

SQL Server

数据库开发技术

汤海滨 编著



科学出版社

SQL Server 数据库开发技术

汤海滨 编著

科学出版社

2000

内 容 简 介

SQL 是一种通用的数据库语言。SQL Server 是基于 SQL 语言基础上的客户/服务器数据库。

本书可以帮助数据库开发人员快速学习 SQL Server 7.0。本书对 SQL Server 7.0 的数据库开发和 SQL Server 7.0 新增功能的使用进行了详细地介绍, 这些新内容使 SQL Server 7.0 更成熟、更易于使用。

本书语言简洁、流畅, 提供了大量的程序实例, 可读性较强, 适合广大初级用户使用。

图书在版编目(CIP)数据

SQL Server 数据库开发技术/汤海滨编著. —北京: 科学出版社, 2000
ISBN 7-03-005369-9

I.S… II.汤… III. 服务器-数据库系统-软件开发 IV.TP311.13

中国版本图书馆 CIP 数据核字(2000)第 64142 号

科学出版社 出版

北京东黄城根北街 16 号
邮政编码: 100717

北京双青印刷厂印刷

科学出版社总发行 各地新华书店经销

*

2000 年 9 月第 一 版 开本: 787×1092 1/16

2000 年 9 月第一次印刷 印张: 23

印数: 1—5 000 字数: 520 000

定价: 29.00 元

(如有印装质量问题, 我社负责调换(环伟))

前 言

SQL Server 7.0 是微软公司对 SQL Server 低版本的升级，新发布的 SQL Server 7.0 与前面几个版本相比，在很多方面都有了很大的提高，并且在各方面都进行了完善。

SQL Server 7.0 不仅可以在 Windows NT 环境下开发数据库，而且还可以在 Windows 95/98 的环境下进行数据库的开发，只是不能作为服务器来使用。

在这本书中，我们对于 SQL Server 7.0 新增加的功能和改进的性能进行了较详细的介绍，使读者对 SQL Server 7.0 有一个较清楚的了解。

在后面的内容中，我们按照循序渐进的规律对 SQL Server 7.0 的使用和开发进行了详细的说明，其中主要包括：

1. 如何创建和管理 SQL Server 数据库；
2. 如何创建和管理 SQL Server 数据表格；
3. 如何创建和管理 SQL Server 视图；
4. 如何创建和使用 SQL Server 存储过程；
5. 如何创建和使用 SQL Server 索引；
6. 在 SQL Server 环境下，如何移动数据。这其中包括：
 - 1) 数据的批量复制；
 - 2) 数据的备份和恢复。
7. 如何检索数据和修改数据库中的数据。

通过对这些操作的详细介绍，使您可以很好地应用 SQL Server 7.0 开发数据库。

本书主要是面向初级学者，希望这本书可以使读者在数据库方面以及 SQL Server 7.0 的使用方面有很大的提高。

作 者
2000 年 2 月

目 录

第 1 章 SQL Server 7.0 介绍	1
1.1 SQL Server 体系结构	1
1.1.1 客户/服务器体系结构	1
1.1.2 数据库结构	4
1.1.3 服务器结构	5
1.1.4 管理体系结构	5
1.1.5 复制体系	6
1.1.6 数据库应用程序开发体系	6
1.2 SQL Server 7.0 中的新内容	7
1.2.1 服务器的改进和新特性	7
1.2.2 图形化管理方面的改进和新增功能	14
1.2.3 SQL Server 7.0 体系结构的新内容	16
1.2.4 新增加的服务器功能	17
1.2.5 SQL Server 数据库开发中的新内容	19
1.3 小结	19
第 2 章 SQL Server 数据库	20
2.1 SQL Server 数据库的结构组成	20
2.1.1 SQL Server 中的文件	21
2.1.2 事务日志	22
2.2 设计 SQL Server 数据库	22
2.2.1 建立创建数据库计划	23
2.2.2 事务处理和决定支持	24
2.2.3 数据标准化	25
2.2.4 数据完整性	26
2.2.5 数据安全性	27
2.2.6 数据库的维护	28
2.2.7 估计数据库的大小	28
2.3 SQL Server 数据库管理	33
2.3.1 创建数据库	34
2.3.2 删除数据库	42
2.3.3 修改数据库	43
2.4 小结	56

第 3 章 SQL Server 数据库中的表格	57
3.1 数据表格的设计	57
3.1.1 指定每一列的数据类型	57
3.1.2 限制、缺省值和空值	62
3.2 创建和修改数据表格	65
3.2.1 创建和修改数据表格	65
3.2.2 数据表格中列属性的设置	73
3.2.3 增加和删除列	84
3.2.4 创建和修改主键限制	86
3.2.5 创建和修改外部键限制	89
3.2.6 创建和修改唯一性限制	91
3.2.7 创建和修改检查限制	94
3.2.8 创建和修改缺省定义	97
3.2.9 创建和修改标识符列	102
3.3 小结	104
第 4 章 SQL Server 的数据库图表	105
4.1 创建和管理数据库图表	105
4.2 向数据库图表中添加数据表格	107
4.2.1 向数据库图表中添加新的数据表格	107
4.2.2 向图表中添加已经存在的表格	108
4.2.3 向图表中添加关系表格	109
4.3 数据库图表导航	109
4.4 数据库图表的格式	109
4.5 数据库图表的细分	111
4.6 多用户环境下工作	112
4.6.1 数据库对象的所有权	113
4.6.2 其他用户对数据库图表的影响	113
4.6.3 数据库对象被其他用户所删除	113
4.6.4 协调被修改了的数据库中的数据库图表	114
4.7 小结	114
第 5 章 SQL Server 数据库中的索引	115
5.1 设计索引	116
5.1.1 簇索引	117
5.1.2 非簇索引	117
5.1.3 唯一索引	118
5.1.4 填充因素	118

5.2 创建索引	119
5.2.1 创建表格时同时创建索引	120
5.2.2 在已经存在的表格中创建索引	121
5.2.3 利用创建索引向导创建索引	127
5.2.4 统计信息	129
5.3 重新创建索引	138
5.3.1 删除索引	139
5.3.2 在一个步骤中重新创建索引	140
5.3.3 在一个表格中重新创建一个或者多个索引	140
5.4 索引的重新命名	141
5.5 观察索引	142
5.5.1 观察一个表格中的索引	142
5.5.2 观察一个数据库中所有的索引	143
5.5.3 查看索引所用的空间	143
5.6 观察索引的属性	144
5.7 小结	145
第 6 章 SQL Server 数据库中的视图	146
6.1 关于使用视图	146
6.2 创建视图	147
6.2.1 创建视图	148
6.2.2 利用创建视图向导	152
6.3 视图的修改和重新命名	155
6.3.1 修改视图	155
6.3.2 视图的重新命名	158
6.4 通过视图修改数据	159
6.4.1 通过视图添加数据	159
6.4.2 通过视图修改数据	159
6.4.3 通过视图删除数据	159
6.5 获得关于视图的信息	160
6.5.1 获得视图的信息	160
6.5.2 观察视图中的数据	161
6.5.3 显示视图相关的数据库对象	162
6.6 小结	163
第 7 章 SQL Server 数据库中的存储过程	164
7.1 扩展的存储过程	164
7.1.1 使用企业管理器创建扩展的存储过程	165
7.1.2 利用 Transact-SQL 语句创建扩展的存储过程	165

7.2	创建存储过程	166
7.2.1	创建存储过程	167
7.2.2	利用向导创建存储过程	172
7.2.3	指定参数	174
7.2.4	设计存储过程	177
7.2.5	从存储过程中返回数据	179
7.3	运行存储过程	182
7.3.1	运行一个存储过程	182
7.3.2	自动运行的存储过程	184
7.4	对存储过程的修改和重新命名	185
7.4.1	修改存储过程	185
7.4.2	存储过程的重新命名	189
7.5	重新编译存储过程	189
7.6	查看存储过程	190
7.6.1	查看存储过程的定义	190
7.6.2	查看关于扩展的存储过程的信息	190
7.7	删除存储过程	191
7.7.1	删除存储过程	191
7.7.2	删除扩展的存储过程	192
7.8	小结	193
第 8 章	SQL Server 中的触发器	194
8.1	触发器同其他维护数据完整性的方法之间的比较	194
8.2	创建触发器	195
8.2.1	使用企业管理器创建触发器	195
8.2.2	利用 Transact-SQL 语句创建触发器	196
8.2.3	设计触发器	203
8.2.4	隐式和显式的空值	206
8.2.5	嵌套的触发器	207
8.3	触发器的修改和重新命名	211
8.3.1	修改触发器	212
8.3.2	触发器的重新命名	214
8.4	查看触发器	214
8.4.1	查看触发器的类型	214
8.4.2	查看触发器	215
8.5	删除触发器	215
8.5.1	使用企业管理器	216
8.5.2	利用 Transact-SQL 语句	216

8.6 小结	217
第 9 章 移动数据	218
9.1 输入数据和输出数据	218
9.1.1 选择输入和输出数据的工具	218
9.1.2 使用数据转换服务向导来复制数据	220
9.1.3 使用 bcp 或者 BULK INSERT 复制数据	220
9.2 数据转换服务 (DTS)	237
9.2.1 了解数据转换服务	237
9.2.2 数据转换服务包	239
9.2.3 DTS 包的保存	244
9.2.4 运行 DTS 包	245
9.2.5 计划 DTS 包	245
9.3 数据转换服务输入和输出向导	247
9.3.1 计划使用 DTS 输入和输出向导	248
9.3.2 使用 DTS 输入和输出向导	252
9.4 DTS 设计器	256
9.4.1 DTS 包的管理	257
9.4.2 为 DTS 包添加功能	257
9.5 备份和恢复	261
9.5.1 计划编制数据库的备份和恢复	262
9.5.2 备份和恢复用户数据库	266
9.6 小结	278
第 10 章 检索数据和修改数据	279
10.1 查询工具和程序设计接口	279
10.1.1 查询工具	279
10.1.2 程序设计接口	281
10.2 访问和修改数据的基本原理	281
10.3 查询的基本元素	283
10.3.1 SELECT 语句的组成部分	284
10.3.2 使用 Select List	286
10.3.3 使用 FROM 子句	292
10.3.4 使用 WHERE 和 HAVING 进行过滤	293
10.4 高级查询概念	299
10.4.1 在选择列表中使用集合函数	300
10.4.2 使用 GROUP BY 对数据行进行分组	302
10.4.3 利用 UNION 进行联合查询	306
10.4.4 子查询的基本原理	309

10.4.5 分布式查询	311
10.5 添加数据	313
10.5.1 利用 INSERT 语句添加数据行	314
10.5.2 用结果集的位置添加数据	320
10.5.3 向插入行中添加 ntext、text 和 image 类型的数据	321
10.5.4 利用批量复制操作来添加数据	321
10.6 修改数据	321
10.6.1 利用 UPDATE 改变数据	321
10.6.2 利用指针修改数据	327
10.6.3 修改 ntext、text 或 image 类型的数据	327
10.7 删除数据	330
10.7.1 使用 DELETE 语句删除数据	330
10.7.2 删除结果集中的数据	334
10.7.3 使用 TRUNCATE TABLE 删除表格中的所有数据行	335
10.8 小结	335
第 11 章 锁定	336
11.1 四个并发性问题	336
11.2 理解 SQL Server 中的锁定	337
11.2.1 锁定的兼容性	337
11.2.2 动态锁定	338
11.3 显示锁定信息	339
11.4 在 SQL Server 中定制锁定	341
11.4.1 死锁	341
11.4.2 定制锁定的暂停时间	343
11.4.3 定制事务隔离的层次	344
11.5 小结	344
附录 1 分布式数据库	345
数据分布	346
并发操作控制	347
查询处理	348
恢复处理	348
附录 2 安装 SQL Server 7.0	349
前提条件	349
设置用户帐户	349
运行 SQL Server 安装程序	350

第1章 SQL Server 7.0 介绍

SQL Server 是基于 SQL（结构查询语言）的客户/服务器数据库。SQL Server 本身是许多不同组件的集合，这些组件包括：SQL Server 本身，SQL Server Agent（SQL 服务器代理），SQL Mail（SQL 邮件系统），以及 Distributed Transaction Coordinator（分布式事务协调器，DTC）。在安装 SQL Server 的时候，在缺省条件下，系统还会自动生成几个数据库，叫做系统数据库。

1.1 SQL Server 体系结构

SQL Server 是一种数据库应用程序，通过核心数据引擎实现数据库服务。为了了解 SQL Server 的工作机制并且能够熟练地应用，首先应该了解 SQL Server 的体系结构。

1.1.1 客户/服务器体系结构

SQL Server 可以设计成两种系统环境，并能够正常使用：

1. 作为双重或多重的客户/服务器数据库系统；
2. 作为桌上型数据库系统。

下面就来介绍一下客户/服务器数据库系统。

客户/服务器系统是指数据库建立在中央计算机上，被作为一个服务器，为多个用户所共享。用户则通过客户或服务器应用程序来访问服务器。

在双重客户/服务器系统中，用户在本地机上运行一个应用程序，也就是客户响应，它通过网络连接到一个正在运行着 SQL Server 的服务器。客户应用程序通过逻辑和编码两种方式进行工作并显示输出给用户，这也叫作瘦客户。

在多重客户/服务器系统中，客户应用程序的逻辑方式在两种场所进行工作。

瘦客户应用程序在用户的本地机上运行并且被集中用来显示结果给用户。

商业逻辑位于一个在服务器上运行的服务器应用程序中，瘦客户从这个服务器应用程序中获得响应。这个应用程序本身能够为多个同时提出要求的用户提供服务，它是开放式的数据连接并且能够在同一个服务器上作为数据库来使用，它还可以通过网络连接到其他正在作为数据库服务器工作的服务器上。

这是一种典型的 Internet 应用程序。举个例子，一个服务器应用程序可以运行在 IIS 上，并能够为成千个通过 Internet 或 Intranet 工作的瘦客户提供服务。这个服务器应用程序使用连接线路同 SQL Server 进行信息沟通。SQL Server 可以被安装在同一台计算机上作为 IIS 使用，也可以被安装在网络中单独的服务器上。

下面介绍一下 SQL Server 在数据的存贮和管理方面的几点优越之处：

1. 每一条数据记录都存贮在中央计算机上，所有的用户都可以使用它。
2. 单独的记录不能存贮在每个客户端上，这样可以消除一个问题，就是用户在使用同一条信息时所造成的安全问题。
3. 商务规范和安全原则可以一次进行定义，并且在所有的用户中平等地使用。
4. 数据的处理在数据库系统中通过存贮过程和触发过程进行，当然也可以在服务器应用程序中进行。
5. 一个相关的服务器可以通过只返回应用程序所要求的数据来进行网络优化。

下面举一个例子来说明一下：如果一个数据库应用需要服务器提供一份在某地区的销售代表名单，那么服务器必须要找到全体雇员的名单。如果这个应用是工作在一个数据库服务器上，那么它会发送如下的指令：

```
SELECT first_name, last_name
FROM employees
WHERE emp_title = 'Sales Representative'
AND emp_state = 'OR'
```

在接受到上面的指令后，相关的数据库会只返回“OR”地区的销售代表的名称，而不是全部雇员的名称。

6. 硬件消耗降低到最小。
- 由于数据不是存贮在每一个客户端上，所以客户不需要提供硬盘空间来存贮数据。客户也不需要本地管理数据的处理能力，服务器也不需要提供显示数据的设备。
7. 服务器可以通过优化配置硬盘的 I/O 能力来达到检索数据的需要，而客户可以优化配置从服务器上达到的数据的格式和显示方式。
 8. 服务器可以安装在相对安全的地方并配备 UPS 电源。这样比对每一个客户都进行保护措施要经济得多。
 9. 对数据库的维护工作可以被简化，因为这些工作可以都在中央处理器上进行。

在大型的客户/服务器系统中，可能会有成千的用户同时连接到 SQL Server 上。SQL Server 具有全面的保护措施来防止有可能出现的问题，比如多个用户同时更新同一条记录。SQL Server 也可以有效地在多用户之间分配有用的资源，比如内存、网宽等等。

SQL Server 应用程序可以在作为 SQL Server 的同一台计算机上运行，这个应用通过 Windows 的 IPC 组件和 SQL Server 进行连接，就象共享内存一样，而不是通过网络来进行连接。这样就允许 SQL Server 能够在一个小的系统中使用，在这个系统中应用程序需要在本地机上保存数据。

图 1-1 就是一个客户/服务器体系结构的示意图。

接下来介绍一下 SQL Server 作为桌上型数据库系统。

SQL Server 作为服务器工作的同时，也可以当作安装在本地机上独立的数据库来使用。SQL Server 能够动态地进行配置，有效地利用客户端的资源，而不需要管理每一个客户端。

当客户使用本地 SQL Server 数据库时，在客户机上运行 SQL Server 数据库引擎，并通过这个引擎管理在客户机上的所有 SQL Server 数据库。数据库的应用程序连接到数据

库引擎的方式与客户端通过网络访问远程服务器的机制是相同的。

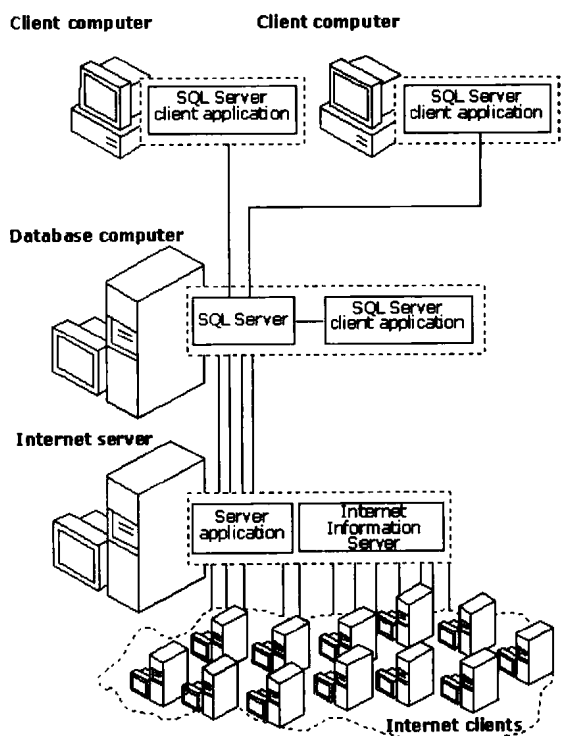


图 1-1 客户/服务器数据库系统示意图

下面的图 1-2 所示的是桌上型数据库系统的示意图。

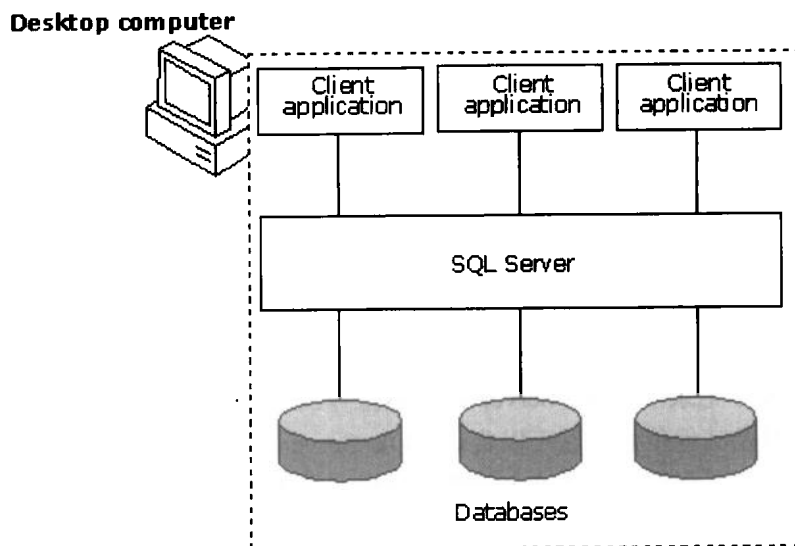


图 1-2 桌上型数据库系统示意图

1.1.2 数据库结构

SQL Server 中的数据存贮在数据库中，数据库中的这些数据有组织地在一个组件中显示给用户。一个数据库在物理意义上也可以被认为是在硬盘上的两个或多个文件。

当使用数据库时，用户可以通过一些组件来进行工作，例如表格、视图、过程和用户等组件。关于数据库在物理上的管理很大程度上是透明的，而只有数据库的管理者才需要对数据库进行物理管理。

图 1-3 很清楚地表示了数据库的体现形式。

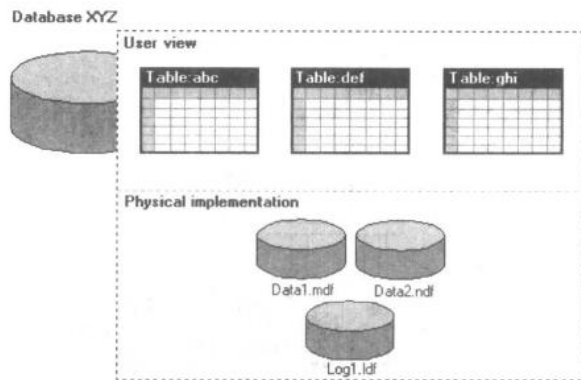


图 1-3 SQL Server 数据库的体现形式

每个 SQL Server 在安装时会产生多个缺省的数据库，叫做系统数据库。SQL Server 有 4 个系统数据库（master、model、tempdb 和 msdb），并且每个 SQL Server 会有一个或多个用户数据库。有些组织结构只有一个用户数据库，在一个数据库中包含了这个结构的所有数据。而有的组织结构对于这个结构中不同的群体有不同的数据库，并且有些时候一个数据库只被一个单独的应用所使用。举个例子，一个组织中可能有一个数据库用来保存销售的信息，另一个数据库用来保存薪水的信息，还有一个数据库用做来响应文档管理的应用等等。用的时候一个应用程序只使用一个数据库，而其他的应用程序会访问多个数据库。

图 1-4 说明了 SQL Server 中的数据库结构。

那么 SQL Server 建立的系统数据库究竟是做什么的呢？下面就先进行简单地介绍。

1. master 数据库

master 数据库中记录了 SQL Server 最基本操作的使用信息。这个数据库包含用户帐号、系统配置信息及其他信息，它可以告诉 SQL Server 其他用户生成的数据库的位置。如果这个数据库发生了问题，那么这个问题就是很严重了。所以在实际的工作过程中应该经常备份 master 数据库。

2. model 数据库

model 数据库是 SQL Server 中可能需要经常修改的系统数据库。在建立系统数据库

时, SQL Server 首先要复制 model 数据库。如果服务器上的所有数据库都需要有某些对象, 比如说是对某项工作很重要的数据, 就可以将这个数据加入到 model 数据库中去, 这样在建立新的数据库时, 这个数据就会随着数据库一起建立了。

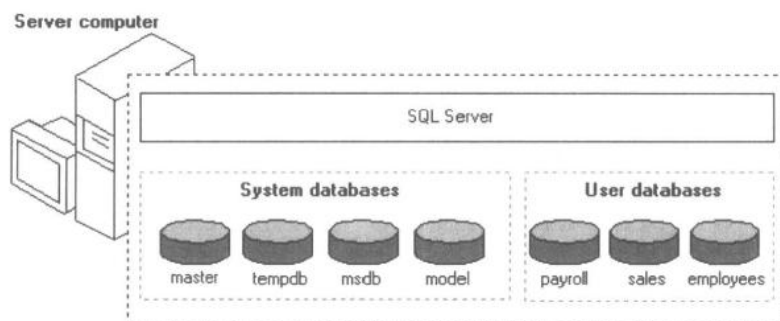


图 1-4 SQL Server 的数据库结构

3. tempdb 数据库

tempdb 数据库是 SQL Server 存放临时工作表格和临时存储过程的数据库。这个数据库在每次启动 SQL Server 时会自动重建。

4. msdb 数据库

msdb 数据库中存储处理作业和警告所需的信息。用户不能直接修改 msdb 数据库, 而需要通过接口生成、修改和删除其中的 SQL Server Agent 对象。

在 SQL Server 中当多个用户同时访问服务器上的数据库时, 不需要运行多个 SQL Server 的复本。因为 SQL Server 能够控制上千个用户同时访问同一个服务器上的多个数据库。SQL Server 使得服务器上的所有数据库对于连接服务器的用户都有效, 当然这些用户需要满足安全原则。

在与 SQL Server 进行连接时, 这个连接只能访问相关的数据库, 这个数据库叫做当前数据库。用户经常访问的数据库可以由系统管理员定义为用户的缺省数据库, 但这样做并不影响用户通过连接选项访问其他的数据库。用户可以通过使用 Transact-SQL 命令从一个数据库切换到另外一个数据库, 或者可以使用 API 函数来进行转换。

SQL Server 7.0 版本允许用户将数据库从一个服务器上删掉后, 移植到另外的一个服务器上, 或者是重新绑定回原服务器。

1.1.3 服务器结构

服务器是 SQL Server 中的一个组件, 它从客户端接受 SQL 语句的指令并执行这些指令。在本书后面的内容中会对服务器进行详细地讨论。

1.1.4 管理体系结构

SQL Server 的每一个版本都在寻求操作的自动化以及消除在前一版本中需要数据库管

理员进行的重复操作。因为数据库管理员在任何一个有关数据库的场所中，都是一个接受过数据库知识系统培训的人员，如果将他的精力消耗在某项重复的工作上，是一种对人力资源的浪费，而 SQL Server 能够使数据库管理员有更多的时间花在数据库的设计和应用上。

SQL Server 的管理机制有着下列的特性：

1. SQL Server 7.0 数据库服务器通过对资源更加动态地使用来减轻数据库管理员的工作量。服务器在需要的时候可以自动地寻求资源，比如内存，硬盘等等，在使用以后可以自动地将其释放。

2. SQL Server 提供了一套图形处理工具，这就使得数据库管理员在进行管理工作时更加简单和有效。

3. SQL Server 提供了一整套服务措施，可以使数据库管理员制定重复性工作的自动执行时间表。

4. SQL Server 可以提供处理意外情况的服务器，或者用来发送 E-mail 和向管理员提供信息的服务器。

5. SQL Server 同时发布了管理应用程序接口 (APIs)。这些 APIs 能够支持 SQL Server 的全部管理工作。这就使得应用程序的开发者可以使用 SQL Server 作为数据的存贮器来完成对用户信息的保护。

1.1.5 复制体系

复制是分配数据和存贮过程中的一项重要技术。SQL Server 中的复制技术使得用户可以制作数据的副本，将这些副本移植到其他位置，并且这些副本能够自动与原来的数据同步，因此使得所有的副本有着相同的值。复制可以在同一服务器上的数据库之间进行，也可以在通过 LANs、WANs 或者 Internet 连接的不同的服务器上的数据库之间进行。

SQL Server 能够支持：

1. 复制不同种类的数据。
2. 复制不同种类数据的标题化界面。
3. 调用复制代理器的界面。
4. 安装和监控复制的 SQL-DMO 界面。

1.1.6 数据库应用程序开发体系

数据库的应用程序可以通过两种方式访问数据库：

1. API

数据库的 API 定义了如何编写应用程序的代码，来连接数据库并传送指令给数据库。一个对象模板 API 通常是独立的语言并且定义了一系列的对象、属性和接口，而 C 或 VB 中的 API 定义的是一系列其应用程序中琐事用的函数。

SQL Server 支持很多的 API 来创建数据库应用程序，这些 API 有：

- (1) ActiveX Data Objects (ADO);

- (2) OLE DB;
- (3) 开放型数据连接 (ODBC), 通过 ODBC 创建的 API 对象: 远程数据访问对象 (RDO) 和数据访问对象 (DAO);
- (4) C 语言中绑定的 SQL;
- (5) C 语言中嵌入 DB-Library;
- (6) VB 中嵌入 DB-Library。

2. 数据库语言

数据库语言定义了发送到数据库指令的语法结构。这些通过 API 传送的指令使得应用程序能够访问并且修改数据, 这些指令也可以允许应用程序生成和修改数据库中的对象。

SQL Server 中所支持的数据库语言是 Transact-SQL。Transact-SQL 这种语言不但遵从 SQL-92 标准语言中的基础部分, 还支持这一标准语言高级部分中的一些属性, 并支持一些 SQL-92 标准语言中的很有效的扩展部分。关于这一语言在这里只是提一下, 如果有兴趣可以参阅有关书籍。

上面所介绍的是关于 SQL Server 的一个简单的介绍, 曾经使用过或接触过旧版本的 SQL Server 的人会发现, SQL Server 7.0 较前版本增添了许多新的内容。先面就来介绍一下 SQL Server 7.0 中的新内容。

1.2 SQL Server 7.0 中的新内容

SQL Server 7.0 在许多方面都进行了改进并增添了新的内容, 下面就分别介绍一下。

1.2.1 服务器的改进和新特性

1. 减少了管理所用的费用和投入

- (1) 在内存使用方面, 内存管理器同服务器中内在的组件合作来优化使用内存。
- (2) 在硬盘方面, 文档的生成和压缩是自动进行的。
- (3) 自动更新系统中的统计表, 自动生成统计表。
- (4) 自动重新编辑计划。
- (5) 减少了配置工作。一些设置可以自动地进行配置, 如: 连接、锁定和打开对象。

许多服务器的设置已经改进和简化, 例如, 在缺省状况下, 服务器动态地调整内存和锁定使用资源。在数据库中可以增加分配的使用资源, 并在不再需要的条件下释放掉。而前版本的 SQL Server 需要人工的进行这些方面的配置。

数据库安装在操作系统当中, 而不是位于 SQL Server 的设备上。因此当需要生成数据库及其文档时, 只需要使用简单的 CREATE DATABASE 语句或使用 SQL Server 企业管理器来生成新的数据库, 修改 (变更) 已有的数据库。

数据库文档能够从原始的大小自动的扩充。在 SQL Server 中, 定义一个数据库文档的时候, 同时可以指定这个文档的的增量。每次这个文档的空间被占满后, 会自动地按照事先设定的增量对文档的大小进行扩充。SQL Server 中数据库的大小可以指定一个最大尺