实现在网页间和网站间的相互转换。链接标识是 Web 页面非常重要的功能,也是页面的基本元素之一。

2. 页面的交互性

交互性是网站响应用户动作和选择的方式。交互式的网站会吸引更多用户参与、允许用户选择自己想看的内容,而不是静态地显示信息。交互类型有交互式导航、交互式多媒体、交互式广告等。例如用户间的交互包括论坛、聊天、视频会议等。Web 页面包括的公告栏(又叫论坛或新闻组),用户可以公开发布信息,而其他用户可以阅读这些信息并进行回复。在聊天室里用户间可以实时交换信息。视频会议是一种使用日益广泛的交互式功能。随着宽带网络技术的发展,视频会议技术日趋成熟,有效地实现了用户间的交互功能。

3. 信息发布功能

网站通过 Web 主页实现信息发布功能,它反映了整个网站发布的信息内容,提供了站点主要部分的链接。在主页中通过超链接,用户可以根据需要浏览相关内容。

1.2.4 Web 页的分类

根据 Web 页技术的发展,Web 页分为:静态 Web 页面和动态 Web 页面。

1. 静态 Web 页面

静态页面就是其内容是由一些 HTML 代码组成的页面,且这些代码可以直接输入到文本编辑器之中,并保存为.htm 或.html 文件,在访问之前,完全确定了页面的内容。

静态 Web 页面具有局限性,不具备个性化的 Web 页面功能,即不具备动态特征;另外没有任何的安全措施。

2. 动态 Web 页面

目前创建动态 Web 页面,大致有两种不同的模型。

1) 客户端动态 Web 页面

在客户端模型中,在浏览器端运行的模块(如插件)完成创建动态页面的全部工作。通常 HTML 代码中包含了存储一组指令的单独文件,该文件在 HTML 页面中引用,其传送到浏览器上由浏览器执行这些指令;另外这些指令也可直接与 HTML 代码混合在一起。当用户请求 Web 页面时,浏览器利用这些指令为页面生成纯 HTML。

这种技术也有缺陷。首先需要长时间的下载,特别是当需要下载多个文件时;其次每一个浏览器以不同的方式解释客户端脚本代码,无法保证所有的浏览器都能理解;再者是当编写使用服务器端资源(如数据库)的客户端代码时,代码在客户端被解释,非常不安全。

2) 服务器端动态 Web 页面

利用服务器端模型,HTML 源代码与混合在其中的一套指令被传回到 Web 服务器。当用户请求页面时,这套指令用于给页面生成 HTML,页面根据请求动态生成。

1.2.5 动态页面技术概述

1. 提供动态内容的客户端技术

这类技术都依赖于内置在浏览器中的模块(即插件)来处理指令,客户端技术就是脚本语言、控件以及功能完善的编程语言的综合。这类技术包括如下。

1) JavaScript

JavaScript 是最早的浏览器脚本语言,用于给其 Web 服务器和浏览器增加交互功能。所有的浏览器都支持,语法借用了 Java 的一些语法和基本结构。

2) VBScript

VBScript 是 Microsoft 推出的脚本语言,基于 Visual Basic,语法简单易学,但运行速度慢,效率低,只有微软浏览器支持。

3) Java 小应用程序

Java 是用于开发应用程序的跨平台语言,这些小应用程序借助于<applet>标记插入到 Web 页面中形成 Java 组件。可为诸如图形功能、文件处理等提供较好的支持,也可通过 JDBC 提供强有力的数据库支持,且不牺牲安全性。通过 Java 虚拟机(Java Virtual Machine, JVM)得到了内置的 Java 支持。Java 小应用程序需要先下载到浏览器上,再在 JVM 上执行和处理。

4) Flash

Macromedia 的 Flash 是 Web 上的动态图形工具,它允许开发人员创建动画、交互式的图形和用户界面元素。在 Web 页面上使用 Flash 插件已经是浏览器的标准,因而广受欢迎。

2. 提供动态内容的服务器端技术

提供动态内容的服务器端技术依赖于 Web 服务器端的模块运行来实现,只有 HTML 和客户端脚本代码才是通过 Web 服务器传递到浏览器上运行。服务器端技术有以下几种。

1) CGI 公共网关接口(Common Gateway Interface, CGI)

这是在服务器上创建脚本的一种机制,可用来创建动态 Web 应用程序。它是通过调用 Web 服务器上的另一个程序并将用户提供的数据传递给该程序进行处理来创建动态 Web 应用程序。缺点是:初学者不易掌握编写这样的模块;会占用许多的服务器资源;需要另外运行 CGI 程序提供动态内容等。

2) ASP 动态服务器页面(Active Server Pages, ASP)

它依赖脚本语言创建动态 Web 页面。ASP 是用一个附加到其 Web 服务器上的模块 asp.dll 在 Web 服务器上处理这些脚本,然后转换成 HTML。能够提供数据库存取、图形处理、网络功能、系统功能等,但性能低下,脚本语言功能不完善。

3) JSP Java Server Pages(JSP)

这是一种允许用户将标记(HTML 或 XML)与 Java 代码组合,动态生成 Web 页面的技术。主要的优点是代码在不同的服务器间的兼容性,功能强大,运行速度快于 ASP。JSP 允许 Java 程序利用 Java 2 平台的 JavaBeans 和 Java 2 库。

4) ColdFusion

ColdFusion 能够使服务器在创建 HTML 页面时访问数据。它是安装在 Web 服务器上的模块。利用 HTML 样式的标识符, 将功能封装, 并由 ColdFusion Server 软件处理。该服务器软件可运行在不同的平台上, 不是免费的。

5) PHP(HyperText Preprocessor, 超文本预处理语言)

PHP 指超文本预处理程序, 也是一种脚本语言, 当访问者打开页面时, 服务器处理 PHP 命令, 并将结果传回浏览器。具有开放的源代码和跨平台, 缺点是安装复杂, 会话管理功能弱。

6) ASP.NET

它是当前流行的技术之一, 利用 ASP.NET 模块和 .NET 架构处理数据, 其运行机制更为优秀。

1.3 Web 数据库系统的实现技术

1.3.1 Web 与数据库技术的结合

随着 Internet 的飞速发展, WWW(World Wide Web)得到了很大的发展。WWW 系统和数据库是网络化信息服务的基础。Internet 用户主要通过访问站点以获取信息, 而大部分的站点都是静态的或者由发布者隔一段时间对其进行修改。如何才能得到动态的页面? 这就是由数据库 WWW 服务器来完成的。通过 Web 页面实现对数据库访问的基本框图如图 1.1 所示。

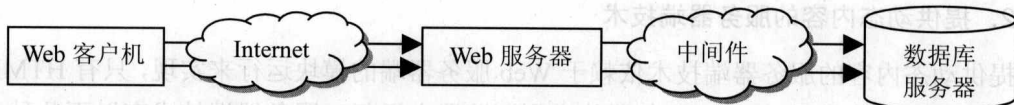


图 1.1 通过 Web 页对数据库访问模型

Web 客户机指需要带 Web 浏览器的客户端, Web 服务器指任何基于 HTML 的服务器软件, 如 Microsoft 的 IIS(Internet Information Server)。数据库应用程序驻留在 Web 服务器上, 最终用户通过 Web 浏览器生成大量的数据库应用程序。浏览器借助 HTTP 协议, 通过 Internet 与 Web 服务器通信。

中间件(Middle Ware)负责管理 Web 服务器和数据库之间的通信并提供应用程序服务。它能直接调用外部程序或脚本代码来访问数据库, 提供与数据库相结合的动态 HTML 页面, 然后通过 Web 服务器返回给用户浏览器。

数据库服务器负责管理驻留在数据库服务器中的数据。

1.3.2 Web 数据库系统的体系结构

目前应用广泛的 Web 数据库系统是一种基于 B/S(Browser/Server)结构的系统。B/S 模式