



SQL Server 2000

数据库教程

• 唐学忠 主编 费贤举 腾 刚 胡智喜 副主编

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



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

编委会名单

主 编：庄燕滨

副主编：常明华 华容茂 邵晓根 陈 雁
顾元刚 杨萃南

委 员：（以姓氏笔画为序）

邓 凯	朱宇光	华容茂	庄燕滨	刘红玲
许秀林	杨晓云	杨萃南	吴国经	沈玉书
宋依青	张永常	张家超	陈 雁	陈志荣
邵晓根	周维武	郑成增	顾元刚	徐煜明
高 波	常明华	常晋义	韩 雁	谢志荣
薄继康				

前 言

Microsoft SQL Server 2000 是在 Microsoft SQL Server 7.0 的基础上不断完善推出的一门新的数据库管理工具,使用户能更方便快捷地管理数据库开发应用程序。Microsoft SQL Server 2000 使用了工业界最先进的数据库构架,与 Microsoft Windows DNA 2000 平台紧密集成,具有完全的 Web 功能。通过对高端硬件平台及最新网络和存储技术的支持,可以为最大的 Web 站点和企业级的应用提供可扩展性和高可靠性,使用户能够在 Internet 商业领域快速创建应用,从而减少了建立电子商务应用、商业智能数据仓库和商业线路应用所需的时间。此外,Microsoft SQL Server 2000 提供了重要的安全性方面的保护措施,支持灵活的基于角色的安全策略,拥有安全审计工具,并提供高级的文件加密和网络加密功能。

作为一门高质量的数据库开发和管理工具,很多高校纷纷开设了相关课程,但遗憾的是,目前国内 Microsoft SQL Server 2000 语言方面的教材虽然品种繁多,但真正适合教学的教材很少。

本书是由多年从事数据库开发和设计语言教学的教师和科研人员根据基础教学的特点精心组织和编写的,从 SQL 语言及 Microsoft SQL Server 2000 的基本操作入手,结合具体的实例深入浅出,系统地介绍了 Microsoft SQL Server 2000 的运用。

本书共分 14 章,分别讲述了 SQL Server 2000 的 Transact-SQL 语言基础数据库管理、表、存储过程等数据库对象的管理、数据完整性与数据查询用户和安全性管理、数据库的备份恢复等有关内容。

本书面向初、中级用户,尤其适合于使用面向对象语言编写 C/S、B/S 数据库应用程序的用户,在内容安排上,本着从入门到精通的原则,内容合理,语言通俗易懂,实例详尽,如果能结合上机实践,则一定能收到更好的效果。

本书由唐学忠主编。第 1、2、3、4、10 章由胡智喜编写;第 6 章由费贤举编写;第 5、7、8、12 章由唐学忠编写;第 9 章由李亦飞编写;第 11 章由吴倩编写;第 13、14 章由腾刚编写;钱池波参与了第 12 章部分内容的编写。全书由唐学忠统稿,由郑成增主审,最后由唐学忠定稿。另外,王文琴、王文和袁军为本书的出版做了大量的文字工作,在此一并表示感谢。

由于作者水平有限,书中的错误在所难免,敬请广大读者批评并提出宝贵的修改意见。读者在学习过程中,如有其他问题和意见,可以直接与编者联系,具体联系方式:

E-mail: tangxz@czu.cn, huzx@czu.cn

编 者

目 录

第 1 章 数据库技术简介	1
1.1 数据库系统应用程序设计方案简介	1
1.1.1 数据库及数据库系统	1
1.1.2 数据库系统应用程序模型	2
1.1.3 常用数据库访问接口	3
1.2 SQL Server 2000 简介	5
1.2.1 什么是 SQL Server 2000	5
1.2.2 SQL Server 2000 的特点	5
1.2.3 SQL Server 2000 具有的新特性	6
1.3 SQL Server 2000 版本	6
1.3.1 企业版 (Enterprise Edition) 与开发版 (Developer Edition)	6
1.3.2 标准版 (Standard Edition)	6
1.3.3 个人版 (Personal Edition)	6
1.3.4 MSDE/Desktop Engine	6
1.3.5 SQL Server 2000 CE 版	7
1.4 SQL Server 2000 的组件	7
1.4.1 SQL Server 2000 服务器组件	7
1.4.2 SQL Server 2000 客户组件	8
1.5 SQL Server 2000 的系统需求	11
1.5.1 安装、运行 SQL Server 2000 的硬件需求	11
1.5.2 安装、运行 SQL Server 2000 的软件需求	11
1.6 SQL Server 2000 中常用的数据对象	11
1.6.1 数据库对象	11
1.6.2 数据库对象的引用方法	13
1.7 本章小结	13
第 2 章 SQL Server 2000 企业管理器	15
2.1 企业管理器的环境	15
2.2 启动与关闭服务器	15
2.2.1 启动 SQL Server 2000	15
2.2.2 暂停、停止 SQL Server 2000	17
2.3 注册服务器	17
2.4 连接和断开服务器	19
2.5 配置服务器	19
2.6 本章小结	22
第 3 章 Transact-SQL 语言	23
3.1 SQL 语言	23

3.2	Transact-SQL 语言概述	24
3.2.1	Transact-SQL 编程语言	24
3.2.2	Transact-SQL 的元素	24
3.3	数据类型	26
3.3.1	系统提供数据类型	26
3.3.2	用户定义数据类型	29
3.4	变量	30
3.4.1	局部变量	30
3.4.2	全局变量	31
3.5	注释符、运算符和通配符	31
3.5.1	注释符	31
3.5.2	运算符	32
3.5.3	通配符	34
3.6	流程控制命令	34
3.6.1	BEGIN...END	35
3.6.2	IF...ELSE	35
3.6.3	GOTO label	36
3.6.4	RETURN	36
3.6.5	WHILE	36
3.6.6	WAITFOR	37
3.6.7	BREAK 和 CONTINUE	37
3.6.8	CASE 语句	37
3.7	其他命令	39
3.7.1	批处理	39
3.7.2	调试命令	40
3.7.3	切换数据库命令 USE	41
3.8	常用函数	41
3.8.1	集合函数	41
3.8.2	数学函数	42
3.8.3	字符串函数	43
3.8.4	日期函数	43
3.8.5	转换函数	44
3.8.6	系统函数	45
3.9	本章小结	45
第 4 章	管理数据库	47
4.1	创建数据库对象	47
4.1.1	创建新数据库的注意事项	47
4.1.2	文件与文件组	48
4.1.3	使用企业管理器创建数据库	49
4.1.4	使用 Transact-SQL 语句创建数据库	51

4.1.5	数据库对象属性设置	53
4.2	更改数据库	55
4.2.1	使用企业管理器修改数据库	55
4.2.2	使用 Transact-SQL 语句修改	56
4.3	删除数据库	58
4.4	压缩数据库	58
4.4.1	压缩整个数据库	59
4.4.2	压缩数据库中的数据文件	59
4.4.3	自动压缩数据库	59
4.5	系统数据库介绍	60
4.5.1	数据库目录系统表	60
4.5.2	Master 数据库	61
4.5.3	Msdb 数据库	62
4.5.4	Model 数据库	62
4.5.5	Tempdb 数据库	62
4.5.6	Pubs 数据库和 Northwind 数据库	63
4.6	本章小结	63
第 5 章	管理数据库表及其完整性	64
5.1	创建数据库表	64
5.1.1	设计数据库表	64
5.1.2	使用 SQL Server 2000 企业管理器创建表	65
5.1.3	使用 Transact-SQL 语句创建表	67
5.1.4	创建和使用 5 种基本约束	74
5.1.5	数据完整性分类	75
5.1.6	PRIMARY KEY 约束	75
5.1.7	FOREIGN KEY 约束	76
5.1.8	UNIQUE 约束	77
5.1.9	CHECK 约束	78
5.1.10	DEFAULT 约束	78
5.2	修改数据库表	79
5.3	查看数据库表	87
5.4	删除数据库表	87
5.5	创建和使用规则	87
5.5.1	创建规则	88
5.5.2	规则应用	88
5.5.3	删除规则	89
5.6	创建和使用默认	91
5.6.1	创建默认	91
5.6.2	默认的应用	91
5.6.3	删除默认值	93

5.7	本章小结	94
第 6 章	索引	95
6.1	索引简介	95
6.1.1	为什么要建立索引	95
6.1.2	为什么不在每一列上创建索引	96
6.1.3	考虑建立索引的列	97
6.1.4	不考虑建立索引的列	97
6.2	创建索引	98
6.2.1	使用创建索引向导创建索引	98
6.2.2	使用 Transact-SQL 语句创建索引	98
6.2.3	惟一索引	104
6.2.4	复合索引	105
6.2.5	创建索引的选项	105
6.3	索引信息查看和删除索引	107
6.4	维护索引	107
6.4.1	数据分段	108
6.4.2	DBCC SHOWCONTIG	108
6.4.3	DBCC DBREINDEX 语句	114
6.4.4	索引统计值	114
6.4.5	索引分析	115
6.5	本章小结	117
第 7 章	数据库查询技术	118
7.1	SELECT 语句	118
7.1.1	SELECT 子句	119
7.1.2	INTO 子句	121
7.1.3	FROM 子句	121
7.1.4	WHERE 子句	123
7.1.5	GROUP BY 子句	124
7.1.6	HAVING 子句	124
7.1.7	UNION 操作符	124
7.1.8	ORDER BY 子句	125
7.1.9	COMPUTE 子句	125
7.1.10	FOR BROWSE 子句	126
7.1.11	OPTION 子句	127
7.2	简单查询	129
7.2.1	选择列	129
7.2.2	选择行	130
7.2.3	使用 ORDER BY 子句对查询结果排序	131
7.2.4	查询结果分组	131
7.2.5	使用聚集函数	132

7.3	连接查询	133
7.3.1	等值连接查询	133
7.3.2	非等值连接查询	134
7.3.3	自连接查询	134
7.3.4	外部连接查询	134
7.3.5	复合条件连接查询	135
7.4	合并查询	136
7.5	嵌套子查询	136
7.6	存储查询结果	139
7.6.1	存储查询结果到表中	139
7.6.2	存储查询结果到变量中	139
7.7	全文检索	139
7.7.1	全文索引和目录	140
7.7.2	创建全文索引目录	141
7.7.3	创建全文索引	141
7.7.4	全文查询	143
7.8	本章小结	146
第 8 章	数据库更新	147
8.1	添加数据	147
8.1.1	INSERT 语法	147
8.1.2	使用 INSERT INTO...VALUES 插入单行数据	149
8.1.3	使用 INSERT ... SELECT 语句插入多行数据	150
8.1.4	使用存储过程插入数据	151
8.2	修改数据	151
8.2.1	UPDATE 语法	151
8.2.2	使用 UPDATE 修改一行数据	156
8.2.3	更新多行数据	157
8.2.4	含子查询的数据更新	157
8.2.5	大量数据的更新	157
8.3	删除数据	158
8.3.1	DELETE 语法	158
8.3.2	删除一行数据	160
8.3.3	删除多行数据	160
8.3.4	含子查询的数据删除	160
8.3.5	删除当前游标行数据	160
8.3.6	使用 TRUNCATE TABLE 命令	160
8.4	使用事务	160
8.4.1	事务由来	160
8.4.2	事务概念	161
8.4.3	事务的使用	161

8.4.4	事务回滚	162
8.5	使用锁	163
8.5.1	锁的概念	163
8.5.2	锁的类型	163
8.5.3	隔离级 (Isolation)	164
8.5.4	查看锁	164
8.5.5	死锁 (Deadlocking) 的预防	164
8.6	本章小结	165
第 9 章	存储过程和触发器	166
9.1	存储过程概述	166
9.1.1	存储过程的种类	166
9.1.2	存储过程的优势	167
9.2	创建存储过程	167
9.2.1	创建存储过程	167
9.2.2	执行存储过程	169
9.3	修改存储过程	169
9.3.1	修改存储过程	169
9.3.2	删除存储过程	170
9.4	参数化存储过程	170
9.4.1	带输入参数的存储过程	170
9.4.2	指定默认值	171
9.4.3	带输出参数的存储过程	171
9.5	存储过程中的错误处理	172
9.5.1	错误处理	172
9.5.2	用户自定义错误信息	173
9.6	触发器概述	174
9.7	管理触发器	175
9.7.1	创建触发器	175
9.7.2	删除触发器	177
9.8	触发器的工作原理	177
9.8.1	inserted 表和 deleted 表	177
9.8.2	触发器的类型	179
9.8.3	触发器限制	179
9.8.4	触发器的嵌套调用	180
9.9	INSTEAD OF 触发器	181
9.9.1	INSTEAD OF INSERT 触发器	181
9.9.2	INSTEAD OF UPDATE 触发器	183
9.9.3	INSTEAD OF DELETE 触发器	183
9.10	触发器的应用	184
9.10.1	Insert 型触发器的应用	184

9.10.2	Update 型触发器的应用	184
9.10.3	Delete 型触发器的应用	185
9.11	触发器的高级应用	185
9.12	本章小结	188
第 10 章	视图、游标和自定义函数	189
10.1	视图简介	189
10.1.1	视图的概念	189
10.1.2	视图的优缺点	190
10.2	创建和管理视图	190
10.2.1	创建视图	190
10.2.2	管理视图	197
10.2.3	编辑视图	198
10.2.4	删除视图	199
10.3	通过视图管理数据	199
10.3.1	通过视图检索数据	199
10.3.2	通过视图插入数据	200
10.3.3	通过视图删除数据	200
10.3.4	通过视图更改数据	201
10.4	游标简介	202
10.4.1	游标的定义及种类	202
10.4.2	游标声明	203
10.4.3	游标的使用	204
10.5	游标的应用	206
10.5.1	使用游标查询	206
10.5.2	使用游标更改数据	207
10.5.3	使用游标删除数据	208
10.6	创建和使用自定义函数	209
10.6.1	标量函数	209
10.6.2	内嵌表值函数	210
10.6.3	多语句表值函数	212
10.7	本章小结	214
第 11 章	用户和安全管理	215
11.1	SQL Server 2000 的登录认证	215
11.1.1	身份验证阶段	215
11.1.2	权限验证阶段	215
11.2	管理 SQL Server 2000 登录	216
11.2.1	验证模式	216
11.2.2	设置验证模式	216
11.2.3	管理登录账号	217
11.3	数据库用户	220

11.3.1	查看数据库用户	220
11.3.2	新增数据库用户	221
11.3.3	修改和删除数据库用户	222
11.4	权限管理	222
11.4.1	权限设置对话框	222
11.4.2	权限设置	223
11.5	角色管理	224
11.5.1	标准角色	224
11.5.2	应用程序角色	227
11.5.3	用户和角色的权限问题	228
11.6	本章小结	229
第 12 章	数据库备份和恢复	231
12.1	备份和恢复概述	231
12.1.1	备份和恢复概述	231
12.2	数据库备份	234
12.2.1	用企业管理器管理备份设备	234
12.2.2	使用系统存储过程管理备份设备	235
12.2.3	用企业管理器管理备份	236
12.2.4	Transact-SQL 语句备份数据库	239
12.3	恢复数据库	242
12.3.1	利用企业管理器恢复数据库	242
12.3.2	使用 RESTORE 命令进行恢复	244
12.3.3	使用 RESTORE 命令恢复文件或文件组	245
12.3.4	使用 RESTORE 命令恢复事务日志	245
12.4	本章小结	246
第 13 章	数据库应用程序设计概述	247
13.1	ADO 和客户端/服务器体系	247
13.1.1	VB6 中的数据访问技术	247
13.1.2	数据对象接口	248
13.1.3	数据控件	248
13.1.4	ADO 对象模型	248
13.1.5	客户端/服务器体系	249
13.2	ADO.NET 和 n-层体系	250
13.2.1	ADO.NET 简介	250
13.2.2	n-层体系结构	251
13.3	ADO.NET 的新功能	252
13.3.1	ADO.NET 的新功能	252
13.3.2	数据共享的促进	253
13.4	DataReader 和 DataSet	254
13.4.1	DataReader	255

13.4.2	DataSet	255
13.4.3	使用 DataSet 代替 DataReader 的时机	256
13.5	托管提供程序	257
13.6	用于数据访问的命名空间	259
13.6.1	使用命名空间	259
13.6.2	使用 System.Data.SqlClient 名称检索数据	261
13.6.3	使用 System.Data.OleDb 命名空间检索数据	264
13.7	Visual Studio 的数据库应用程序工具	265
13.7.1	服务器资源管理器	265
13.7.2	数据工具箱	266
13.7.3	数据适配器配置向导	266
13.8	本章小结	269
第 14 章	使用 ADO.NET	270
14.1	使用数据组件访问数据库	270
14.1.1	用于数据访问的组件	270
14.1.2	DataView 对象	275
14.1.3	数据绑定	276
14.2	SqlConnection 对象	278
14.2.1	SqlConnection 的属性	279
14.2.2	SqlConnection 的方法	280
14.2.3	连接池	281
14.3	SqlCommand 对象	282
14.3.1	SqlCommand 的属性	283
14.3.2	SqlCommand 的方法	284
14.3.3	传递存储过程参数	287
14.4	SqlDataReader 对象	291
14.4.1	SqlDataReader 的属性	291
14.4.2	SqlDataReader 的方法	291
14.4.3	SqlDataReader 的使用	292
14.5	SqlDataAdapter 对象	297
14.5.1	SqlDataAdapter 的属性和方法	298
14.5.2	更新数据	299
14.6	DataSet 对象	301
14.6.1	DataSet 的属性	301
14.6.2	DataSet 的方法	302
14.7	DataTable 对象	304
14.7.1	DataTable 的属性	305
14.7.2	DataTable 的方法	306
14.8	DataRow 对象	308
14.8.1	DataRow 的属性	309

14.8.2	DataRow 的方法	310
14.8.3	RowVersion	311
14.9	DataColumn 对象	311
14.10	DataRelation 对象	315
14.10.1	DataRelation 的属性	316
14.10.2	使用 DataRelation	317
14.10.3	约束条件	317
14.11	本章小结	319

第 1 章 数据库技术简介

随着计算机技术的蓬勃发展，计算机应用已经涉及到人们日常生活、工作的各个领域。尤其在当今信息社会，计算机已成为人们日常工作中处理数据的得力助手和工具。数据处理是计算机四大应用（科学计算、过程控制、数据处理和辅助设计）的一个主要方面，而且已经渗透到许多其他应用领域。本章将从数据库的基本概念出发，介绍数据库及数据库系统的相关概念、知识和技能，然后着重介绍一种典型的关系型数据库管理系统 SQL Server 2000，为进一步学习数据库技术及其应用奠定坚实的基础。

1.1 数据库系统应用程序设计方案简介

本节首先介绍数据库和数据库系统的基本概念，接着介绍几种数据库系统应用程序模型，最后介绍常用的数据库访问接口。

1.1.1 数据库及数据库系统

1. 数据库

数据库，顾名思义，是存放数据的仓库。只不过这个仓库是在计算机存储设备上，而且数据是按一定的格式存放的。数据库就是在计算机存储器中用于存储数据的仓库。

利用数据库技术，人们可以科学地保存和管理大量的复杂的数据，以便能充分利用这些信息资源，对信息资源进行处理，提炼出对决策有用的数据和信息。

2. 数据库系统

数据库系统是指在计算机系统中引入数据库后的系统，一般由数据库、数据库管理系统、应用程序系统和数据库用户几个部分组成。

（1）数据库

数据库是数据库系统的核心和管理对象，是有效数据的存储基地。大量的数据按一定的数据模型组织存储在数据库中，便于实现数据共享。数据库一般由应用程序员利用计算机数据库商家提供的数据库管理系统中的某一工具创建一个库结构（表格），再由数据库管理人员利用数据库管理系统或应用程序系统提供的工具将有用的数据填入设计好的库中，形成一个有效的数据库，并提供给多个终端用户共享和使用。

（2）数据库管理系统

数据库管理系统是对数据库进行管理和实现对数据库的数据进行操作的管理系统。它是建立在操作系统基础上的，是位于操作系统与用户之间的一层数据管理软件，负责对数据库的数据进行统一的管理和控制。用户发出的或应用程序中的各种操作数据库及其中数据的命令，都要通过 DBMS 来执行，如数据库创建，数据的定义、查询、更新（增加、删除和修改）

等都要通过 DBMS 进行。DBMS 一般都由专业的软件商家研制，形成商业软件包，并提供一套较为完整的数据库语言（相当于一种高级语言）。本书的主要内容就是介绍一个典型的 DBMS-SQL Server 2000。

（3）应用程序系统

这里所说的应用程序系统是指数据库应用程序系统，它是针对某一个管理对象（应用）而设计的一个面向用户的软件系统，是建立在 DBMS 基础上的，而且具有良好的交互操作性和用户界面，如学生选课管理系统、人事管理系统、财务管理系统等均为一个数据库应用程序系统。它与数据库管理系统和数据库一同构成数据库软件系统。

（4）数据库用户

严格地说，一个完整的数据库系统还应该包含数据库用户。数据库用户根据他们的工作内容可分成以下三类人员。

① 终端用户：这类人员一般是不要求精通计算机的各级管理人员，他们一般使用应用程序提供的菜单来操作数据库、生成报表等。

② 应用程序员：这类人员是负责设计和编制应用程序的人员。

③ 数据库管理员：这类人员是指全面负责数据库系统的管理维护，保证系统能够正常使用的人员。

1.1.2 数据库系统应用程序模型

数据库应用程序可以有效地对数据源（如 SQL Server 2000 数据源）进行各种操作，它为了给上层用户提供特定的服务而去访问低层的数据库源。为了满足特定的服务要求，应用程序按照某种规则对特定的数据集进行特殊的操作，所以数据库（系统）应用程序本身是一个分层系统，至少包括数据层、规则层和数据表示层等层次。数据层是数据存储的场所，包括数据库定义、数据完整性逻辑及其他与数据密切相关的对象。规则层定义了对数据源进行何种处理，如插入、删除、更新数据源中的数据等。数据表示层规定将数据展示给应用程序的形式。

根据数据层、规则层和数据表示层的位置分布不同，数据库应用程序有单层、两层、三层或 N 层几种模型。

1. 单层

在单层数据库（系统）应用模型中，每一层并没有明显的界定，其功能可以相混以实现一些操作。这种数据库应用模型与文件系统结合紧密。从文件中读取或输入配置信息，并把结果保存到文件系统中。由于应用程序对文件采取了不同的读取方法，文件的内容往往只对特定的应用程序有意义，这样为数据共享带来了极大的不便。这样的应用程序称为文件密集型应用程序。

数据库的出现使得数据共享变得容易，数据库定义了数据的标准组织结构和存储格式。不同的应用程序可以通过数据访问接口操作数据库。此时数据库和应用程序（可以有两层或三层）驻留在一台机器上，该应用程序就是单层数据库应用模式。

2. 两层（客户-服务器）数据库应用模型

在两层（客户-服务器）数据库应用模型中，客户应用向服务器应用提出请求，服务器应用响应请求并将结果返回给客户应用，客户应用解释响应结果。两层（客户-服务器）数据库应用模型能充分利用网络的优势，将任务分解到多台计算机上，从而提高数据库应用的性能。

在客户-服务器模型中，一般把数据层放在服务器上，数据表示层放在客户机上，而规则层的分布比较灵活。根据规则层的分布不同，客户-服务器模型通常有以下两种不同的情况。

一种是“智能服务器”模型。在这种模型中，不仅数据层放在服务器上，规则也主要在服务器上实现。在这种设计中，客户机的任务较轻，只负责接受用户的请求并将请求转交给服务器，把服务器处理后的结果“翻译”后展示给用户；而服务器的任务比较重，根据用户的请求按照一定的规则操作数据库，并将结果返回给客户。在这种设计中，由于数据库和业务处理服务竞争相同的硬件资源，服务器会成为一个瓶颈，因而这种模型对服务器的性能要求较高。

另一种客户-服务器模型是“智能客户机”模型。与“智能服务器”模型相反，大多数数据处理和规则发生在客户机上，服务器仅仅充当数据库服务器。这种设计是目前广泛使用的客户-服务器应用模型。它的缺点在于网络交通会比较繁忙，如果事务（业务处理单元）较长时，数据库应用的性能会大大降低。

需要说明的是，客户和服务是相对于特定的应用时刻而言的。一个客户应用在另一个客户应用中可能是服务，而一个客户应用中的服务程序可能同时在另一个客户应用中充当客户的角色。即使在同一个应用程序中，两者的角色也不固定，关键看谁发起请求和谁提供服务。

3. 三层/N层模型

三层/N层模型把规则层和数据层分开，客户直接对规则层操作，规则层与数据层进行通信，这种模型在当前的 Web 环境中得到了广泛应用，即通常所说的浏览器/服务器（B/S）模型。在这种模型中，数据层放在专用的数据库服务器上，负责规则处理的是中间层（Web Server），它既是浏览服务器，又是应用服务器，可以运行大量的应用程序，从而使客户端变得很简单。前台采用网页浏览器，如 IE、网景浏览器等。根据需要，如果在服务器端和客户端之间添加更多的应用服务器或数据库服务器，则数据库应用演变为 N 层模型。

三层/N层模型中的数据表示层比较灵活，它们的功能比较单一，只要接受请求并表示服务器的响应即可，不需要包含任何规则。这样数据库系统中可以有多种多样的客户，并且可以共享一套规则。

1.1.3 常用数据库访问接口

数据库应用程序要访问数据库，必须使用一定的数据库访问接口。在 Windows 平台下主要有以下几种接口，即开放数据互联、DB 库、内嵌 SQL、数据访问对象、远程数据库对象、OLE DB、ADO 对象和 ADO.Net。

1. 开放数据互联 (ODBC) 和 DB 库

ODBC 和 DB 库提供了函数调用层次上的数据库访问接口, 它们通过一套 API 函数屏蔽低层不同的数据源, 为上层的数据库应用提供一致的访问界面。这两套 API 函数操作数据库的性能几乎相同, 但 ODBC 的 API 函数接口相对于 DB 库具有以下优势:

- ① ODBC 学习较容易。ODBC 用同一 API 函数实现相同的数据库操作, 而 DB 库采用不同的 API。
- ② ODBC 更充分地利用 SQL Server 2000 提供的支持。
- ③ ODBC 已经成为通用的工业标准。

2. 内嵌 SQL

C 语言支持内嵌的 SQL 语句访问数据库。含有内嵌 SQL 语句的程序首先经过预编译, 将内嵌 SQL 语句翻译成对数据库的调用。为了得到可执行程序, 需要连接相关支持库。内嵌 SQL 主要用来向 SQL Server 2000 移植其他数据库应用程序。

3. 数据访问对象 (Data Access Object, DAO)

DAO 提供了一个面向对象的数据库编程接口。与 ODBC 和 DB 库不同, 它没有提供封装成 DLL 的 API 函数, 而是提供了一组对象模型, 同时提供了诸如连接管理、记录集操作的方法, 大大降低了数据库编程的难度。

4. 远程数据库对象 (Remote Data Object, RDO)

RDO 与 DAO 极其相似, 它提供了一组对象模型访问 ODBC 数据源。它包装了 ODBC 低层的所有 API 函数。另外, 它还提供了不用 API 函数就能访问 ODBC 数据库的简单方法。RDO 通常跟 SQL Server 2000 和 Oracle 数据库一起使用。

5. OLE DB

OLE DB 提供了一组非常低层的 COM 接口, 数据库应用程序通过该组 COM 接口可以访问实现了该组接口的数据源。实现 OLE DB 接口的应用程序称为 OLE DB 提供者, 访问该接口的应用程序称为 OLE DB 消费者。

OLE DB 既可以访问关系数据库, 又可以访问非关系数据库, 如 ISAM/VASM、非平面的等级数据库、E-mail、文件系统、文本和图像等。

6. ADO (Active Data Object)

ADO 是 OLE DB 的消费者, 它利用低层 OLE DB 为应用程序提供简单高效的数据库访问接口。ADO 提供了高层的对象模型接口使得数据库编程非常简捷, 同时保持 OLE DB 访问数据库的良好性能。ADO 已成为目前应用最广的数据库访问接口。

用 ADO 实现 Web 数据库应用非常方便。通过 Vbscript 或 Javascript 在 ASP 中很容易操作 ADO 对象, 从而轻松地将数据库带到 Web 前台。