返 回 总 目 录

SQL Sever 7.0篇

第一章

进入SQL Server

本章导读

在安装完SQL Server 7.0之后，用户将能够在启动菜单上看到Microsoft SQL Server 7.0程序组。这个程序组包括了许多有用的图形界面的工具，便于用户管理SQL Server。

本章将逐一地向读者介绍该程序组中的应用程序的功能、用法，以及相应的SQL Server背景知识。读者在今后的学习中将不断地利用这些工具来管理和维护数据库。

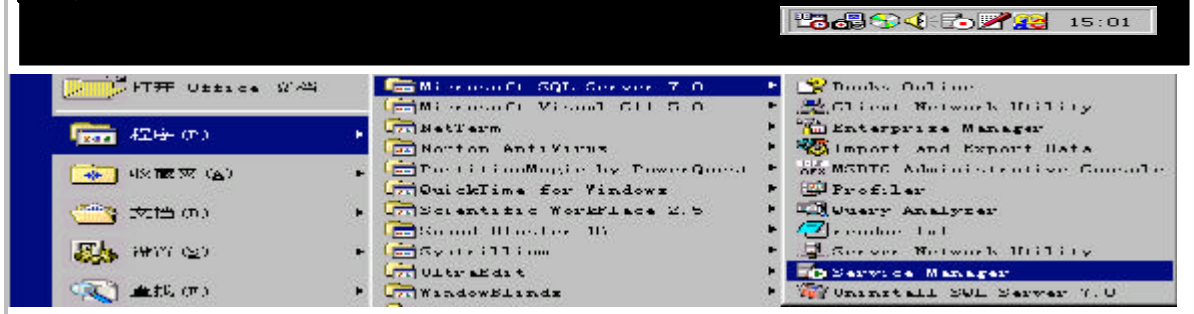
本章将介绍以下应用程序工具：Service Manager、Profiler、Server Network Utility、Client Network Utility、Enterprise Manager、Query Analyzer。

例

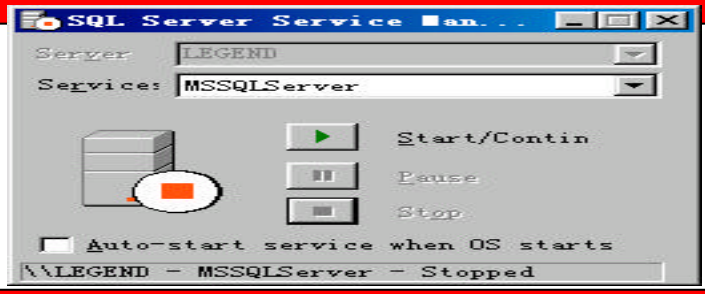
学会使用 Service Manage 启动、关闭服务器

解

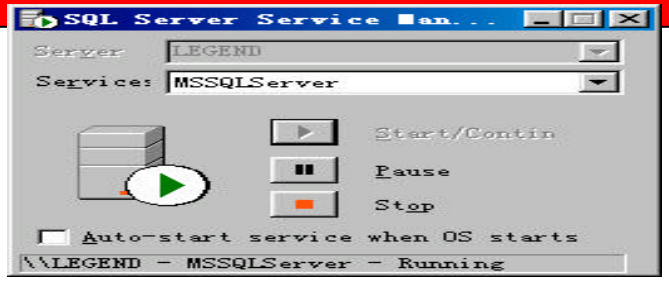
1



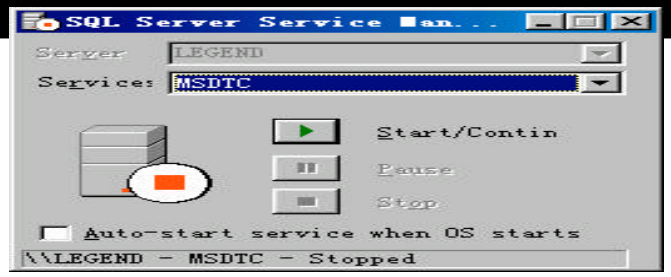
2 Service Manage 的程序窗口如下图所示。在 Services 菜单里可以选择 MSDTC、MSSQLServer、SQLServerAgent。



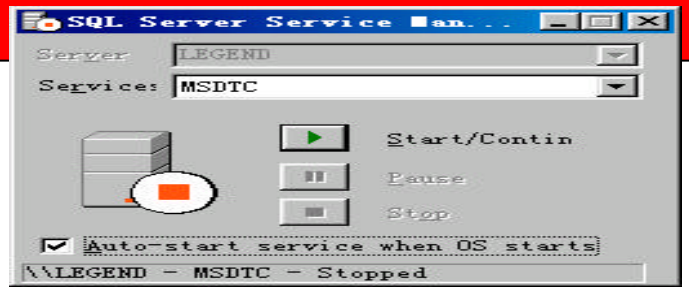
3 选择 MSSQLServer，按下 Start/Continue（启动、继续）。这时，SQL Server 服务器就被启动了。Service Manage 变成下图的状态。



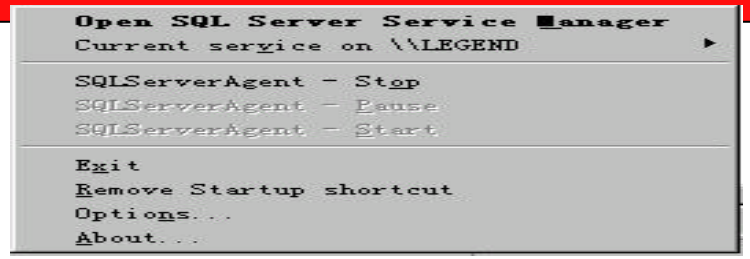
4



5 在上述图中选中复选框 Auto-start service when OS Starts。



6 右键单击任务栏上时钟区的 Service Manage 图标可以弹出下图菜单，通过下图菜单同样可以完成 Service Manage 的工作。



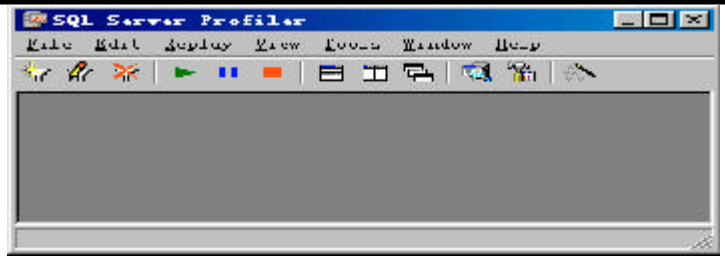
7 通过本例学会了如何打开 Service Manage，并且利用它完成启动、停止、暂停 SQL Server 服务器的操作。

例

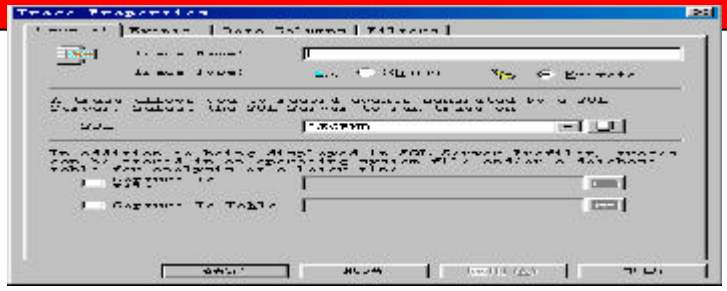
学会使用 Profiler
创建跟踪

解

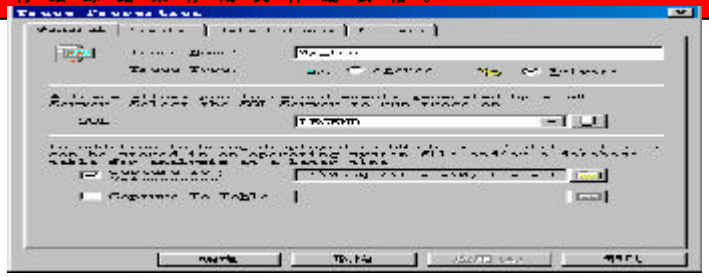
1



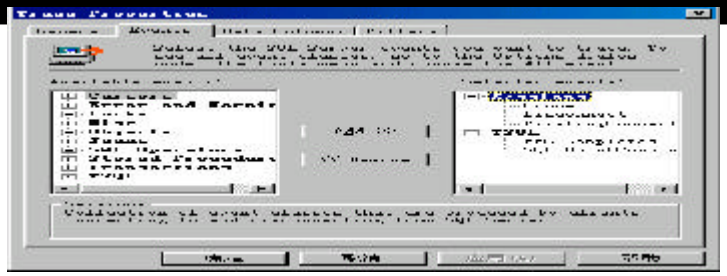
2 创建跟踪。通过 File 菜单中的 New－〉Trace 或者工具栏上的 New Trace 按钮直接打开创建跟踪对话框：



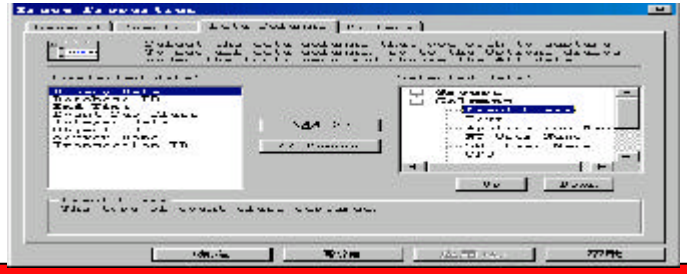
3 该对话框有四页：General、Events、Data Columns 和 Filters。首先是 General 页。选择 Private 属性，Trace Name 为 My_Trace，SQL 为要跟踪的服务器名。可以将跟踪结果存成文件或表格。



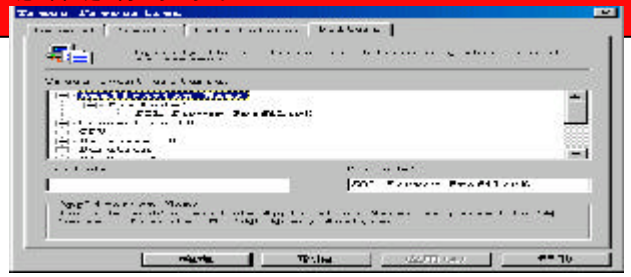
4



5 Data Columns 页面用来指定需要纪录所跟踪事件的那些属性。Unselected data 栏中为被选中的属性，Selected data 栏中唯一被选中的属性。Add、Remove 键用来添加、删除选择的属性。



6 Filter 页面用来限制所跟踪的事件。如用户名与指定名称相匹配，cpu 占用时间在指定的范围之内等等。



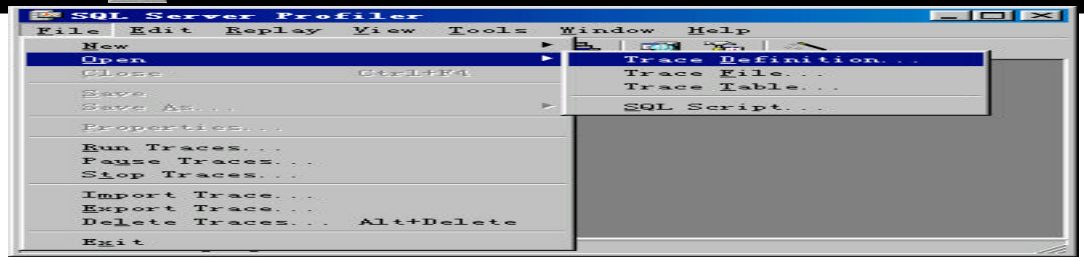
7 完成上述四个页面后，单击确定就创建了一个跟踪。在后面，将继续讲述如何利用跟踪来监视系统的行为和性能。

例

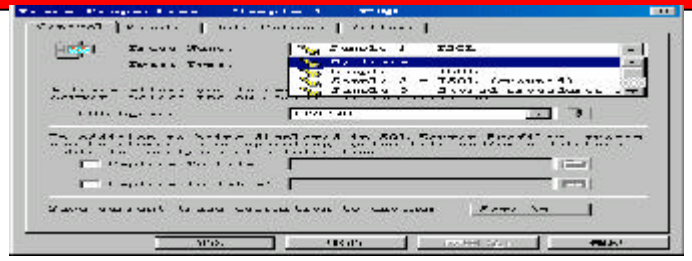
学会使用 Profiler
修改跟踪、跟踪事件

解

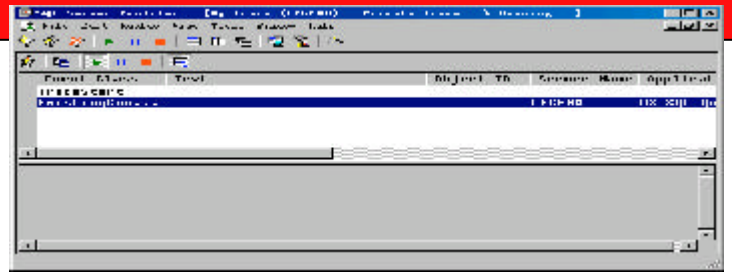
1



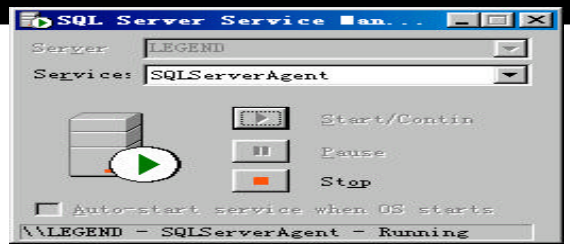
2 接着，将会弹出 Trace Property 对话框如下图所示。在 Trace Name 中选择 My_Trace。单击确定键，就选定了上例中所定义的跟踪。



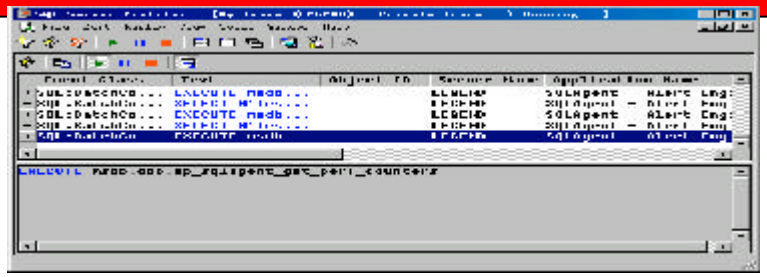
3 打开后的跟踪窗口如下图所示。后面将观察 Profiler 是怎样跟踪事件的。



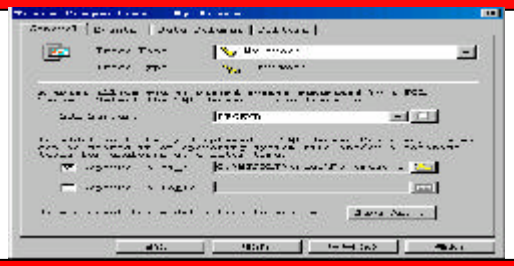
4



5 这时我们再回头看看 Profiler 的跟踪窗口里究竟发生了什么？Profiler 窗口如下图所示。



6 可以对当前跟踪的内容进行保存。可以保存到数据库的某个表格中，也可以保存到文件中。单击 properties 按钮。选择 Capture to File，指定存放跟踪文件的路径。



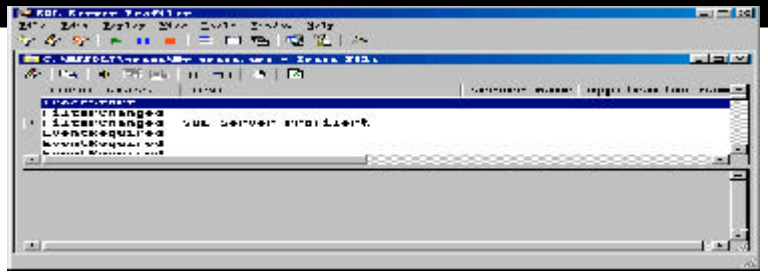
7 本例中，学会了如何利用 Profiler 来跟踪 SQL Server 中发生的事件。并且学会了如何利用 Property 对话框修改跟踪的定义，如何对跟踪的结果进行保存。

例

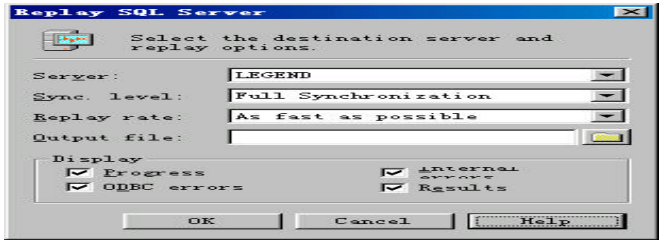
学会使用 Profiler 再现跟踪事件

解

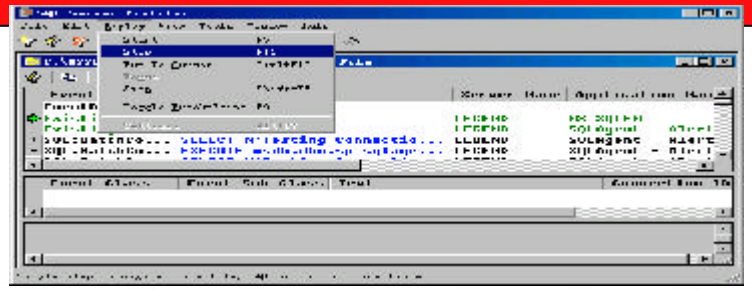
1



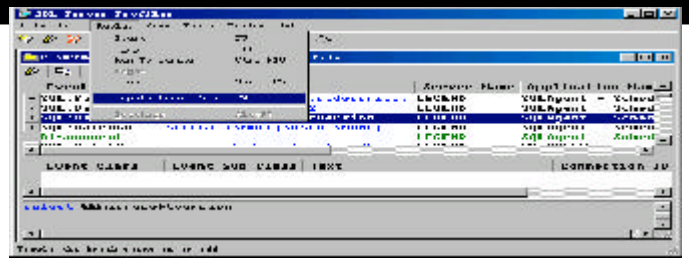
2 对再现进行设定。选择 Replay 菜单中的 Settings 将弹出以下对话框。可以设定同步级（Sync. Level）、再现速率（Replay rate）、输出文件（output file）、显示内容（进度、ODBC 错误、内部错误、结果）。设定完按确定。



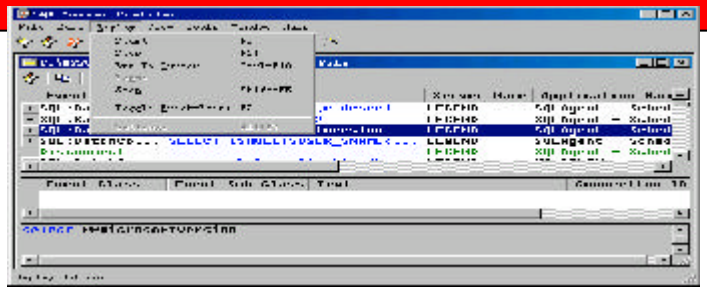
3 单步执行，选择 Replay 菜单中的 Step 项。接着在弹出的 Settings 对话框中设定单步的属性，再按确定。Profiler 就会对该跟踪执行一次事件。



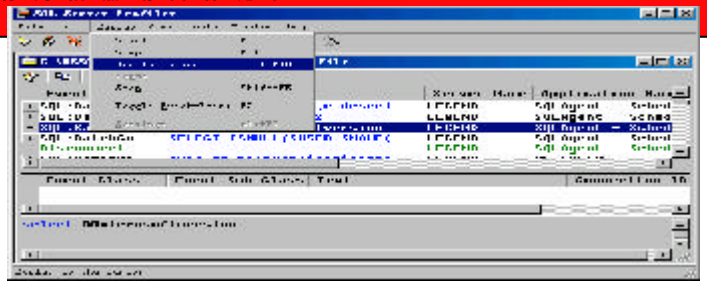
4



5 运行至当前光标的所指事件。首先在跟踪窗口中选择某个事件，然后从 Replay 菜单中选择 Run to Cursor。



6 连续执行。在 Replay 菜单中直接选择 Start（F5），就会按照设定的 Settings 选项一直执行到段点或文件尾。



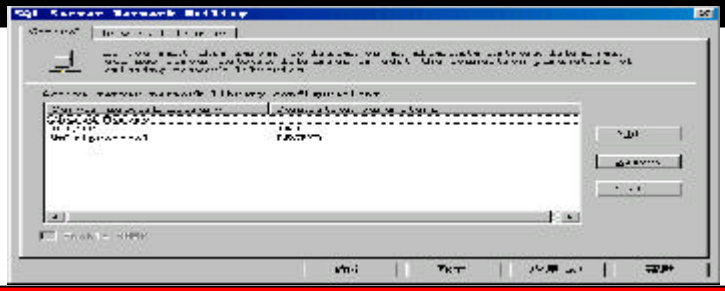
7 通过本例，我们学会了 Profiler 如何通过跟踪文件来再现跟踪事件的。

例

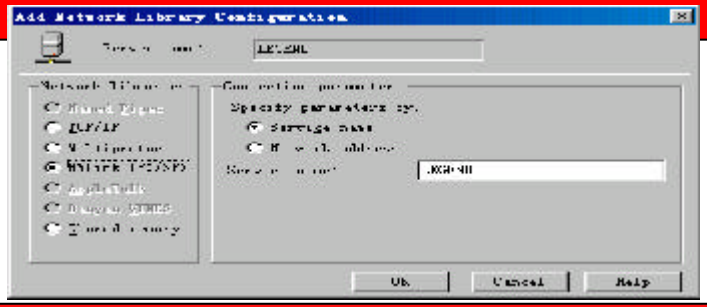
学会使用 Network Utility 设置 Net_Library

解

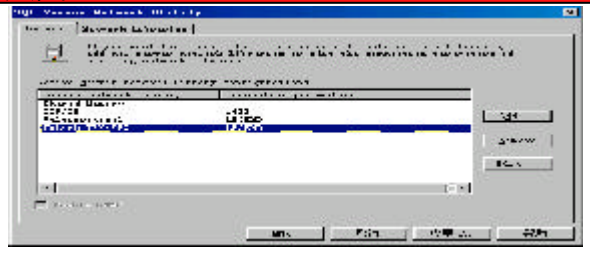
1



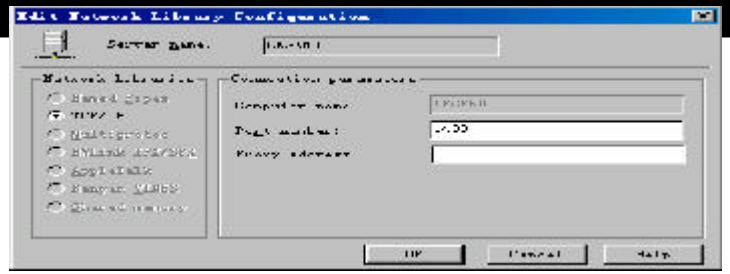
2 在 General 页面中单击 Add 键，可以添加网络协议。如下图：



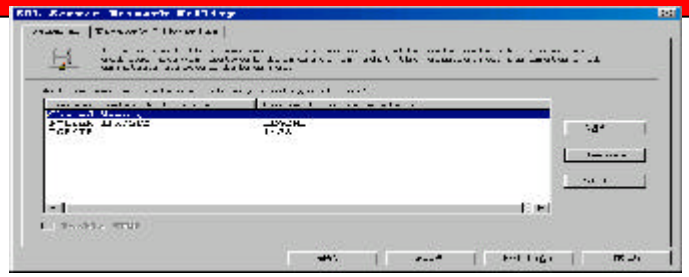
3 这里我们在原有网络协议的基础上添加 IPX/SPX 协议。在 Connection parameters 中对该协议进行设定。然后，按确定键。则 SQL Server 就会对该协议进行侦听。



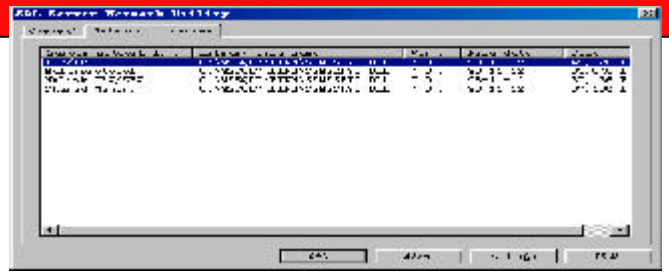
4



5 用户还可以指定某网络协议失效，即SQL Server不对该协议侦听。先指定某个网络协议再单击Remove键。如指定Multiprotocol，再单击Remove。窗口变成下图所示。



6 单击Net_Library可以察看网络协议所对应的动态链接库文件。界面如下图所示：



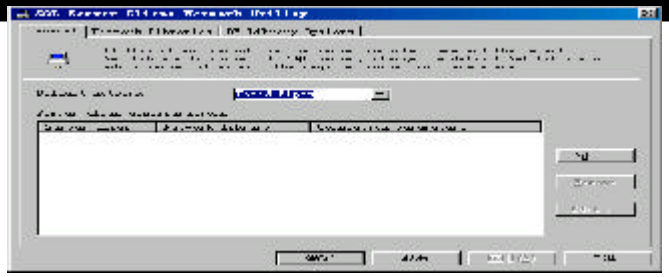
7 在本例中我们学会了如何利用Network Utility来添加、删除、编辑SQL Server使用的网络协议和察看网络协议对应库文件的信息。

例

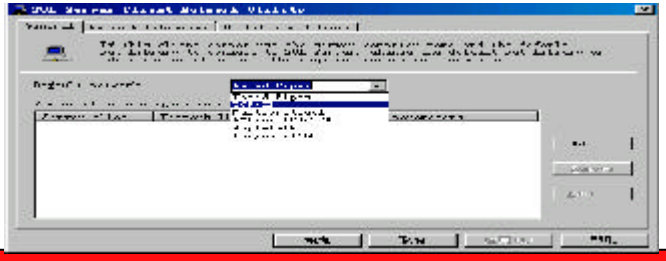
学会使用 Client Network Utility

解

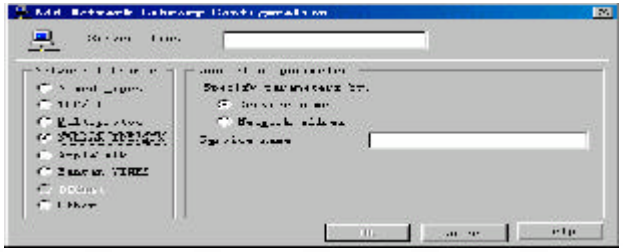
1



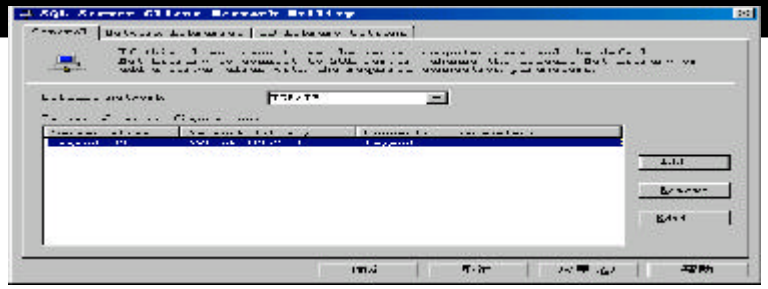
2 在 Default network 菜单栏中，可以修改缺省网络协议。如改成 TCP/IP。建议一般情况下不要修改此项，除非连不上 SQL Server 服务器。可以尝试以下其他的协议。



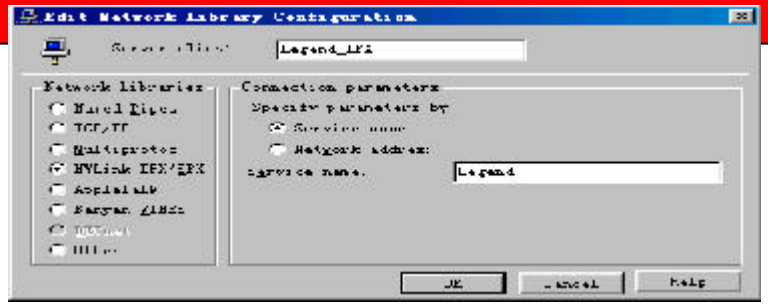
3 用户还可以为连结某个专门的服务器进行别名设定。在上图中按 Add 键，弹出下图对话框。在 Server alias 中输入服务器别名，Service name 中输入 SQL Server 服务器对应该协议设定的 IPX Service name。再单击 OK。



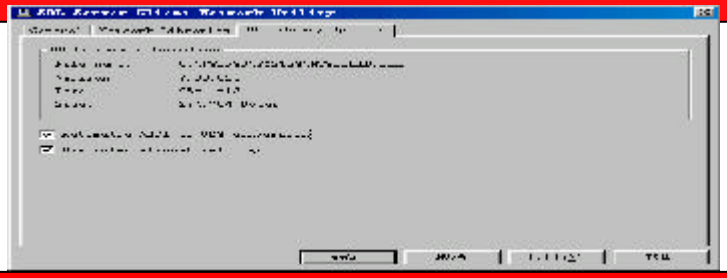
4



5 在弹出的对话框中可以修改该服务器别名的设置。再按 OK 键。



6 单击 DB Library Option 可以察看 DB Library 的属性，并且进行相应的设置。界面如下图所示。



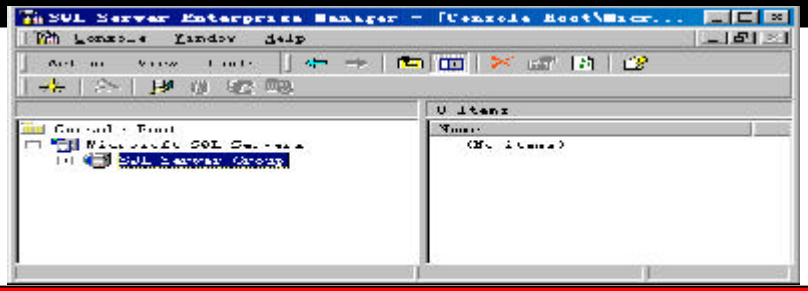
7 通过本例，学习了如何使用 Client Network Utility 设置客户方的网络协议以与指定的 SQL Server 服务器相连。

例

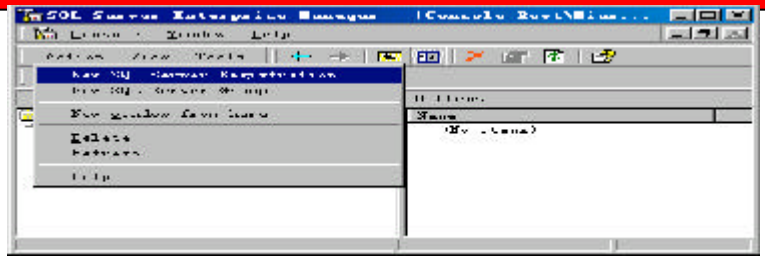
学会使用 Enterprise Manager 注册服务器

解

1

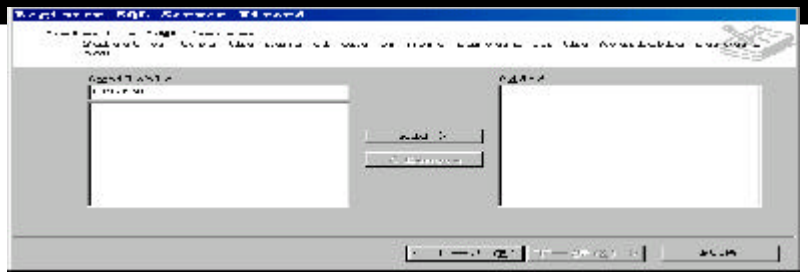


2 从菜单 Action 中选取 New SQL Server Registration 或者用鼠标右键单击 SQL Server Group 如图，选择同上命令，或者单击按钮 。

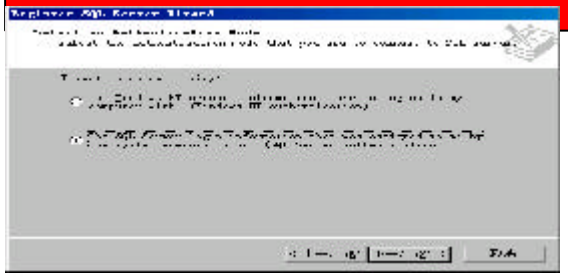


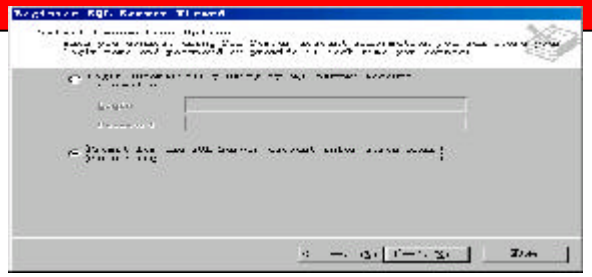
3 将会弹出 Register SQL Server Wizard，如下图所示，接下来只需按照这个向导的提示去做就可以完成注册了。单击下一步继续。

4

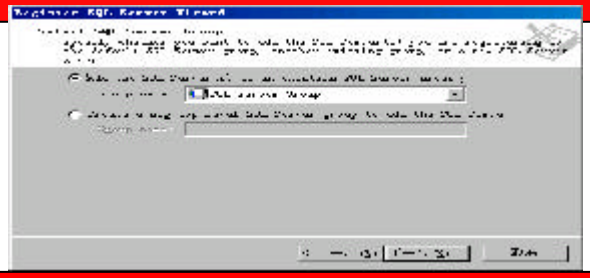


5 下面两幅图中分别选择连接服务器所用的授权模式和登录时所用帐号口令。都选择第二个，单击下一步继续。





6 最后一步选择将新注册的服务器添加到已有的某个服务器组中或者另外新建一个服务器组。选择第一个。按下一步继续就可以完成注册了。



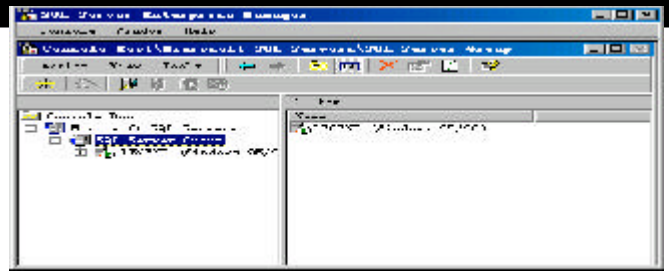
7 本例中，学会了如何在 Enterprise Manager 中注册一个新的 SQL Server。Enterprise Manager 可以同时管理许多个 SQL Server，功能十分强大。今后我们还要经常与它打交道。

例

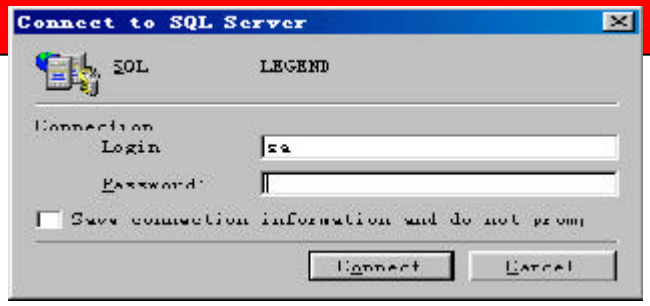
学会使用 Enterprise Manager 登录服务器并修改密码

解

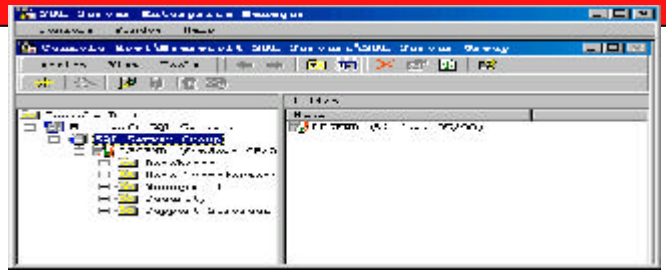
1



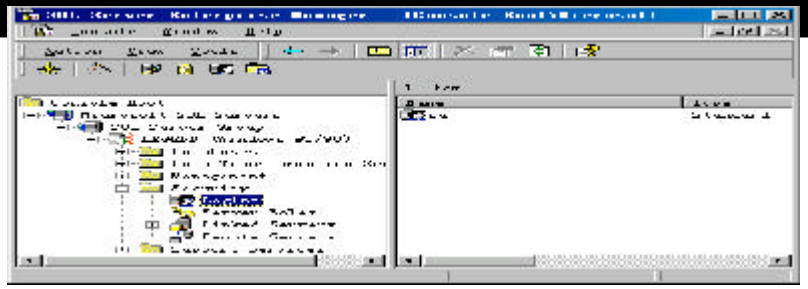
2 连接数据库，只需单击左栏中的 LEGEND 项，就会弹出以下对话框。



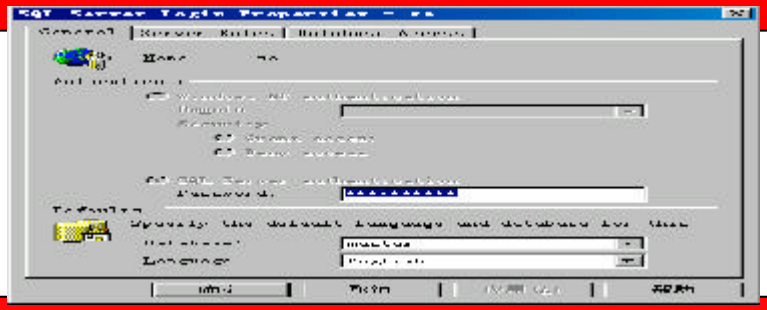
3 在上图中按下 Connect 键，如果 SQL Server 已被启动，就可以连上服务器了。连上后 Enterprise Manager 的界面将会变成下图所示：



4



5 在右栏选中 sa，按右键，在弹出菜单中选择 Properties，或者选中 sa 后直接单击工具栏上的 按钮，会弹出下图对话框：



6 直接在 Password 栏中输入新的密码，Enterprise Manager 会弹出确认对话框，在此输入后，按 OK，密码就被修改了。

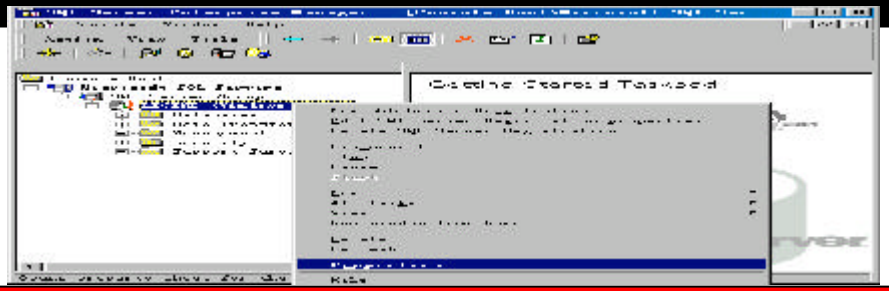
7 通过本例，我们初次认识了 Enterprise Manager，知道了如何注册新的服务器，如何连接服务器，如何修改用户密码。今后还将发现这只是 Enterprise Manager 强大功能的很小一部分。

例

学会使用 Enterprise Manager 设置内存大小

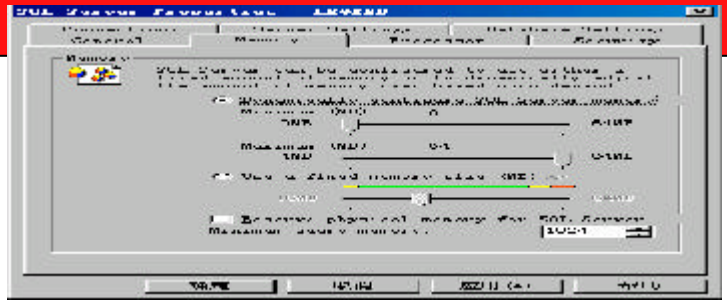
解

1

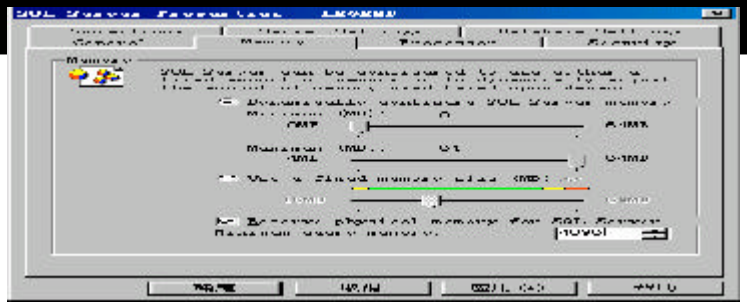


2 弹出的属性对话框如下图：

3 单击 Memory 页面。界面如下图所示：



4



5 单击 Security 页面，界面如下图：

6 在 Audit Level 中选择 All。然后按确定完成设置。

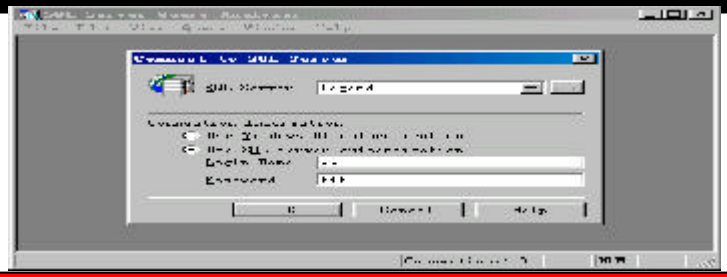
7 通过本例，学会了如何在 Enterprise Manager 中设置服务器属性。至于其他属性，设置的方法大同小异。由于篇幅有限，不能一一枚举。

例

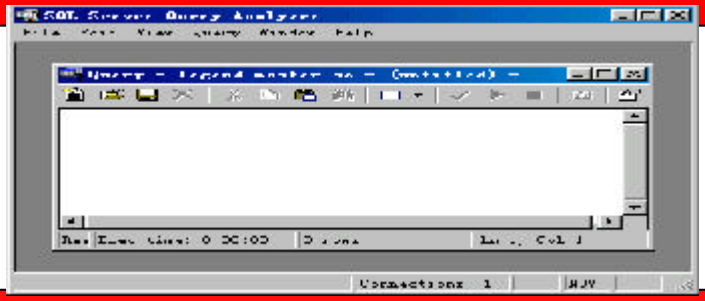
学会使用 Query Analyzer 执行 Transact SQL 语句

解

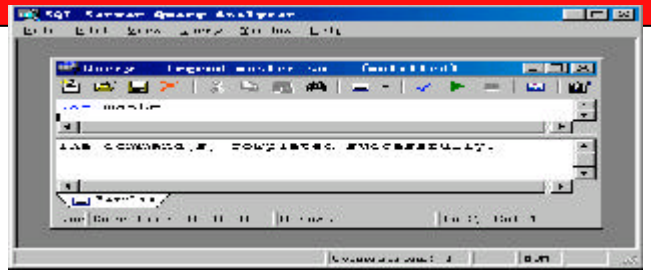
1



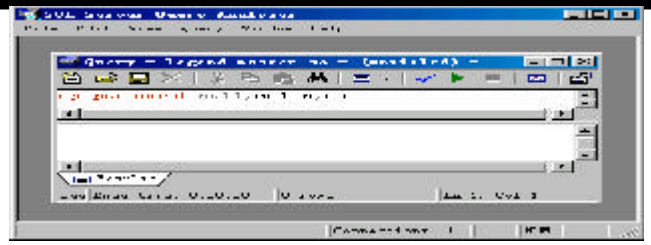
2 分别输入用户名和口令，再按 OK，连接数据库。如果输入正确且用户合法，就可以与数据库连上。界面如下图：



3 在 Query 窗口中输入 Transact-SQL 语句，输入完在 Query 菜单中选择 Execute（F5），或者按下 ▶ 键，就可以执行该语句。例如：输入 Use master

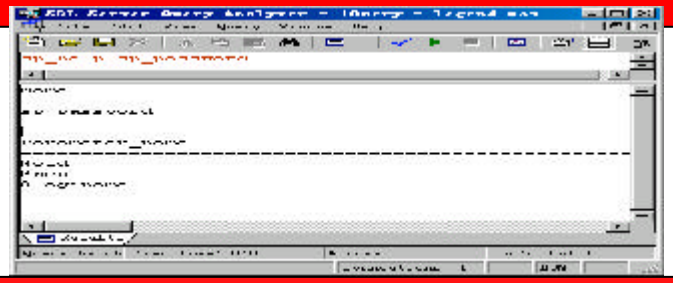


4



5 再单击 ▶，执行结果就出现在输出窗口中，如下图：

6 如果对某个存储过程的用法不太清楚，可以执行 sp_help 察看帮助，如下图：输出窗口中说明 sp_password 有三个参数,旧口令、新口令、帐号。



7 通过本例学会了如何在 Query Analyzer 窗口中执行 Transact SQL 语句。实质上，SQL Server 任何数据库操作都是通过 Transact SQL 语句描述的。Enterprise 完成的功能，在内部也是先翻译成 SQL 语句再执行的。

返回总目录

第 二 章

管理SQL Server

本章导读

Microsoft SQL Server 管理程序和附带工具是用来帮助系统管理员管理任务和监视服务器的性能和活动的。

本章在上一章的基础上，将介绍一些日常的和重要的管理 SQL Server 的功能和方法。这是对上一章学习的巩固和深入。同时，读者也将通过本章了解到有关 SQL Server 的更加深入的主题。

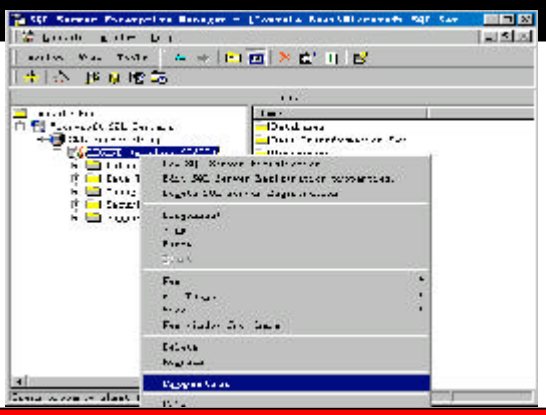
本章的主要内容包括：SQL Server 的启动、暂停、关闭; 备份和恢复数据; 管理服务器端;设定自动执行任务; 管理用户权限; 监视 SQL Server 的行为; 在 SQL Server 中使用其他工具等具体内容。

例

SQL Server 的启动、暂停、关闭
SQL Server 的自动和手动启动

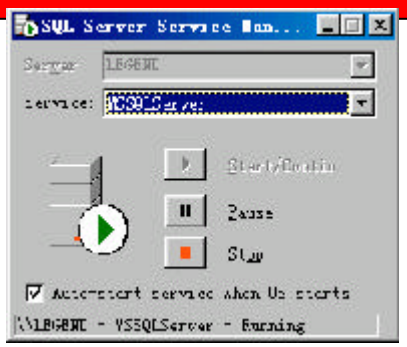
解

1

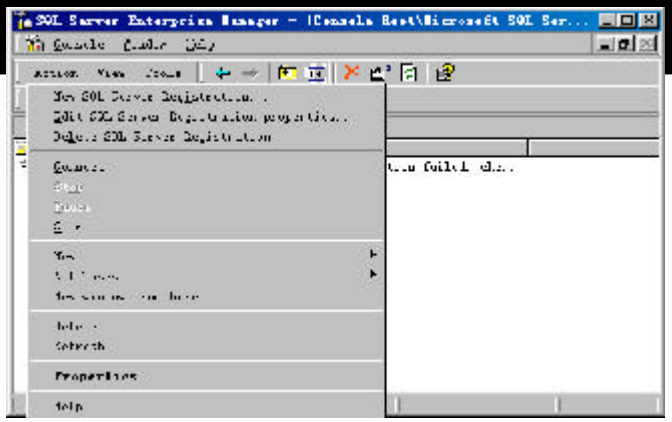


2 弹出如下对话框。在 Windows NT 下，可以选择 Autostart SQL Server。再单击确定。

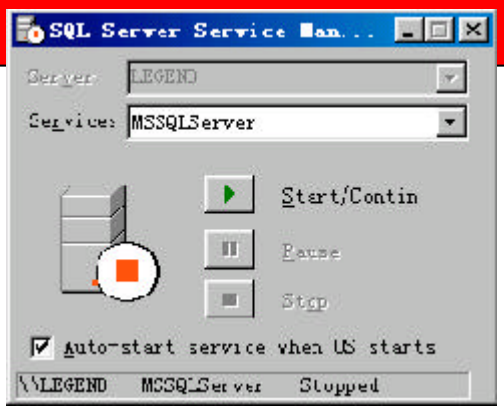
3 在 Service Manager 中设置的方法，前面已经提到，即在 Service Manager 中选择 Auto-start when OS starts。



4

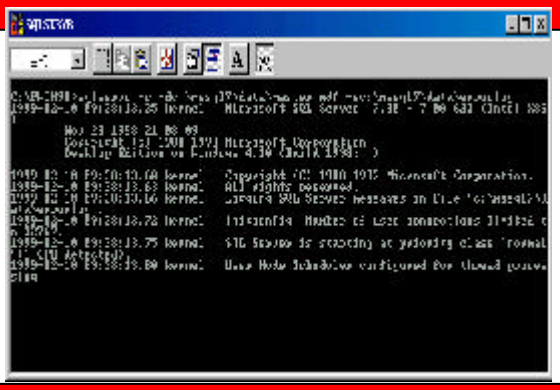


5 在 Service Manager 中选择 Start/Contin MSSQLServer。单击 按钮。



6 可以在命令行方式下用 sqlservr 命令启动 SQL Server，本例中采用命令为：

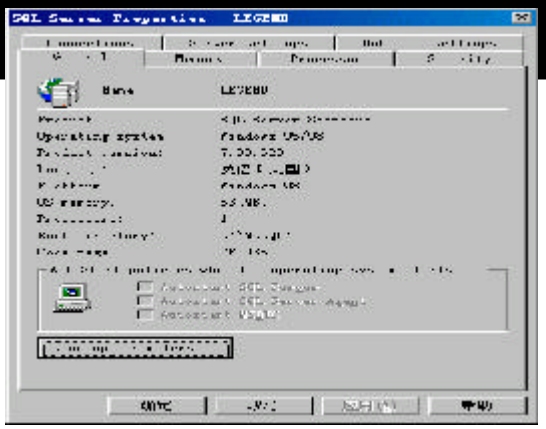
sqlservr -c -dc:\mssql7\data\master.dat -ec:\mssql7\log\errorlog



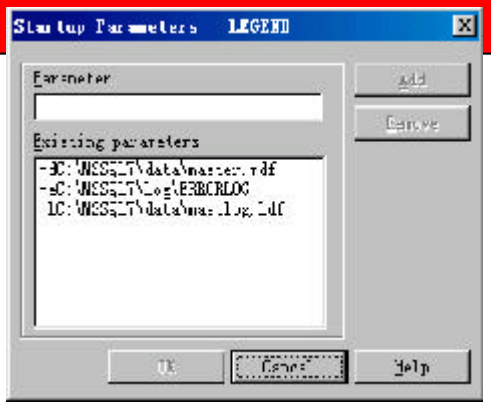
7 通过本例，学习了如何自动和手动启动 SQL Server 的多种方法。

例 SQL Server 的启动、暂停、关闭 设置单用户模式、最小设置 解

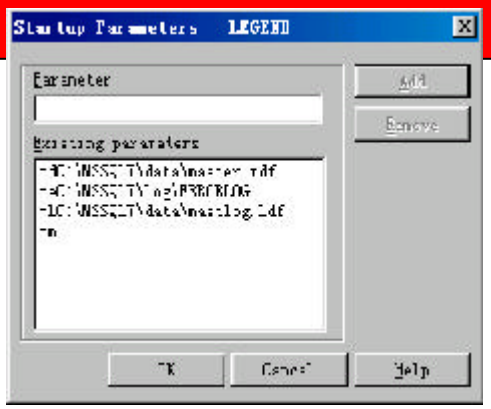
1



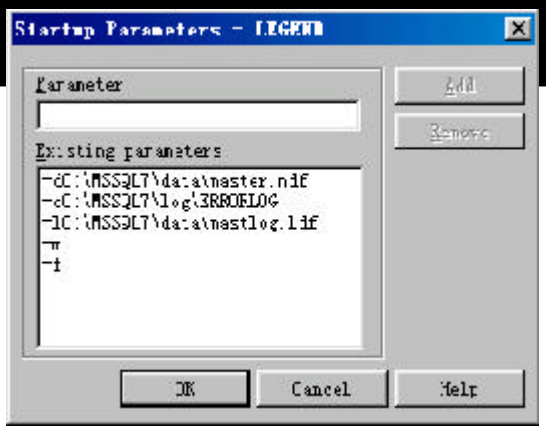
2 在属性对话框中，选择 Startup Parameters 按钮，打开对话框如下图



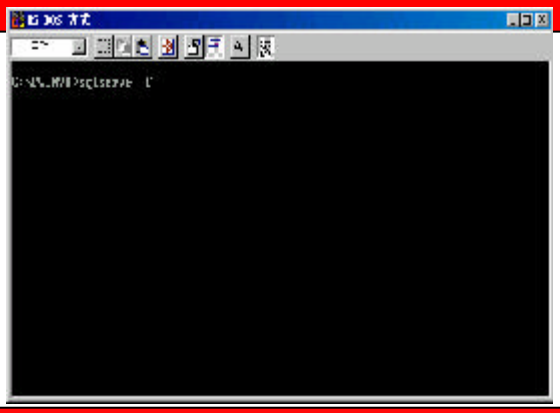
3 在 Parameter 中输入-m，并且点击 Add 按钮。再按 OK。



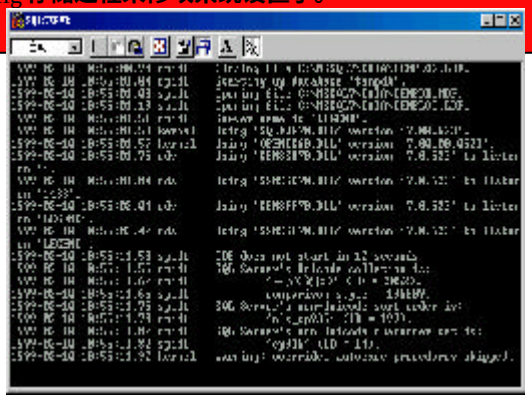
4



5 前面说到两种设置，也可以在命令行方式下设置。命令为 sqlservr –m 或者 sqlservr –f。



6 运行上图中的命令后，SQL Server 将按照最小模式启动。之后就可以运行 sp_config 存储过程来修改系统设置了。



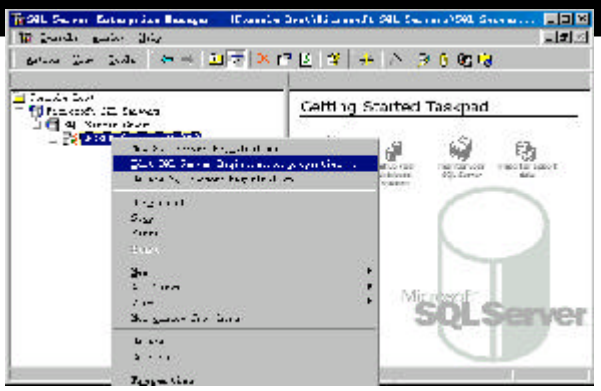
7 通过本例，学会了如何将 SQL Server 设置成单用户模式和最小模式。后面将会学习如何进行 SQL Server 的设置。

例

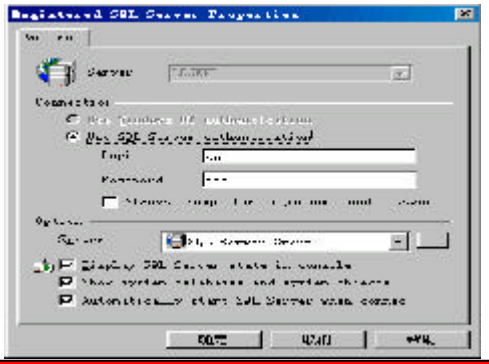
SQL Server 的启动、暂停、关闭
SQL Server 的登录

解

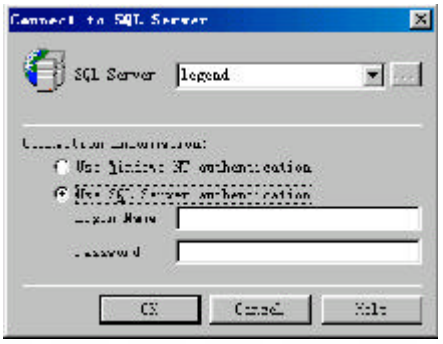
1



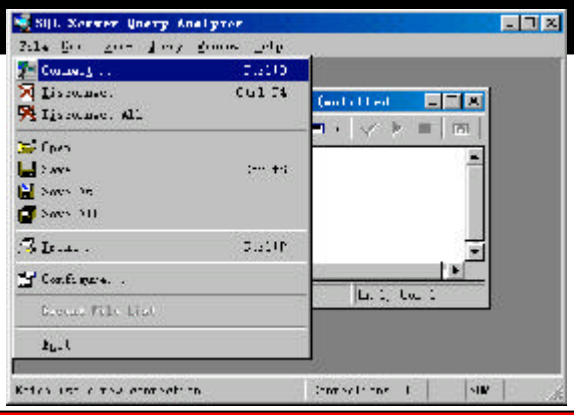
2 弹出对话框中，在 SQL Server authentication 中设置用户名和口令，再按确定。以后每次通过 Enterprise Manager 连接该服务器时都自动用该用户和口令登录。如果需要修改，可以再到该对话框中设置。



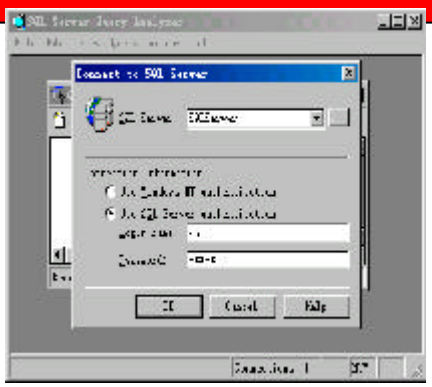
3 Query Analyzer 登录：进入 Query Analyzer 后自动会有下图对话框，选择要连接的服务器名和连接方式，如果用 SQL Server authentication 则要输入用户名和口令，再按 OK。



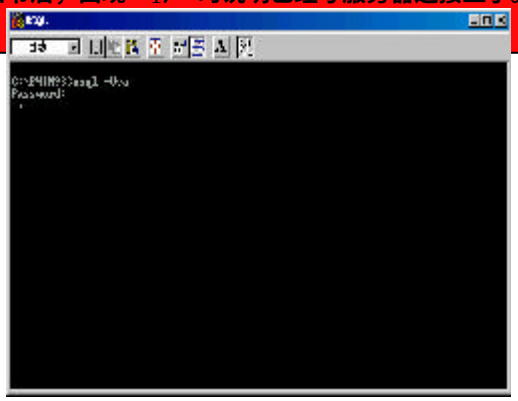
4



5 如下图所示：在新弹出的对话框中可以按照新的用户名和口令建立新的连接服务器。输入新的服务器名、用户名和相应的口令，再按 OK。



6 命令行方式 osql 登录：运行 osql –Uxx（xx 用的用户名代替），在 Password 中输入口令后，出现‘ 1〉’时说明已经与服务器连接上了。



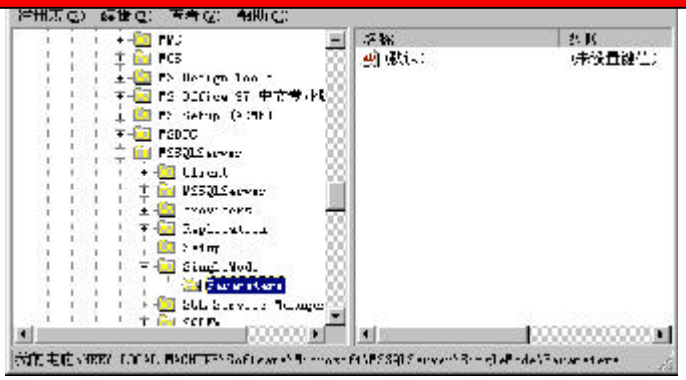
7 本例中，通过几种工具学会了如何与服务器建立连接。

例 SQL Server 的启动、暂停、关闭创建启动属性 解

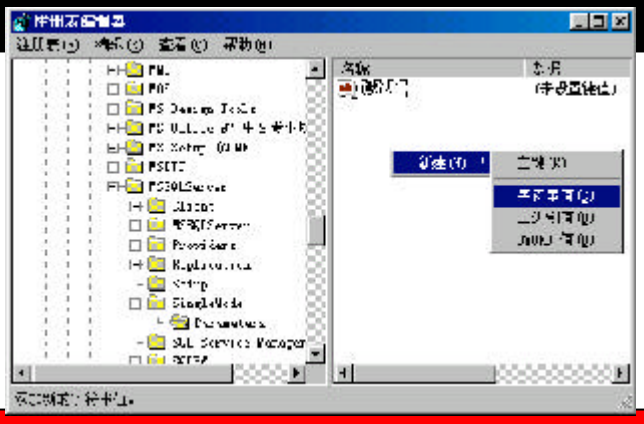
1

2 找到 HKEY_LOCAL_MACHINE －》 SOFTWARE －》 Microsoft －》 MSSQLServer－》主键。选择菜单编辑－》新建－》主键。

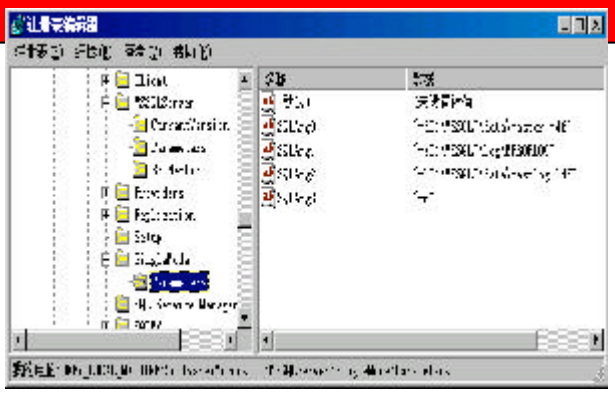
3 输入新建主键名称为 SingleMode，按照同样的方法，在 SingleMode 中新建新的主键 Parameter，如图所示：



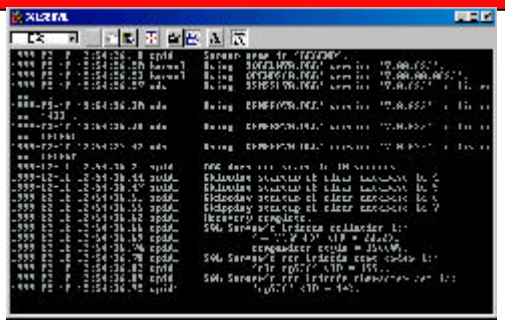
4



5 建立了四个参数的启动参数项后，退出重新启动 Windows，该注册库被修改的内容就发生作用了。



6 以后启动 SQL Server 时，要选择用上述生成的启动选项来替换缺省选项时可以在命令行输入如下命令：sqlservr -c –sSingleMode。运行结果如下图：



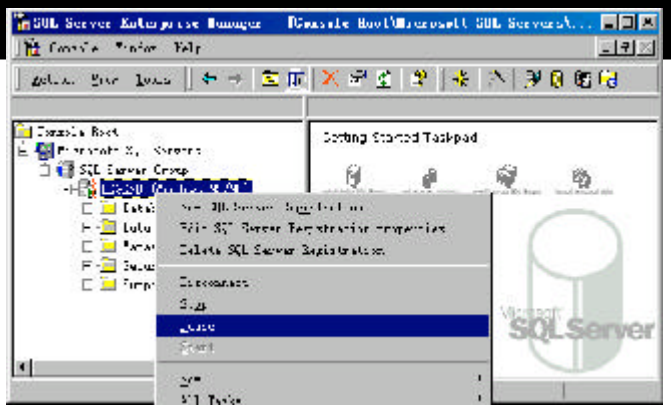
7 通过本例学会了如何制作启动选项来替换缺省启动选项。

例

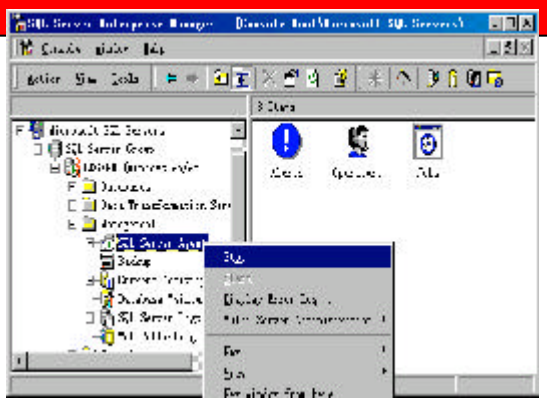
SQL Server 的启动、暂停、关闭
关闭 SQL Server

解

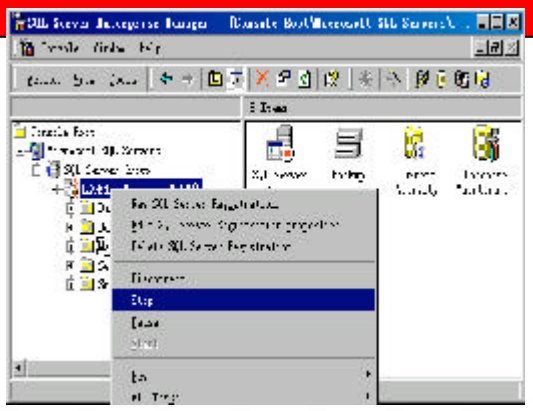
1



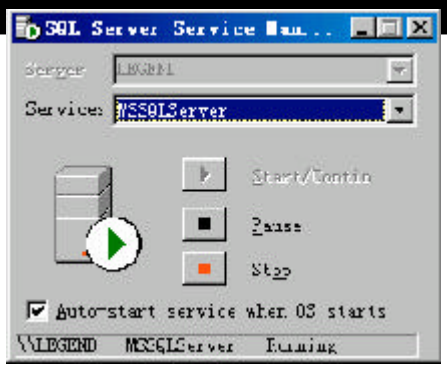
2 过一段时间以后，关闭 SQL ServerAgent。在 Management 中选择 SQL ServerAgent。右键单击弹出菜单，选择 STOP。



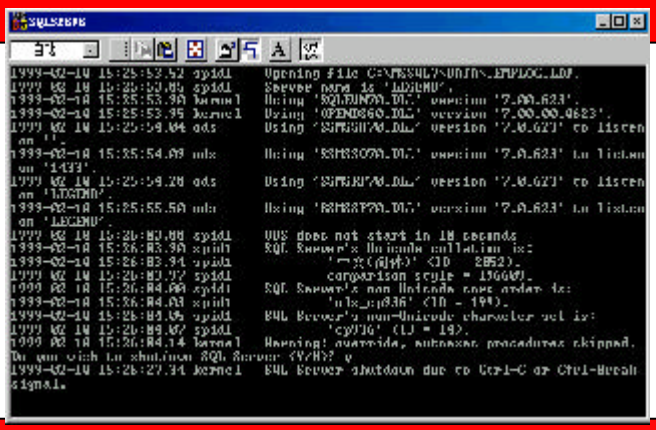
3 接着，右键单击服务器，在弹出菜单选择 Stop 如下图：



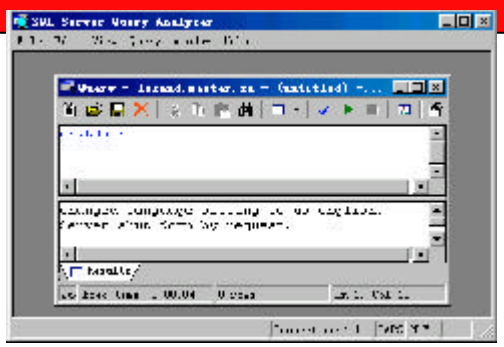
4



5 命令行方式：如果是用 sqlservr.exe 启动 SQL Server,则按下 CTRL+C。在询问是否关闭时选择 y（yes），就可以直接关掉 SQL Server 了。



6 也可以在 Query Analyzer 中，或者在 osql 中执行 Transact-sql 语句 SHUTDOWN 关闭服务器。



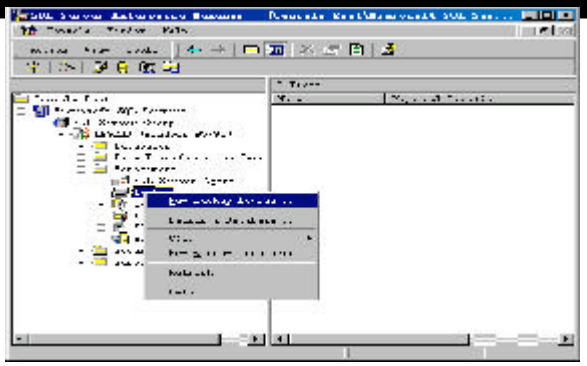
7 本例中学会了使用多种工具来关闭 SQL Server 服务器。

例

备份和恢复数据库
创建备份设备并备份数据库

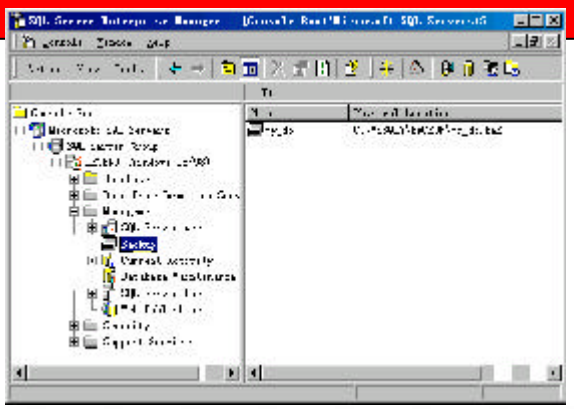
解

1

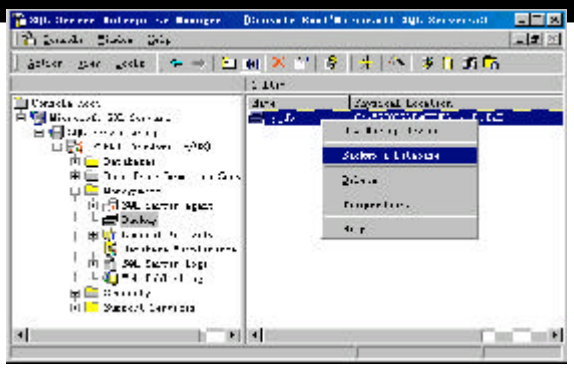


2 弹出对话框如下图：在 Name 中输入备份设备名称。再按确定。这时就创建了一个新的备份设备。

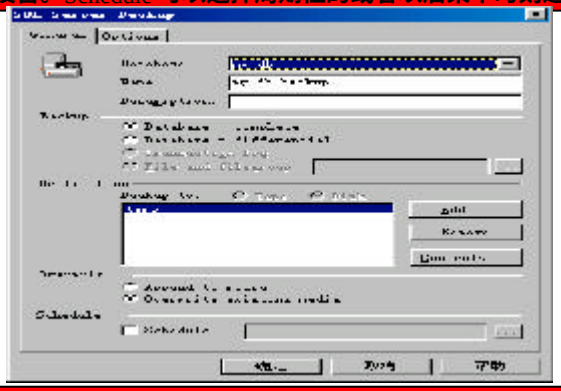
3 这时就能在 backup 窗口中看到新创建的备份设备如下图所示：



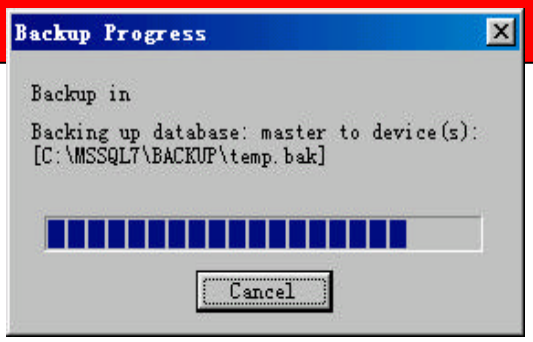
4



5 在弹出的对话框中，选择需要备份的数据库，输入该备份的名称和简单描述。可以在 Destination 中选择备份的设备。通过 Add 、Remove 添加或删除备份设备。Schedule 可以选择周期性的或者以后某个时刻进行备份。



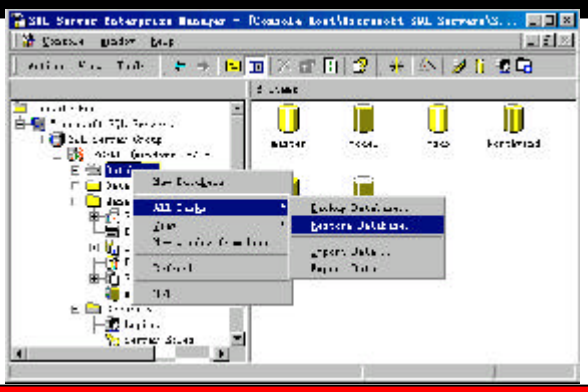
6 设定好了备份以后就按确定，接着就会进行备份操作。



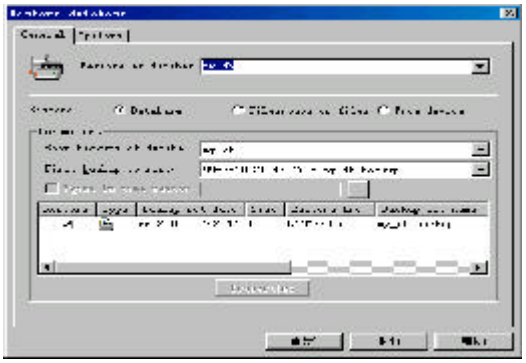
7 通过本例学会了如何创建备份设备和备份数据库的操作。

例 备份和恢复数据库 用备份文件恢复数据库 解

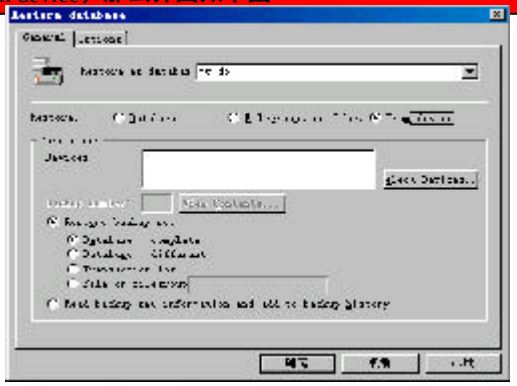
1



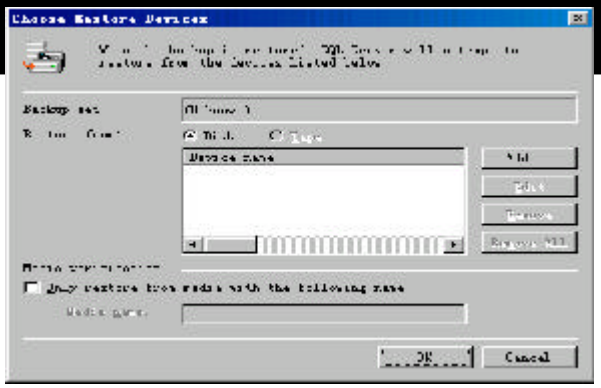
2 在弹出的对话框中，Restore as database 中输入需要恢复的数据库名称。在 Parameter 中可以选择备份的设备。Show Backups of database 用来选择从某个数据库的备份纪录中挑选备份设备。可供挑选的设备显示在最下方。



3 上图中，如果 Restore 选择的是 Database 那么将从某个数据库的备份纪录中找到备份设备进行恢复。这时只需再按确定就可以了。如果 Restore 选择 From device，那么界面如下图：

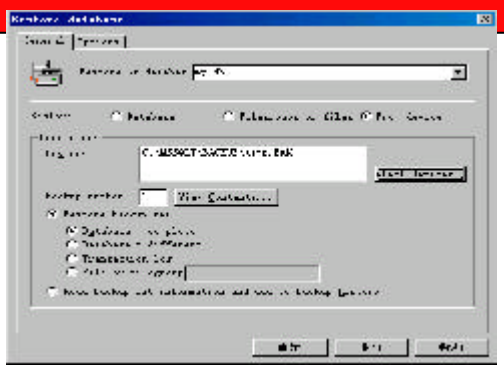


4



5 在下图对话框中选择某个文件或者设备。如果从文件恢复，在 File name 中输入文件名；如果从设备恢复，在 Backup device 中选择备份设备。

6 选好了设备后，Restore Database 对话框如下图所示，这时只需按确定就可以完成恢复操作了。



7 通过本例，学会了如何在 SQL Server 中对用户的数据库进行恢复。对于系统数据库的恢复操作，将在后面学习。

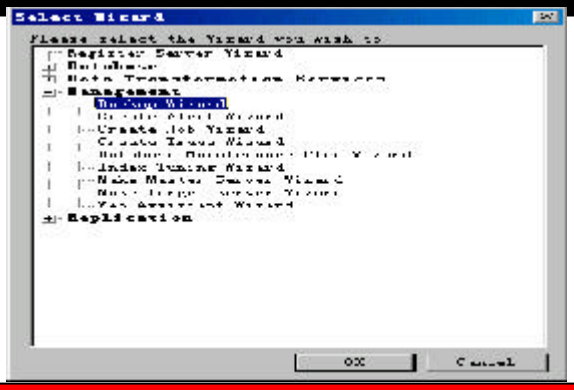
知识卷

例

备份和恢复数据库
使用备份向导备份数据库

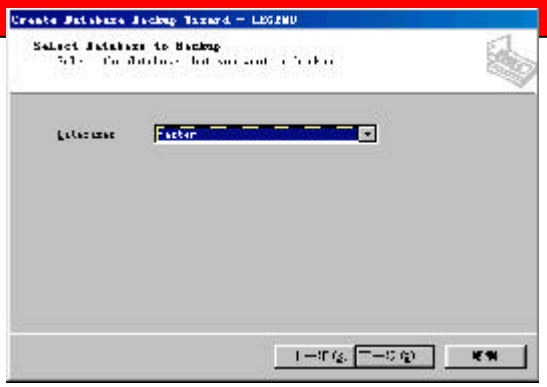
解

1

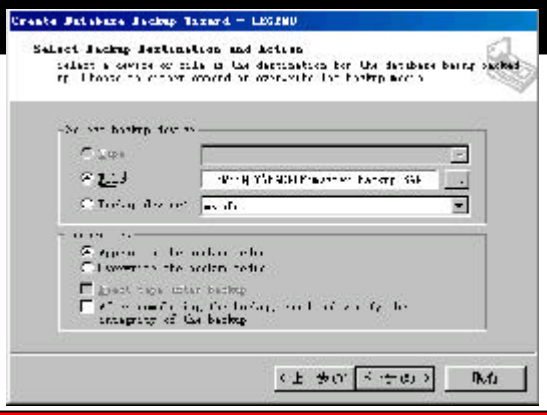


2 在 Management 中选择 Backup Wizard 单击 OK，进入 Backup Wizard 如下图：

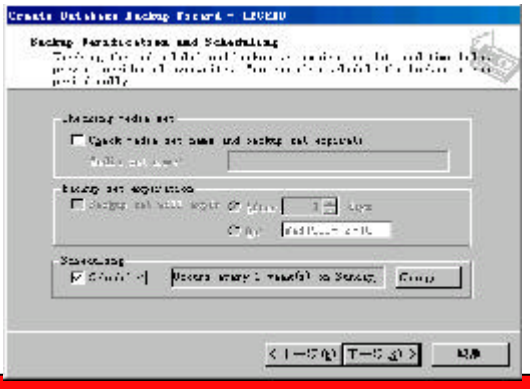
3 单击下一步。在 Database 中选择备份的数据库名称。



4



5 下图用来选择备份时是否需要检查媒介名称。Schedule 可以选择周期性的备份或者不是立刻备份而是在以后的某个时刻备份（可以通过 Change 设定）。单击下一步。



6 向导的最后会将选项汇总，报告给你。如下图，如果确信设置正确，单击完成。如果还要修改某项设置，可以按上一步，回到该处修改。

7 通过本例，学会了如何使用 Backup Wizard 向导。在 SQL Server 7.0 中，向导是非常简单、快捷而有功能完备的工具。

例

备份和恢复数据库
Transact-SQL 备份和恢复数据

解

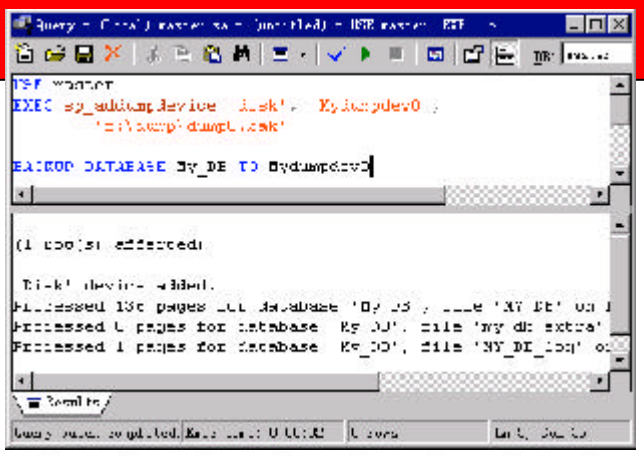
1

```
sp_addumpdevice [@devtype =] 'device_type',
   [@logicalname =] 'logical_name',
   [@physicalname =] 'physical_name'
   [, {
      [@cntrltype =] controller_type
         |
      [@devstatus =] 'device_status'
      }
   ]
```

2 在 Transact-SQL 中用 BACKUP 语句来备份数据库。BACKUP 的句法如下所示：

```
BACKUP DATABASE {database_name | @database_name_var}
TO <backup_device> [,...n]
[WITH
   [BLOCKSIZE = {blocksize | @blocksize_variable}]
   [[,] DESCRIPTION = {text | @text_variable}]
   [[,] DIFFERENTIAL]
   [[,] EXPIREDATE = {date | @date_var}
      | RETAINDAYS = {days | @days_var}]
   [[,] FORMAT | NOFORMAT]
   [[,] {INIT | NOINIT}]
   [[,] MEDIADESCRIPTION = {text | @text_variable}]
   [[,] MEDIANAME = {media_name | @media_name_variable}]
   [[,] NAME = {backup_set_name | @backup_set_name_var}]
   [[,] {NOSKIP | SKIP}]
   [[,] {NOUNLOAD | UNLOAD}]
   [[,] {RESTART}]
   [[,] STATS [= percentage]]
]
```

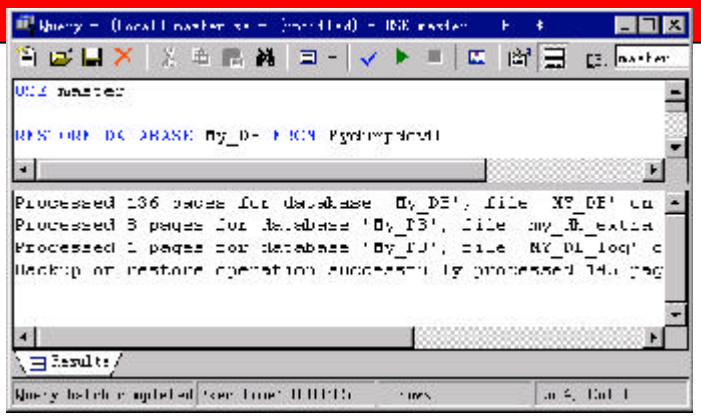
3 在 Query Analyzer 中 执行 sp_addumpdevice 和 BACKUP 对数据库进行备份。



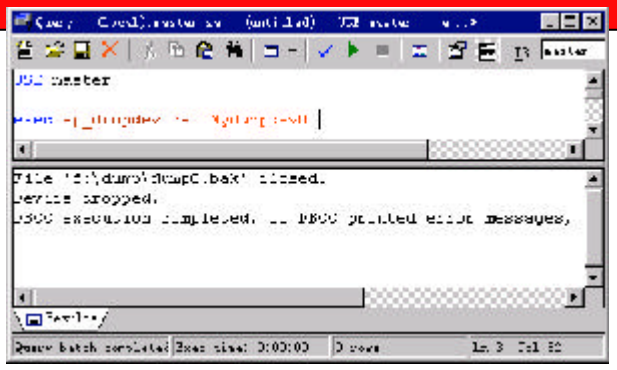
4

```
RESTORE DATABASE {database_name | @database_name_var}
[FROM <backup_device> [,...n]]
[WITH
   [DBO_ONLY]
   [[,] FILE = file_number]
   [[,] MEDIANAME = {media_name | @media_name_variable}]
   [[,] MOVE 'logical_file_name' TO 'operating_system_file_name']
        [,...n]
   [[,] {NORECOVERY | RECOVERY | STANDBY = undo_file_name}]
   [[,] {NOUNLOAD | UNLOAD}]
   [[,] REPLACE]
   [[,] RESTART]
   [[,] STATS [= percentage]]
]
```

5 **典型的执行恢复数据库的操作下图所示：**RESTORE DATABASE my_db FROM Mydumpdev0 。



6 sp_dropdevice **的句法为**: sp_dropdevice [@logicalname =] 'device' [, [@delfile =] 'delfile']，**在** Query analyzer **中执行** sp_dropdevice **的效果如下图**:



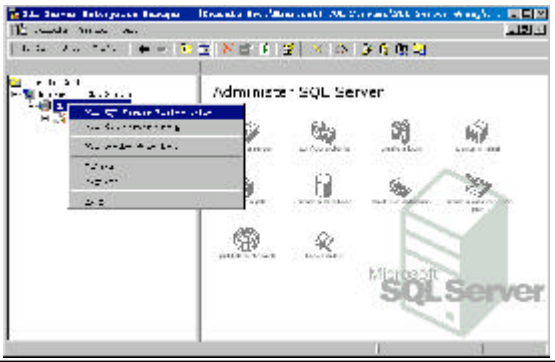
7 **通过本例学习了如何使用** Transact-SQL **语句完成数据库的备份和恢复操作。事实上，**Transact-SQL **语句能够完成** SQL Server **中的任何数据库操作。读者要尽量的熟悉** Transact-SQL **语句的用法。**

例

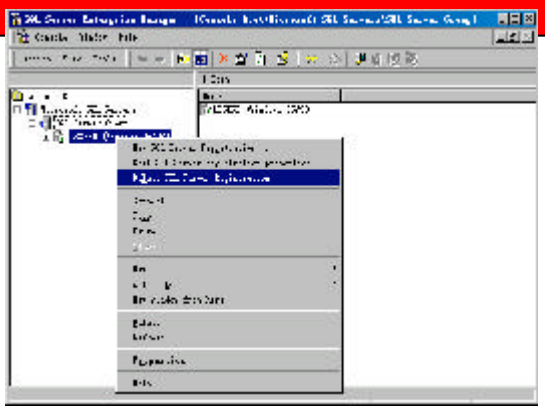
管理服务器端
操作注册后的服务器

解

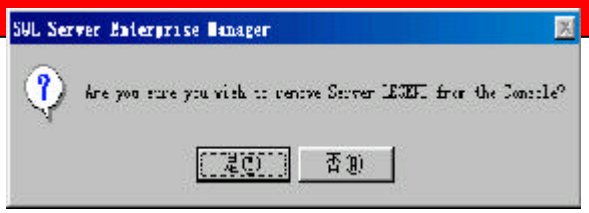
1



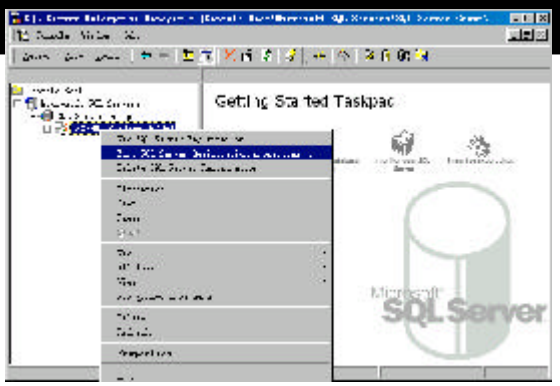
2 删除某个已注册的服务器：选择某个已注册的服务器，右键单击，在弹出菜单中选择 Delete SQL Server Registration。



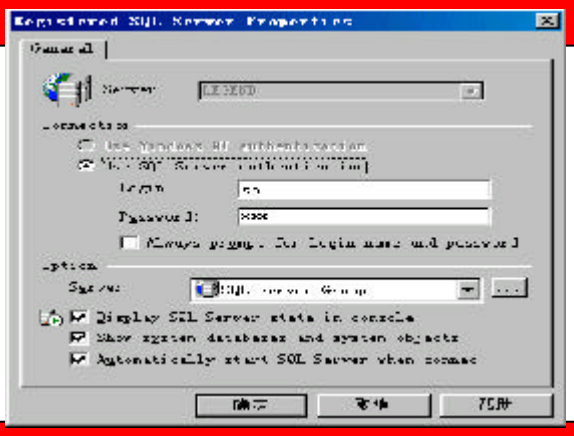
3 在弹出的确认对话框中选择“是（Y）”确认。



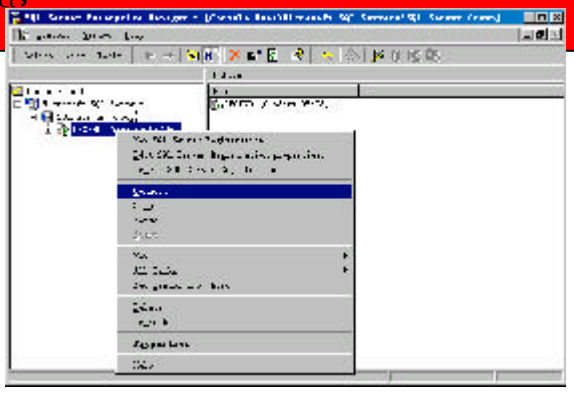
4



5 在弹出的对话框中设置注册对话框的属性。这些属性与服务器注册过程中输入的属性意义完全相同。



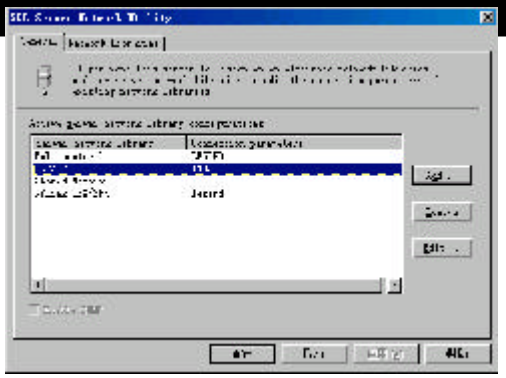
6 连接已注册服务器：右键单击某个已注册的服务器，在弹出菜单中选择 Connect，



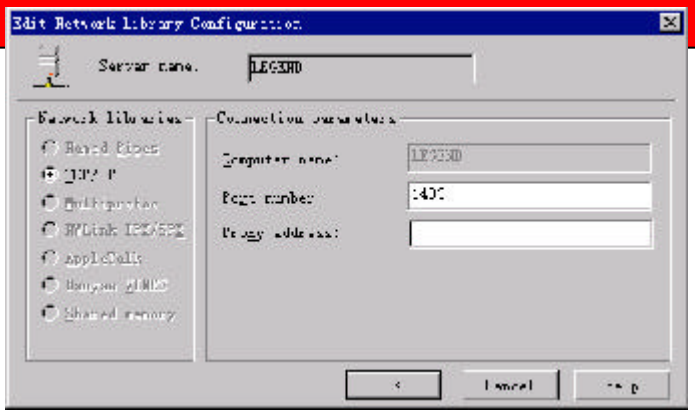
7 通过本例，学会了如何使用 Enterprise Manager 注册 SQL Server 服务器，如何修改注册服务器的信息，如何连接服务器。

例 管理服务器端 设置代理服务器和侦听端口号 解

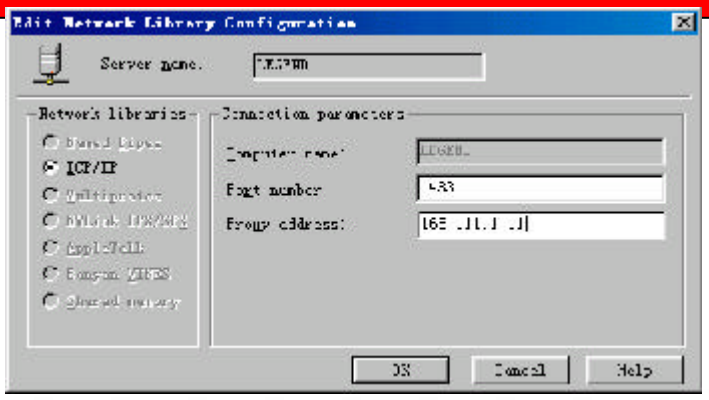
1



2 弹出 TCP/IP 属性对话框如下图。

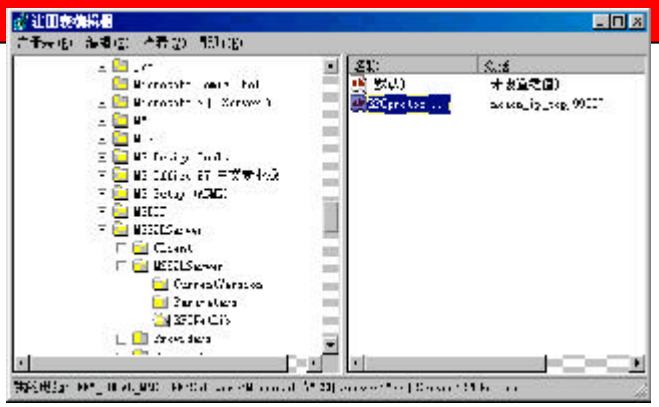


3 在该对话框中的 Proxy Address 输入框中输入你的代理服务器的 IP 地址，或者 DNS 地址。如下图，再按 OK 确定。

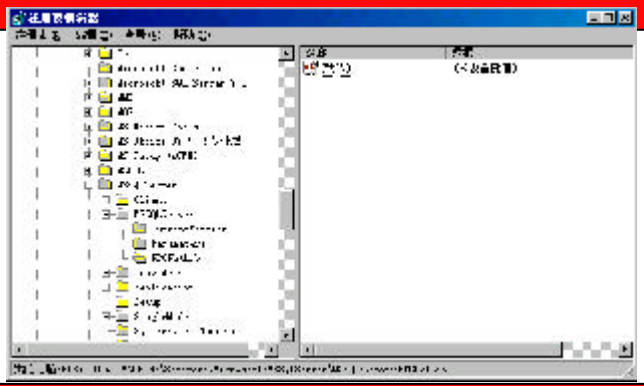


4

5 在注册库中打开以下主键 HKEY_LOCAL_MACHINE－〉SOFTWARE－〉MICROSOFT－〉MSSQLSERVER－〉MSSQLSERVER－〉RPCNetlib



6 在该主键中添加新键 RPCprotocols， 赋值为字符串 ncacn_ip_tcp[,port]。其中如果输入 Port。它代表侦听的端口号。



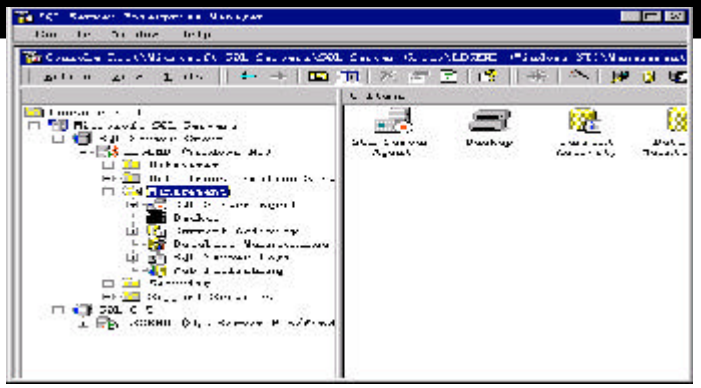
7 通过本例，学会了如何设置 SQL Server 的代理服务器，以及如何设置 SQL Server 侦听专门的端口。

例

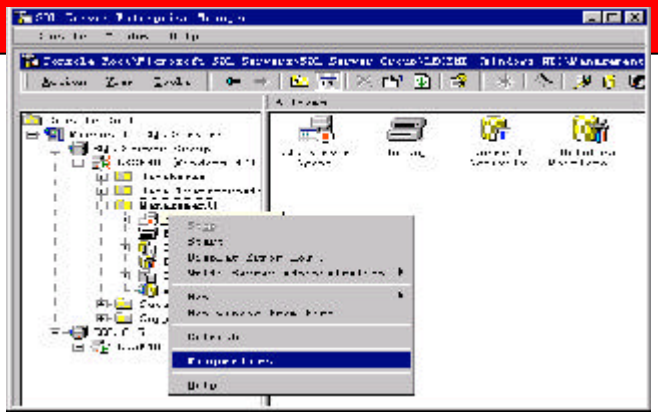
设定自动执行任务
设置 SQL ServerAgent

解

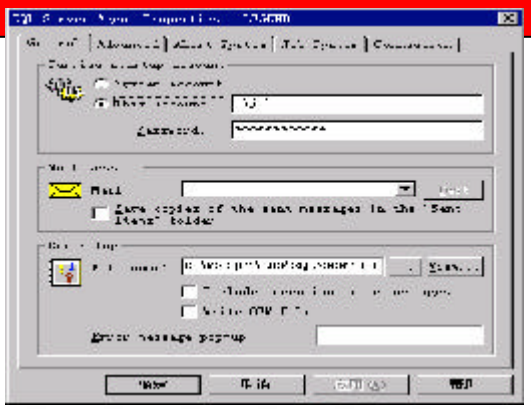
1



2 右键单击 SQL Server Agent，在弹出菜单中选择 Properties。

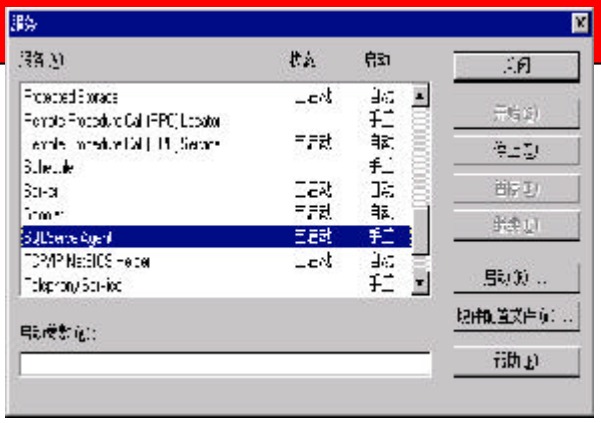


3 选择 Service Startup Account。在 System Account 和 This account 两种选项中选择一种。如为 This account，则在右边输入 Windows NT 帐号及口令。

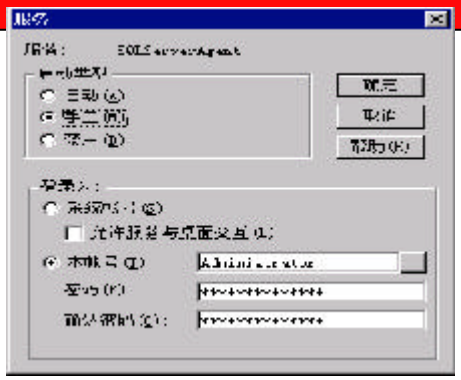


4

5 在控制面板中双击打开服务项，如下图：



6 在服务项目中双击鼠标选择 SQL Server Agend。在“本帐号”中输入 SQL Server Agent 使用的帐号名及相应的密码。



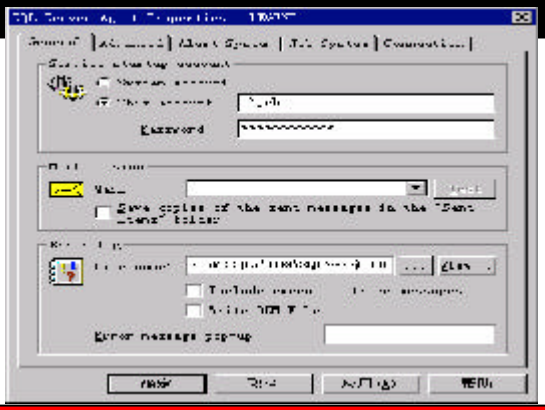
7 通过本例，我们学会了如何设置 SQL Server Agent 使用的用户帐号。

例

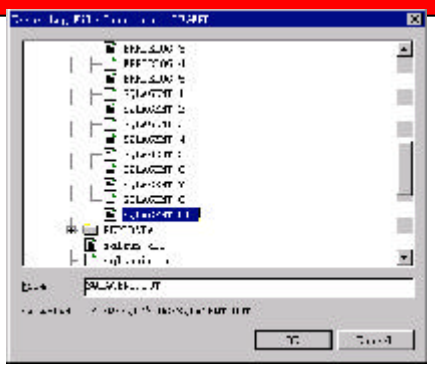
设定自动执行任务
设置 SQL ServerAgent

解

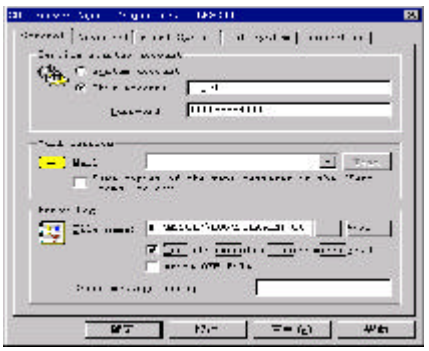
1



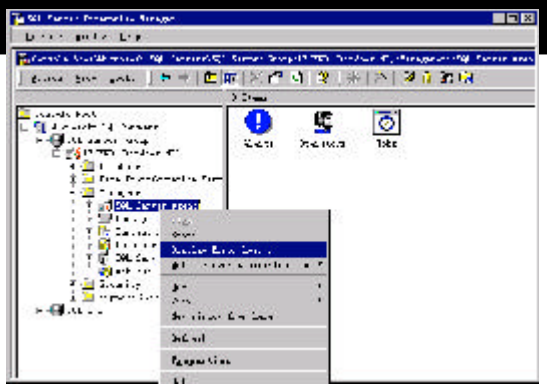
2 在 File name 中可以直接输入文件名，也可以单击按钮 ...，弹出目录树，如下图，在其中可以选择路径名和文件名。



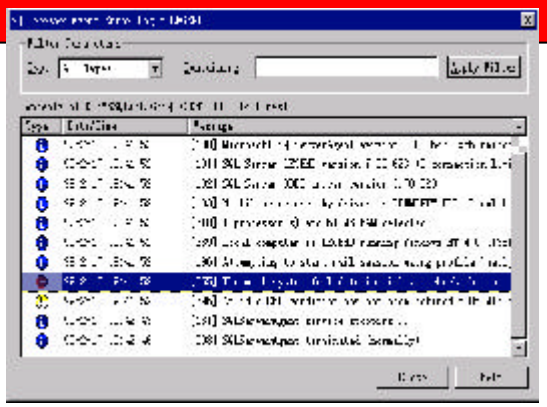
3 在 SQL Server Agent 属性对话框中选择 Include execution trace messages，可以在错误记录文件中添加一般信息。注意：这将增大记录文件，若非调试错误尽量不要选择这个选项。



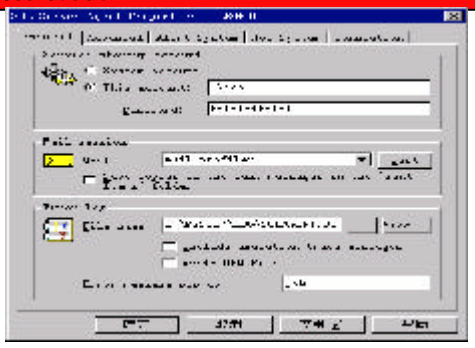
4



5 下图中显示的是错误记录文件的内容，其中 表示一般信息， 表示错误信息， 表示警告信息。



6 在 SQL Server Agent 属性对话框的 Error message pop-up 框中可以选择当 SQL Server Agent 执行过程中发生错误时，将通过对话框通知该文本框中指定的用户或者计算机。



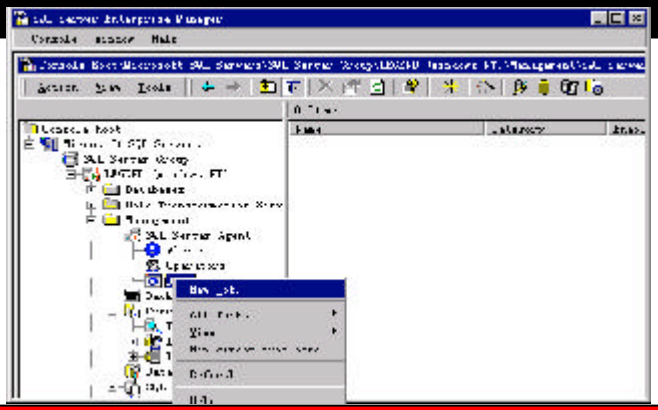
7 通过本例，学习了如何根据 SQL Server Agent 的错误记录文件来跟踪监视 SQL Server Agent 的执行情况。

例

设定自动执行任务
定义任务

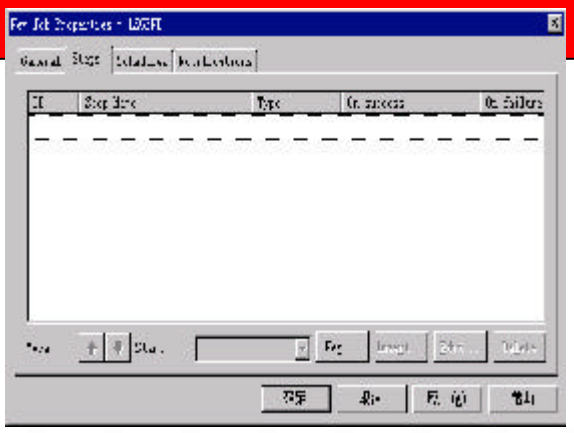
解

1

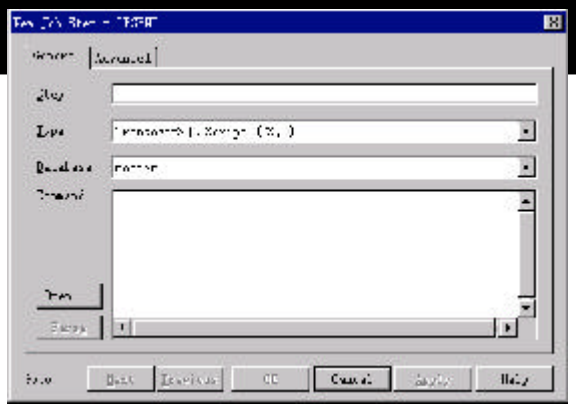


2 在弹出对话框中输入任务的属性：如名称、描述等。如下图。Enabled 项表示任务定义完后可以立刻执行。Target 可以是本地服务器，也可以是网络多台服务器。

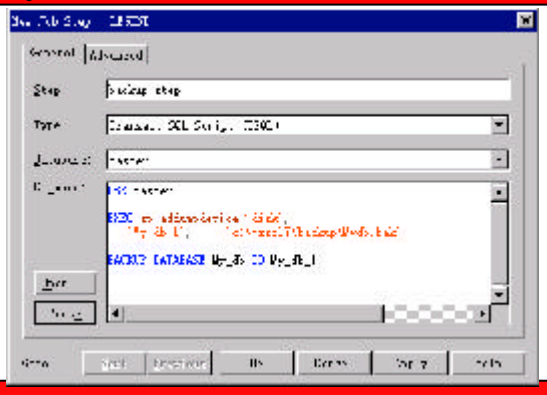
3 单击 Steps 页面，用于设定任务的各个工作步骤，即指明任务的具体内容。



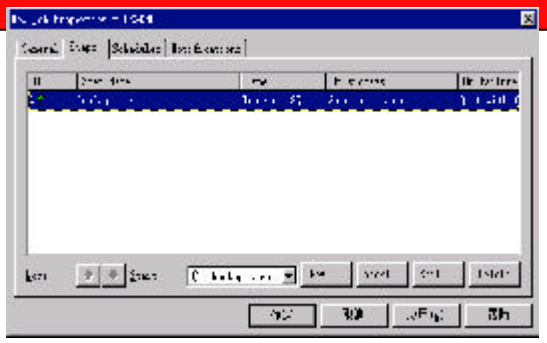
4



5 在上图对话框中输入步骤名称、选择描述步骤的方法等属性。本例使用 Transact-SQL 语言描述。在 Database 中选择要操作的数据库，在 Command 中输入 SQL 语句，Parse 按钮可以执行对 Command 的语法检查。



6 再按下确定键，就可以完成任务工作的设定。如下图所示，表示已经添加了一个工作步骤。再按下确定键，就完成了任务的定义。

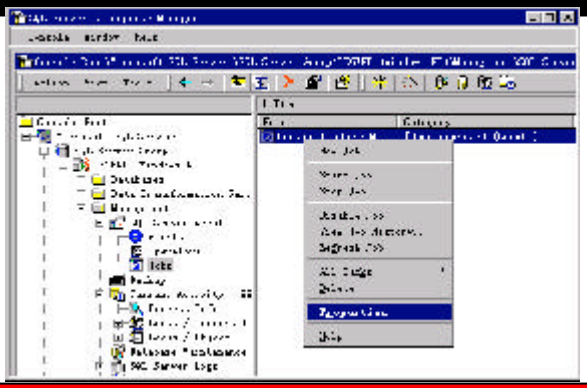


7 通过本例，学会如何在 Enterprise Manager 中定义一个任务。后面将会学习如何设置任务的其他高级属性，如调度等。

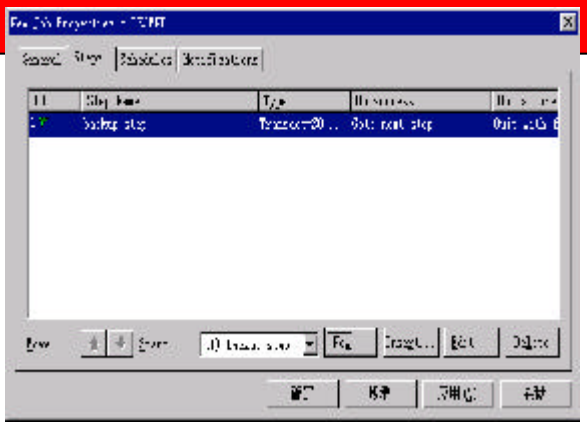
例 设定自动执行任务 解

多种方法描述任务步骤

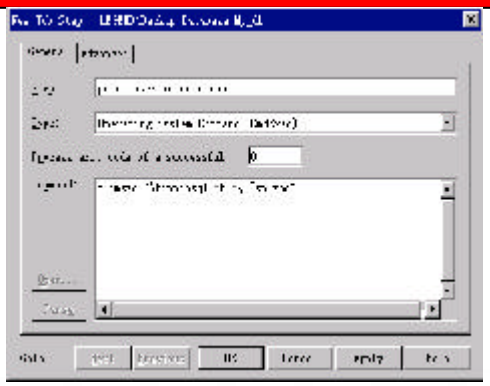
1



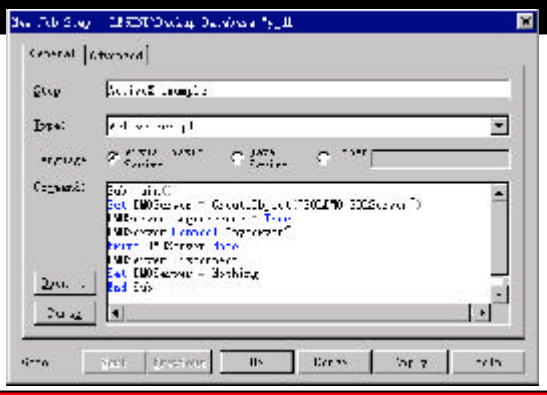
2 在弹出对话框中选择 Steps 页面，单击 New 在上一工作步骤之后添加新步骤。



3 在弹出对话框中，输入步骤名称，选择类型为 Operating System Command (CmdExec)，在 Command 中输入可执行文件的完整路径及相应参数。单击 OK 就完成了添加工作步骤的操作。



4



5 下图中所示的是使用 VBScript 脚本语言描述工作的例子。将其输入 Command 栏，即可执行。

```
Rem VBScript job step example:
Sub main()
Set DMOServer = CreateObject("SQLDMO.SQLServer")
DMOServer.LoginSecure = True
DMOServer.Connect "myserver"
Print DMOServer.Name
DMOServer.Disconnect
Set DMOServer = Nothing
End Sub
```

6 下图所示的为使用 PerlScript 脚本语言描述工作步的例子。将其加入 Command 栏，同时 Language 中选择 other 并输入 PerlScript ，即可执行。

```
#PerlScript job step example:
Sub main()
{
$DMOServer = $SQLActiveScriptHost->CreateObject
("SQLDMO.SQLServer");
$DMOServer->Connect("myserver", "sa");
$SQLActiveScriptHost->Print($DMOServer->Name);
$DMOServer->Disconnect();
$DMOServer = undef;
}
```

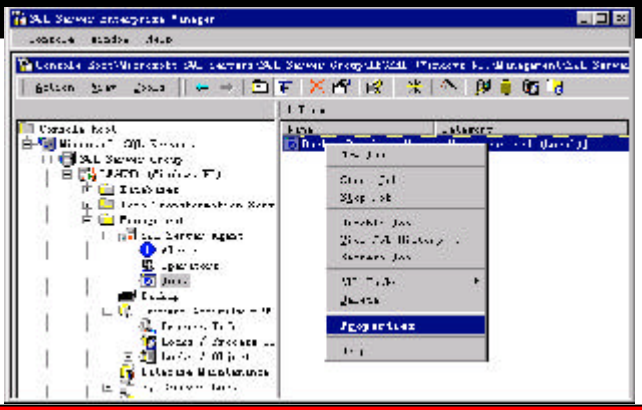
7 通过本例学习了如何描述工作步骤的命令方式和 ActiveX 脚本方式。

例

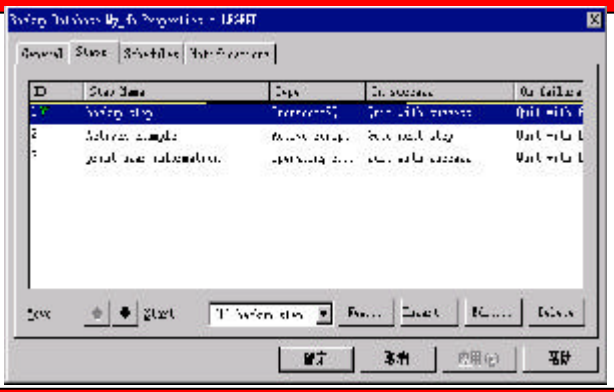
设定自动执行任务
设置工作步执行顺序和任务记录

解

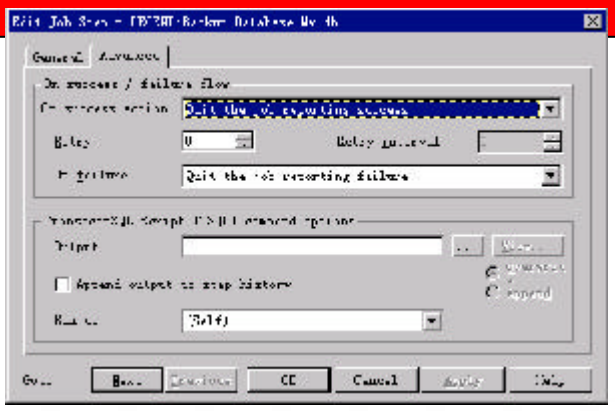
1



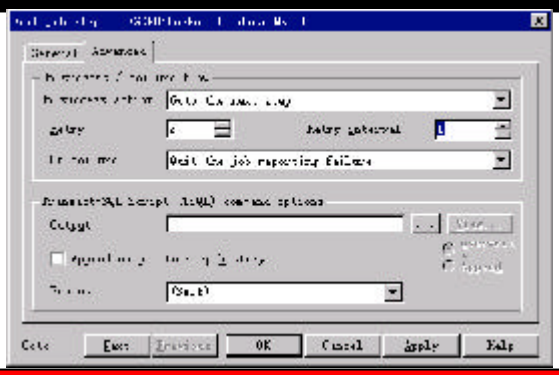
2 在弹出的属性对话框中选择 Steps 页面，如下图。其中，Start 右边可以选择任务的起始工作步。选定某个工作步后， 通过 ↑ 、↓ 两个按钮可以上下移动某个工作步的相对序号。



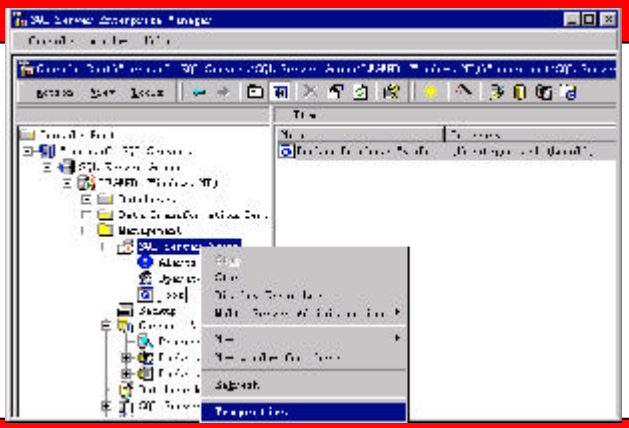
3 选中某个工作步后，双击该工作步，或者单击 Edit 键编辑该工作步的属性。在弹出的属性对话框中选择 Advanced 页面。



4



5 设置任务执行记录：在 Enterprise Manager 中右键单击 SQL Server Agent，在弹出菜单中选择 Properties。



6 在 SQL Server Agent 的属性对话框中，选择 Job System 页面，在 Job History Log 中输入记录文件的最大行数和记录每个执行任务的最大行数。

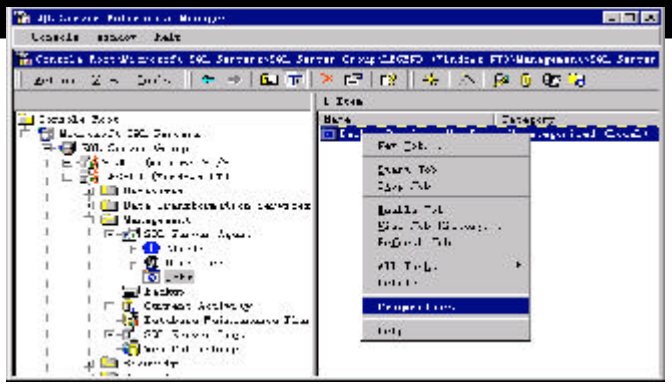
7 通过本例，学习了如何控制任务中各个工作步的执行顺序，和设置任务执行记录文件的大小。

例

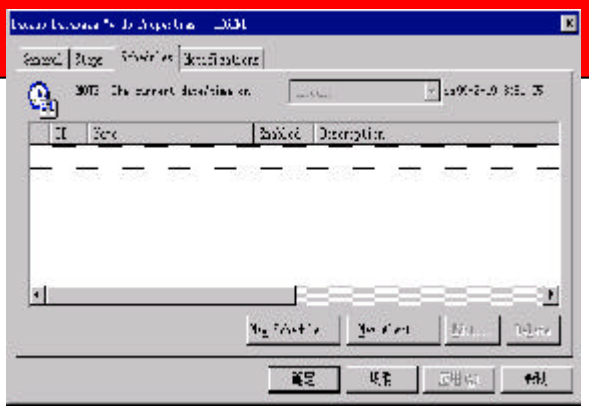
设定自动执行任务
调度任务的执行时机

解

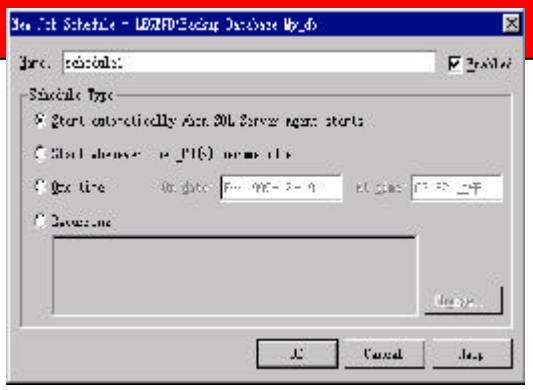
1



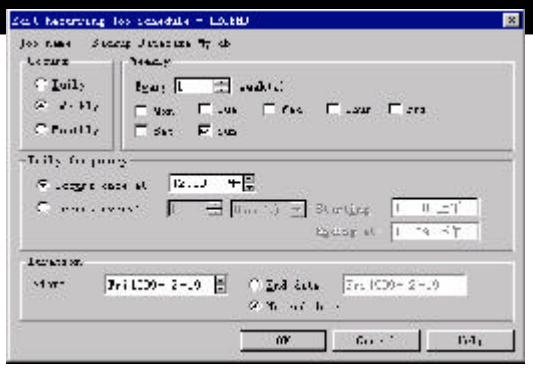
2 在弹出的对话框中选择 Schedules 页面，如下图所示：



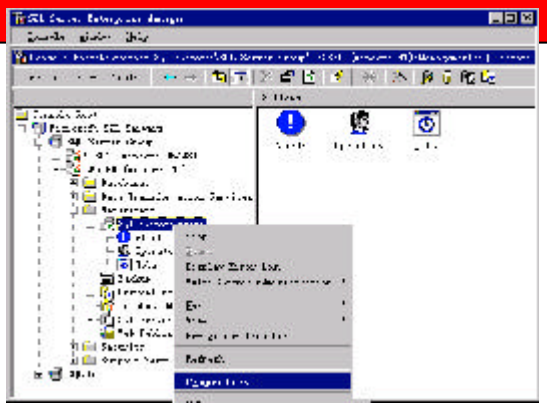
3 单击 New Schedule 按钮，弹出对话框如下。在 Name 中输入调度的名称。



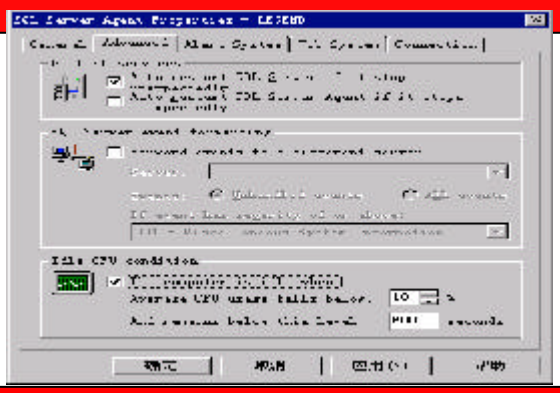
4



5 设置 CPU 空闲指标：在 Enterprise Manager 中右键单击 SQL Server Agent，在弹出的菜单中选择 Properties 项。



6 在弹出对话框中选择 Advanced 页面，在 Idle CPU Condition 中选择该复选框，然后就可以设定 CPU 空闲的指标了。



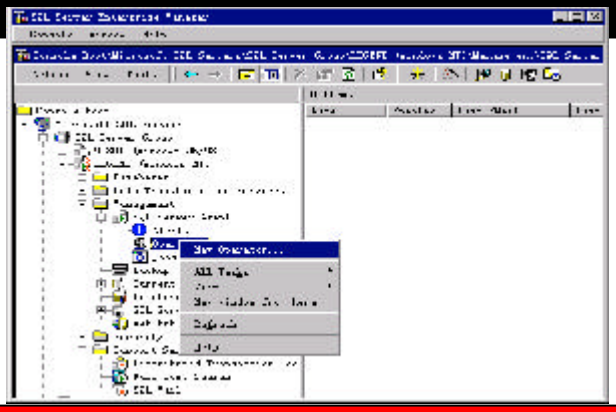
7 通过本例，学会了如何设置调度管理任务的自动执行。以及如何设置 CPU 的空闲标准。

例

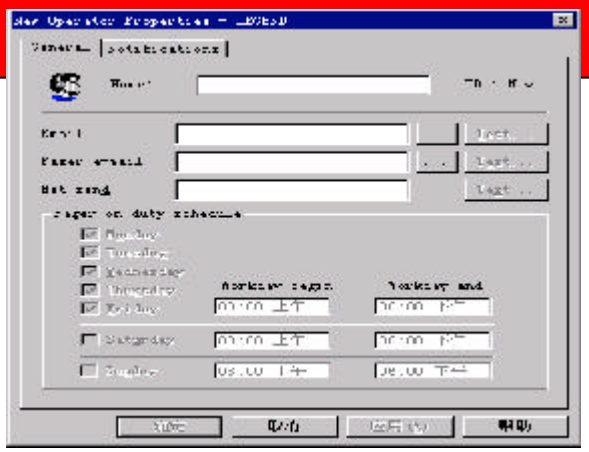
设定自动执行任务
设置操作员

解

1



2 弹出对话框如下图。



3 在对话框中输入名称、联系方式等信息如下图：其中的电子邮件和呼叫器地址可以设为操作员的电子邮件地址。Net Send 中可以输入操作员的帐号名或者所在服务器的名称。

4

5 单击 Net Send 右边的 Test 栏可以试着将消息发送至操作员所在服务器上。发送成功，操作员会接受到如下对话框。

6 在操作员属性的 notification 页面可以设置，当发生那些报警时，分别采用何种方式通知操作员。如下图：

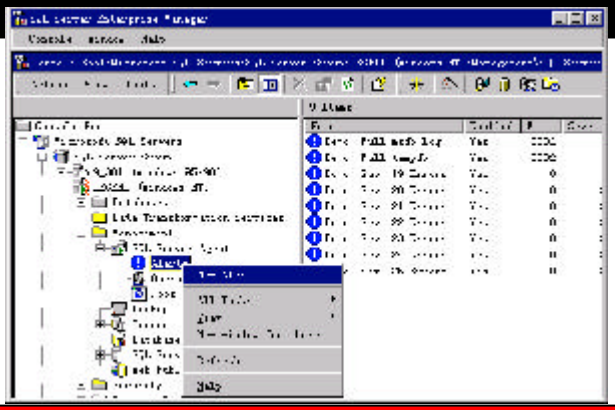
7 通过本例，学会了如何通过 Enterprise Manager 创建一个操作员。

例

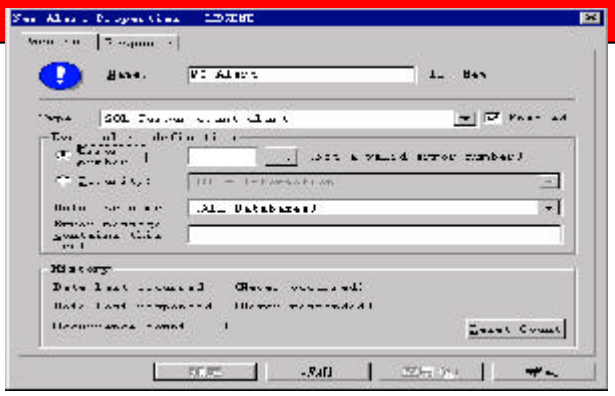
设定自动执行任务
设置警告信息

解

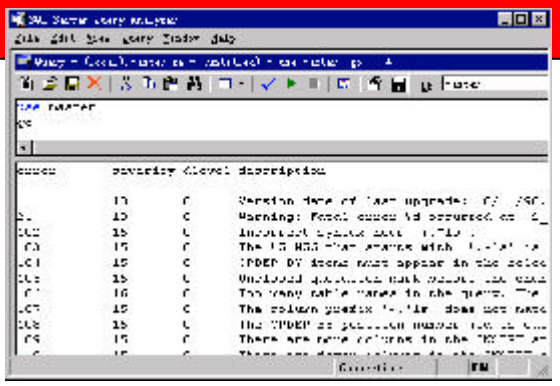
1



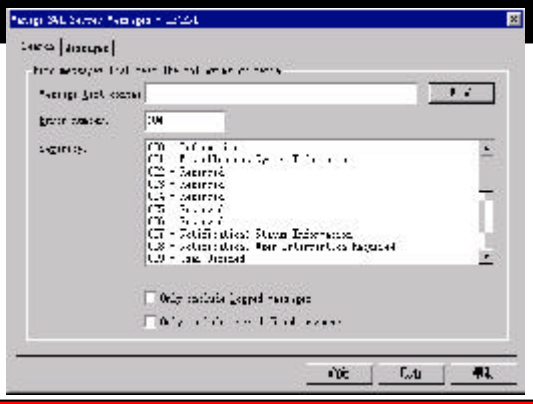
2 在弹出对话框中，设置警报的属性。如名称、类型、使能等。在 Event Alert definition 中选择 Error number 即可以选择错误序号。



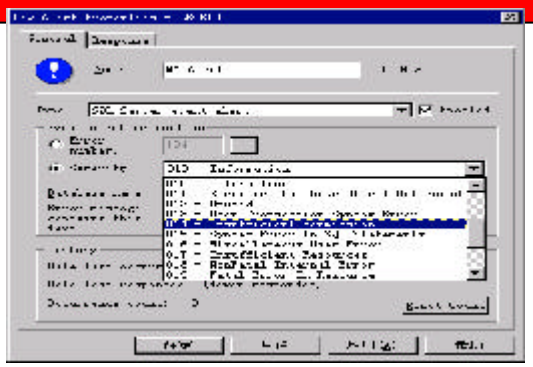
3 使用 select * from sysmessages 语句可以查出所有的错误序号如下图：



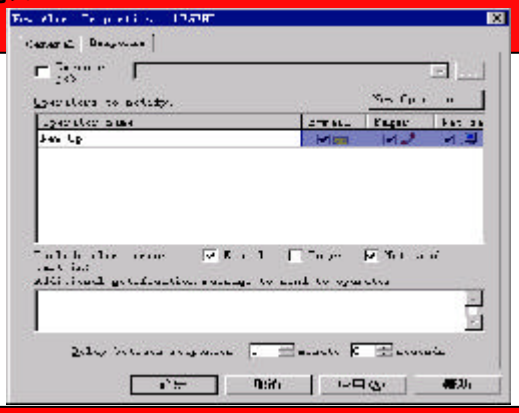
4



5 如果选择 Severity，可以对指定的严重性级别的事件报警。如下图列出了所有可供选择的严重性级别。



6 单击 Alert 属性的 Response 页面可以设置发生该警报时需要执行的操作。如下图所示：



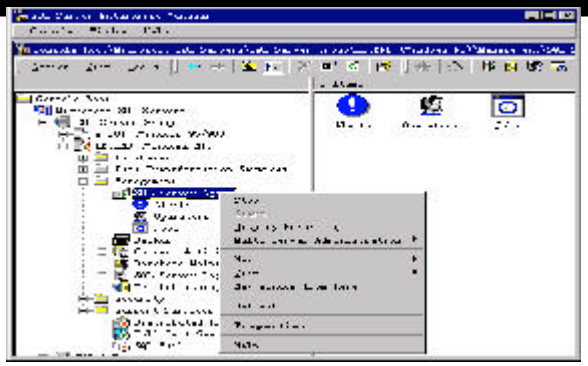
7 通过本例，学会了如何创建自己的警报，并且设置对该警报的事件处理。

例

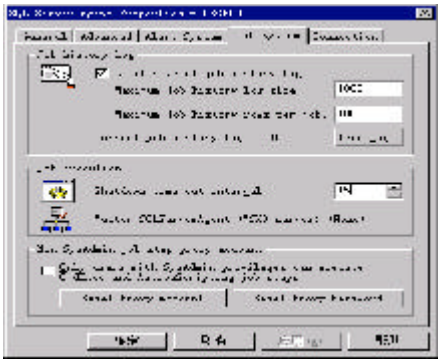
设定自动执行任务
SQL Server Agent 监视器

解

1

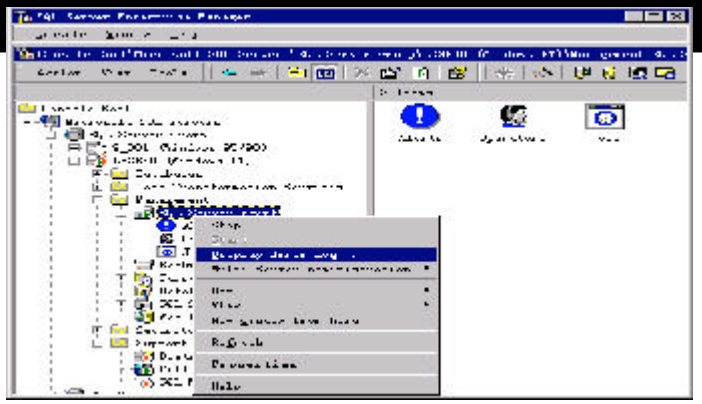


2 在弹出的属性对话框中的 Job System 里面的 Job Execution 栏中设定 Shutdown time-out interval 用来设定 SQL Server Agent 在关闭时最多可以等待当前执行任务结束的时间。超过该时间，SQL Server Agent 就会关闭。

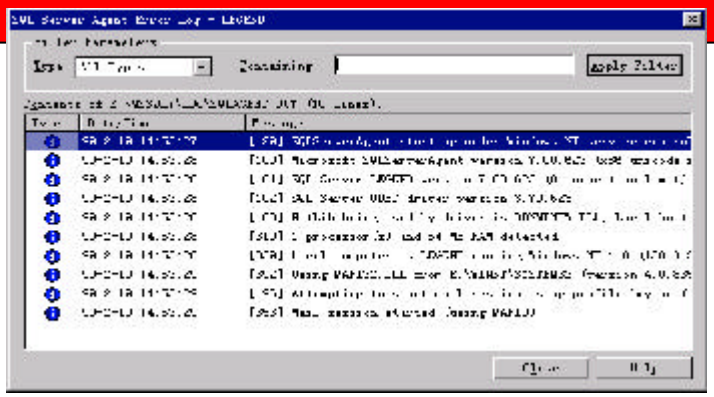


3 设置 SQL Server Agent 监视器的自启动：在上图的属性对话框中选择 Advanced 页面，在 Restart Services 中选择 Auto restart SQL Server if it stops unexpectedly 和 Auto restart SQL Server Agent if it stops unexpectedly

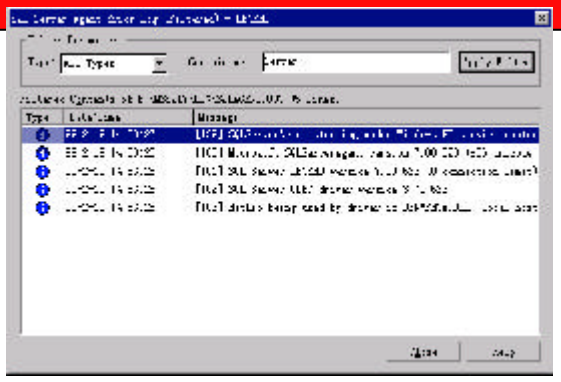
4



5 弹出的对话框中显示了错误记录文件的内容。注意：错误记录文件并不一定记录的都是错误，它可以记录消息、事件、警告等。



6 在 Containing 栏中输入特定文本，再按 Apply Filter 可以对现实的消息进行过滤，只有那些包含指定文本的消息才被显示。如下图：



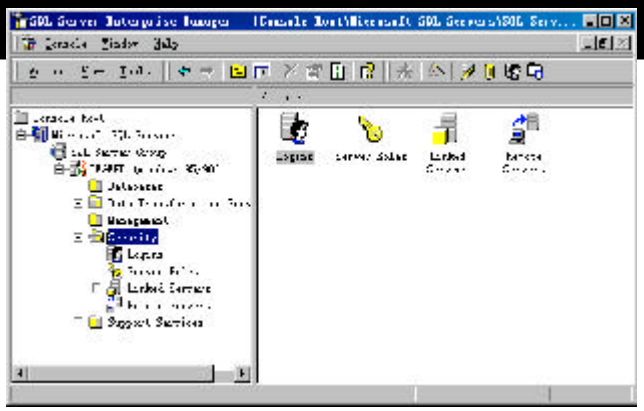
7 通过本例学会了如何在 Enterprise Manager 中设置任务关闭等待时间、设置 SQL Server Agent 监视器的自启动和查看 SQL Server Agent 的错误记录文件。

例

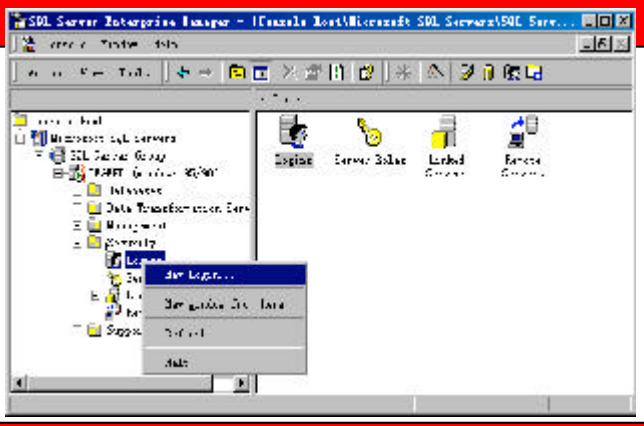
管理用户权限
添加用户

解

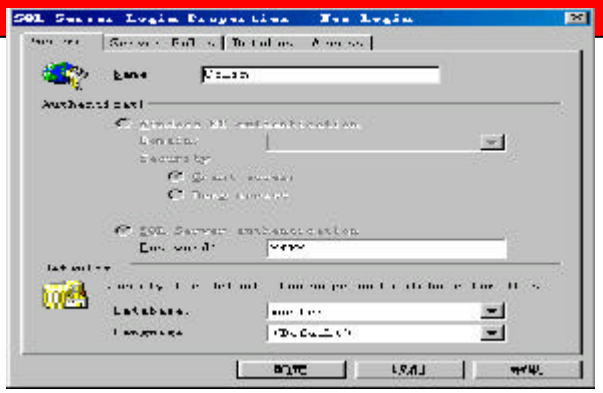
1



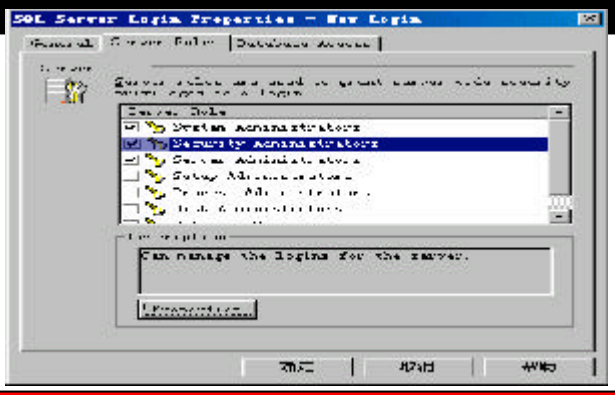
2 右键单击 Logins，在弹出菜单中选择 New Logins。



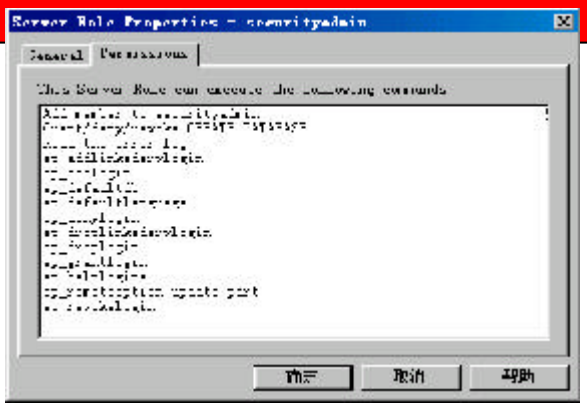
3 在弹出对话框中的 General 页面中，输入登录名称、口令，选择缺省数据库，选择使用的语言。



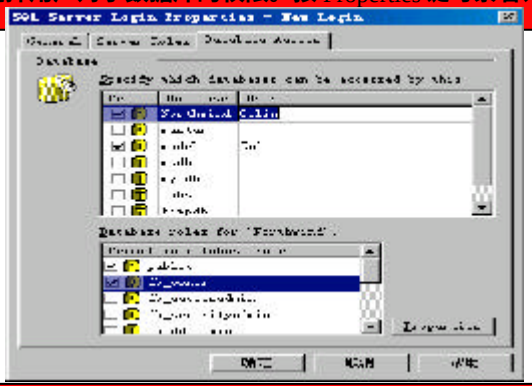
4



5 按下 Properties 键后在弹出的对话框中选择 Permissions 可以详细察看该角色可以执行的系统存储过程。如下图：



6 在 Database Access 页面中设置该登录对于各个数据库的访问权限。如下图，先选择数据库名，在 User 中可输入该 Login 对于数据库的用户名。在下栏中选择用户对于数据库的权限。按 Properties 键可察看详细信息。



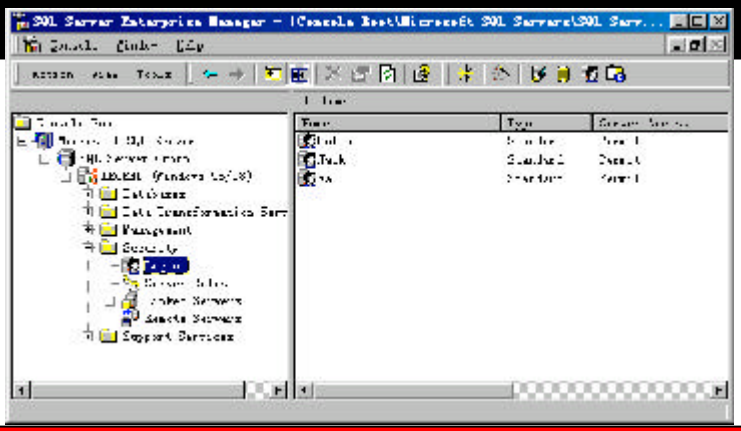
7 通过本例，学会了如何在 SQL Server 中添加新的登录名。以及如何设置新的登录名对于系统和各个不同数据库的访问权限。

例

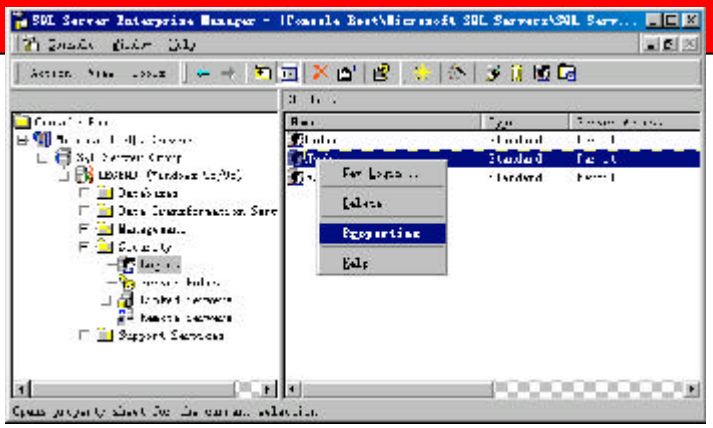
管理用户权限
察看、修改用户信息

解

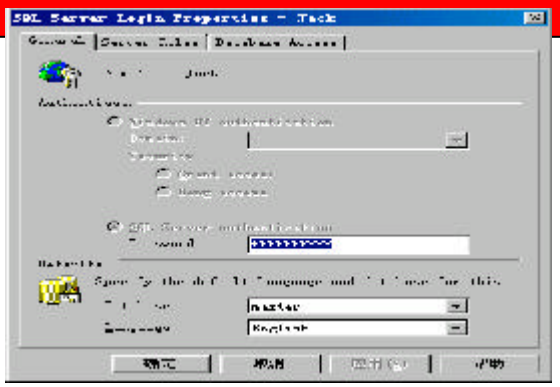
1



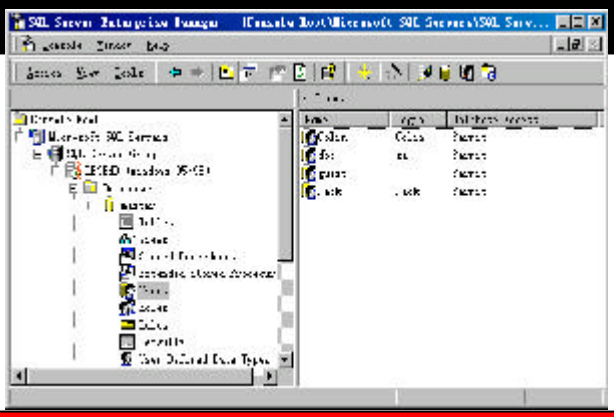
2 右键单击要察看的登录名，在弹出菜单中选择 Properties。



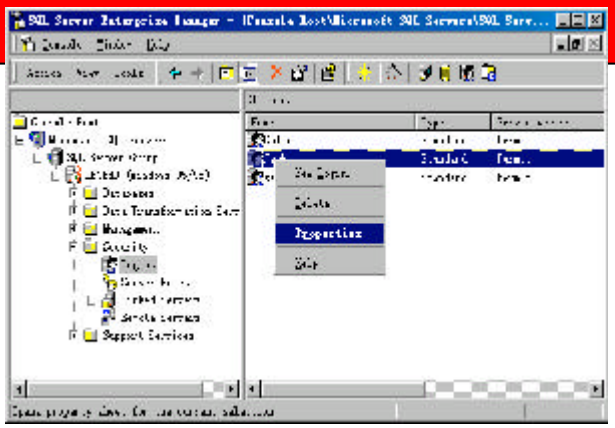
3 在弹出的属性对话框中可以察看登录的属性，并且可以修改其属性，如：用户密码、缺省数据库、缺省使用语言等。



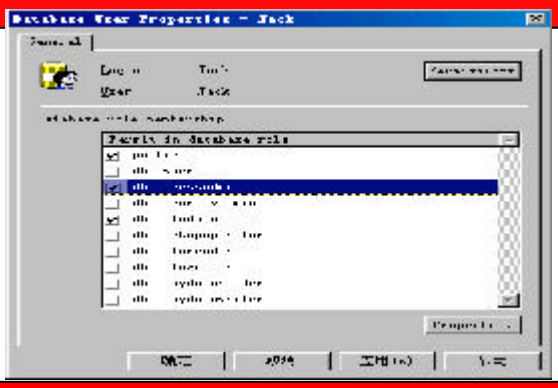
4



5 右键单击某个用户，在弹出菜单中选择 Properties。



6 弹出的对话框中，是有关选定用户相对于该数据库的操作权限。如下图：可以在其中修改用户权限。单击 Permissions 键可以展开更详细的信息。



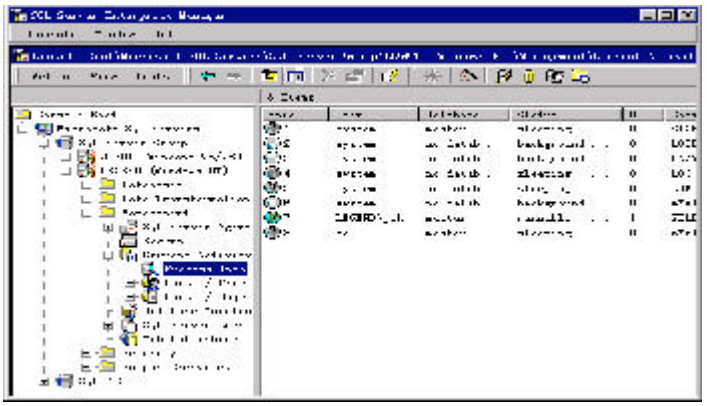
7 通过本例，学习了如何察看和修改用户对于系统和数据库的访问权限等信息。

例 监视 SQL Server 的行为 使用 Enterprise Manager 监视 解

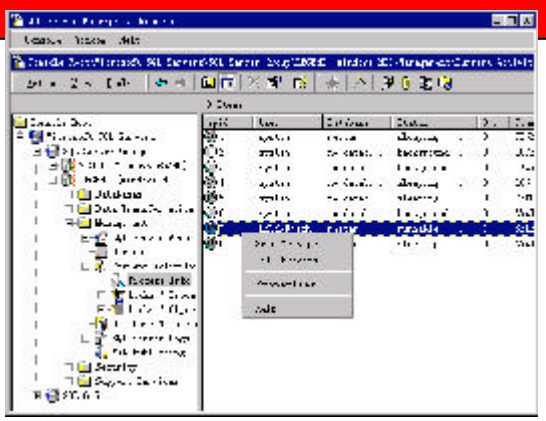
1

Event or activity	SQL Server Profiler	Windows NT Performance Monitor	Current activity window	Transact-SQL	Error logs
Trend analysis	Yes	Yes			
Replaying captured events	Yes				
Ad hoc monitoring	Yes		Yes	Yes	Yes
Generating alerts		Yes			
Graphical interface	Yes	Yes	Yes		Yes
Using within custom application	Yes (1)			Yes	

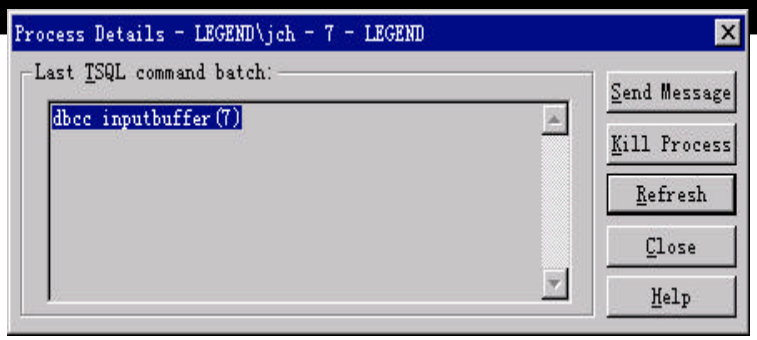
2 查看服务器当前状态：在 Enterprise manager 中展开服务器，展开 Management，打开 Current activity 项，单击 Process info 显示当前进程。



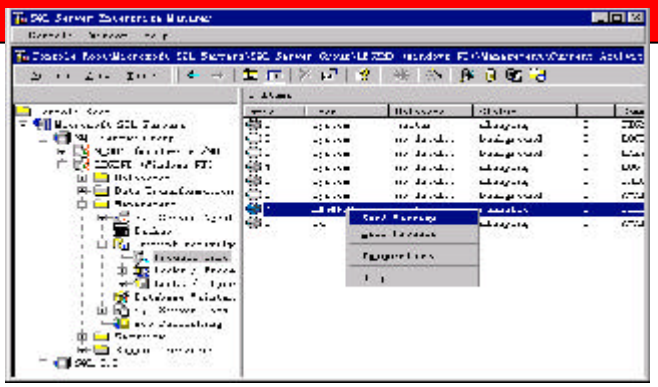
3 查看用户最近执行的命令：在上图右栏中右键单击某个 spid，在弹出对话框中选择 properties。



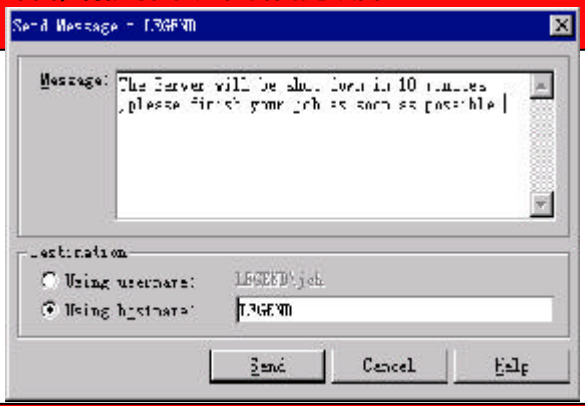
4



5 向进程用户发送消息：在 Current Activity 窗口中，右键单击某个进程，在弹出菜单中选择 Send Message。



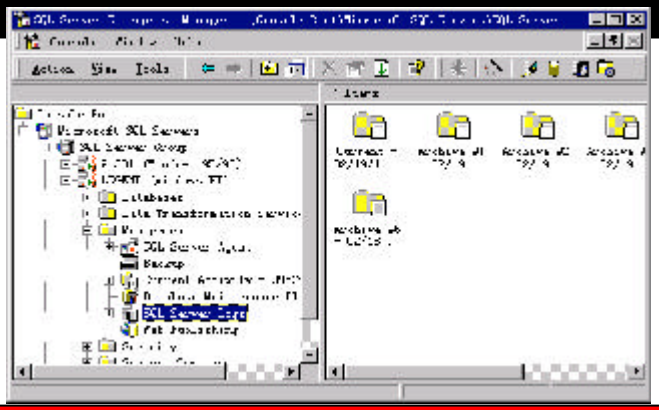
6 在弹出的对话框中输入消息的内容，选择向用户所在的计算机上发送消息。如下图，再按 Send 键即可将消息发出。



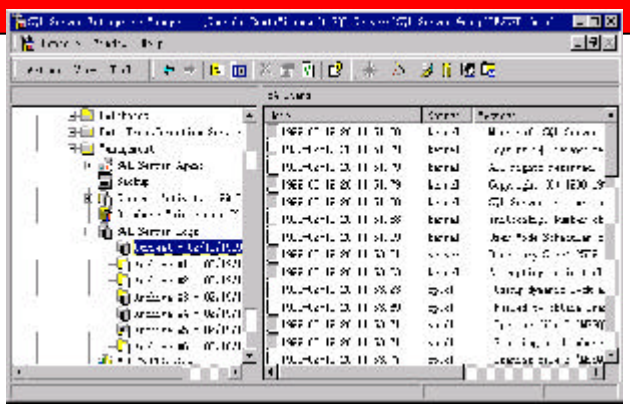
7 通过本例，学会了如何通过 Enterprise Manager 监视 SQL Server 服务器的用户连接情况、并且向用户进程发送消息。

例 监视 SQL Server 的行为 查看日志文件 解

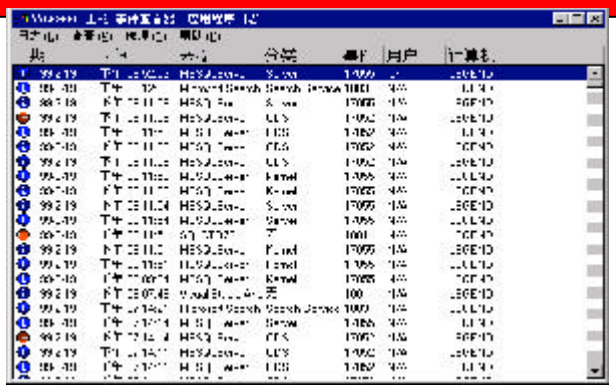
1



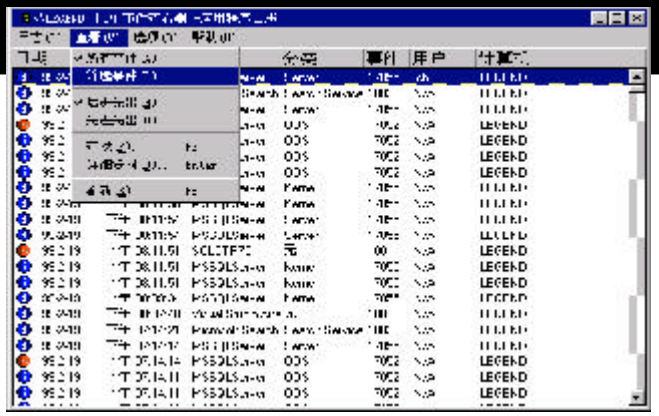
2 在 SQL Server logs 项，单击选择某个要查看的日志名称。日志内容就会出现在右栏窗口中。



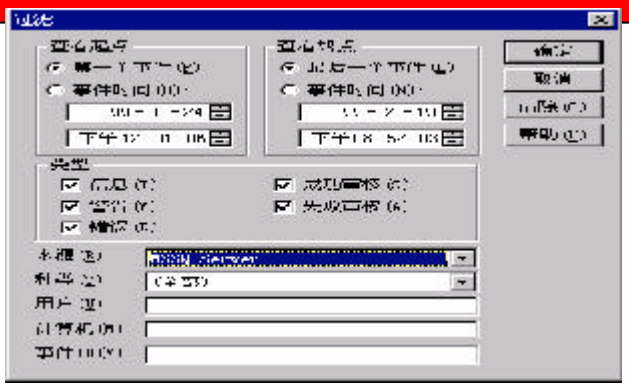
3 用 Windows NT 的事件查看器查看应用程序日志文件。打开启动菜单的程序组－》管理工具》事件查看器。如下图：



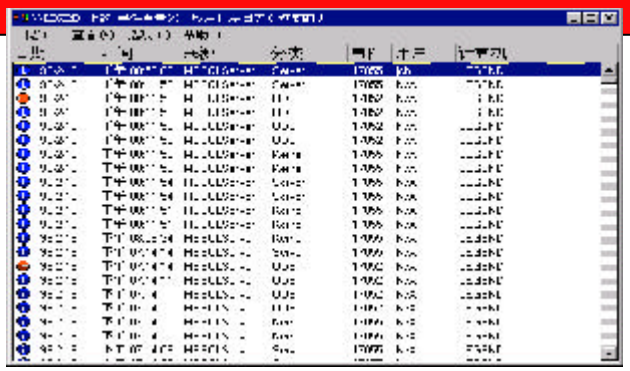
4



5 在弹出对话框中设置如何筛选事件。如筛选 SQL Server 产生的事件。设置来源为 MSSQLServer 如下图，按确认键。



6 下图显示的经过筛选后，事件过滤器中显示的只由 SQL Server 产生的事件。

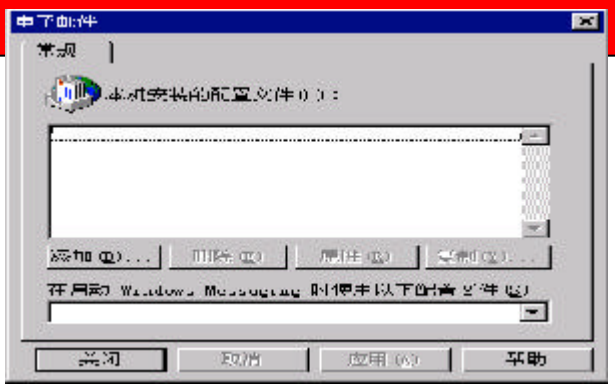


7 通过本例，学会了如何查看 SQL Server 中的日志文件和 Windows NT 的应用程序日志文件。

例 **在 SQL Server 中使用其他工具使用 SQL 邮件功能** 解

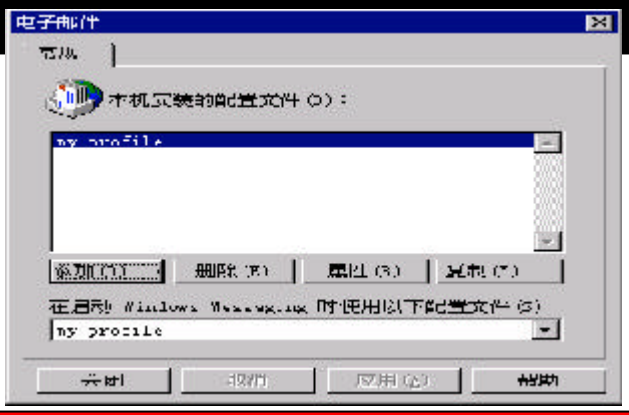
1

2 在打开的对话框中，添加新的邮件配置信息。单击“添加”按钮。

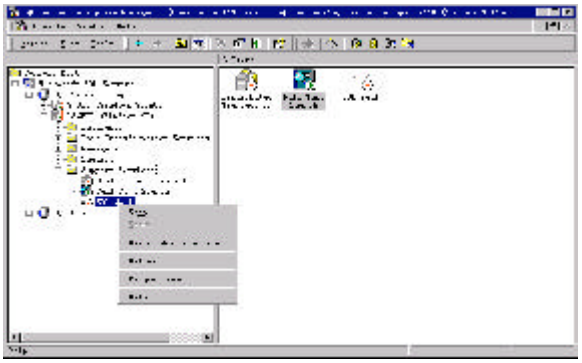


3 根据配置项的提示设置邮件配置信息。直至完成。

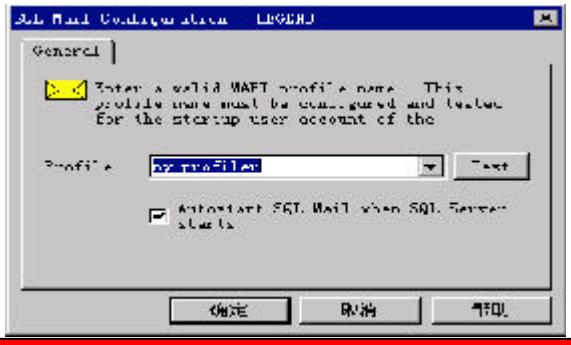
4



5 在 SQL Server 中启用 SQL 邮件功能：在 Enterprise Manager 中展开服务器，展开 Management 项，右键单击 SQL mail，在弹出菜单中，选择 Properties 项，激活属性对话框。



6 在属性对话框中的 Profile 栏中选择刚刚建立的邮件配置文件，Test 键可以测试该邮件功能是否可用。下面的复选框，可以设定是否自动启动 SQL 邮件功能。



7 通过本例，学会了如何设置 SQL Server 中的邮件功能。

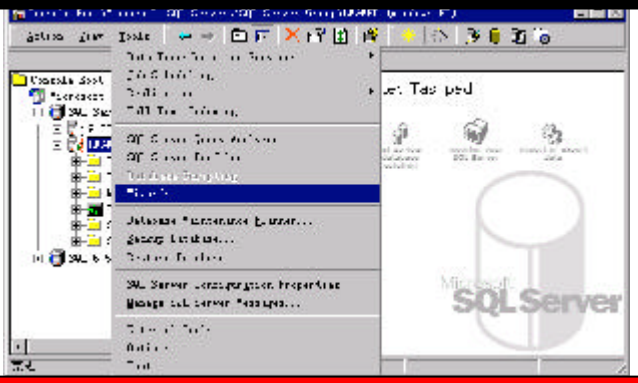
知识卷

例

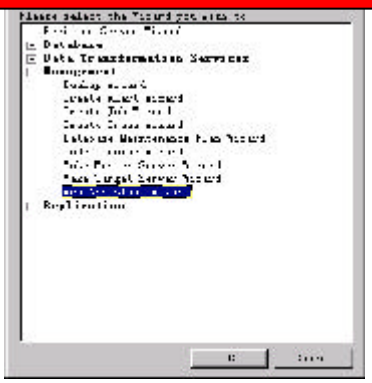
在 SQL Server 中使用其他工具使用 Web 页面向导

解

1



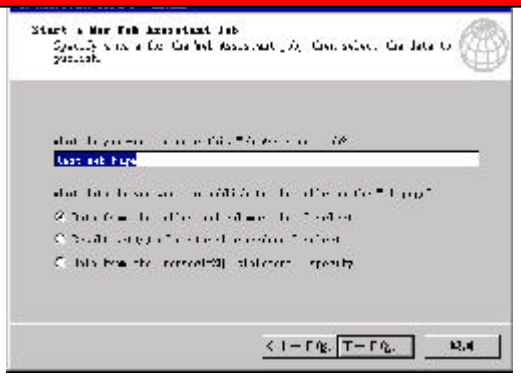
2 在弹出对话框中选择 Management/Web Assistant Wizard。如下图对话框所示，接着单击 OK 继续。



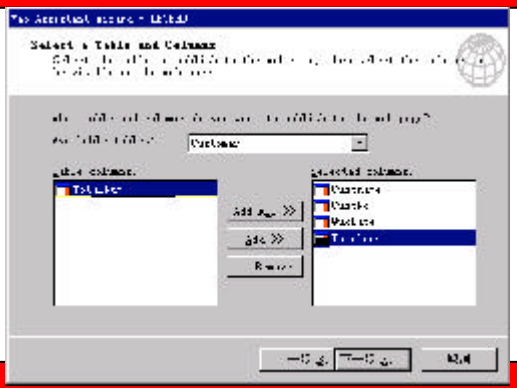
3 下图弹出的是 Web 页面向导的起始对话框，它简要介绍了该向导能够完成的操作。阅读后单击下一步继续。

4

5 在下图所示的对话框中，输入该出版任务的名称。选择是将数据库内的表格出版，还是将 T-SQL 语句的查询结果出版，或者将存储过程的执行结果出版。选定后单击下一步继续。本例中，选定将表格出版。



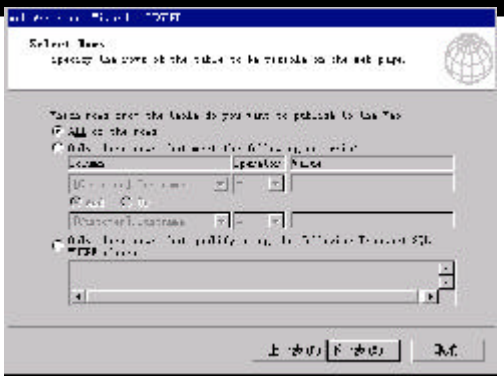
6 在下图所示的对话框中， 选定要出版的表格名称，在左窗栏中选定要出版的子段名称。单击 Add>>按钮，可以添加到右栏后出版。单击 Add All>>后可以将表格内的所有字段全部出版。选定后单击下一步。



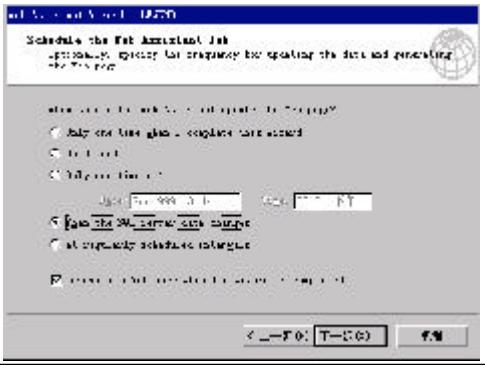
7 操作未完，接下一节。

例 在 SQL Server 中使用其他工具使用 Web 页面向导（续） 解

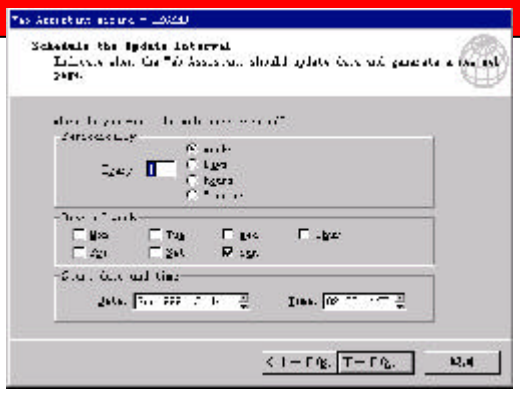
1



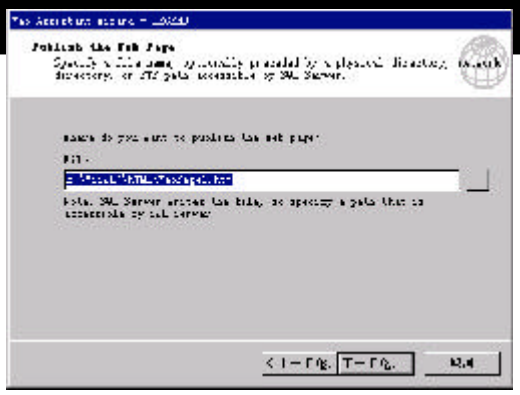
2 在下图所示对话框中，可以选定何时执行出版页面的任务。如向导结束后立刻执行，在向导结束后通过存储过程执行，在今后的某个指定时刻执行，当数据库数据变动时执行，在指定的时刻调度执行。



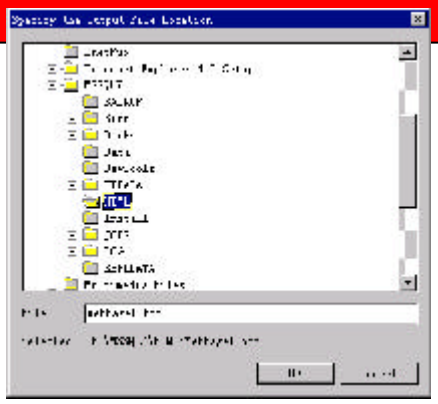
3 如果上图中选定在指定的时刻调度执行，下图对话框中可以设定执行的调度时刻。如每周、每日、每分钟等。选定后单击下一步继续。



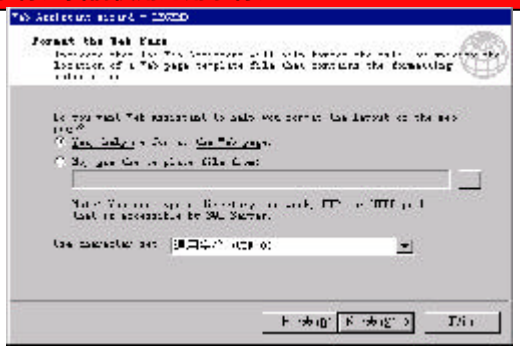
4



5 在上图中如果单击 ··· 按钮，可以弹出如下对话框，在其中输入文件路径和文件名。完成后按确定。



6 在下图对话框中，可以指定让向导帮你完成设置网页样式，或是使用已有的模板文件。一般第一次都选第一个选项。最下方的列表栏中可以选定网页使用的字符。我们使用通用字符。



7 操作未完，接下一节。

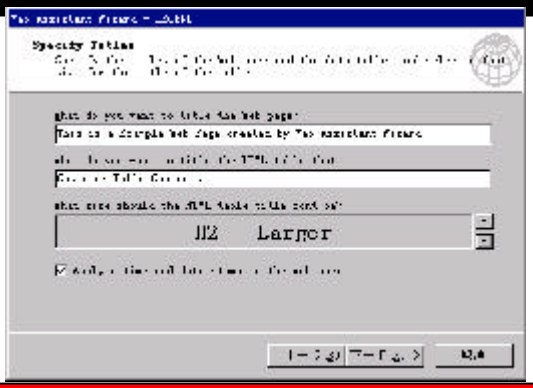
知识卷

例

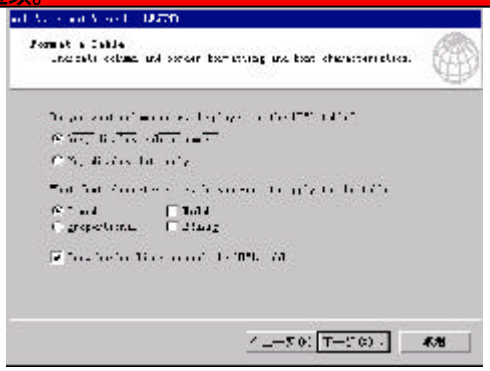
在 SQL Server 中使用其他工具使用 Web 页面向导（续）

解

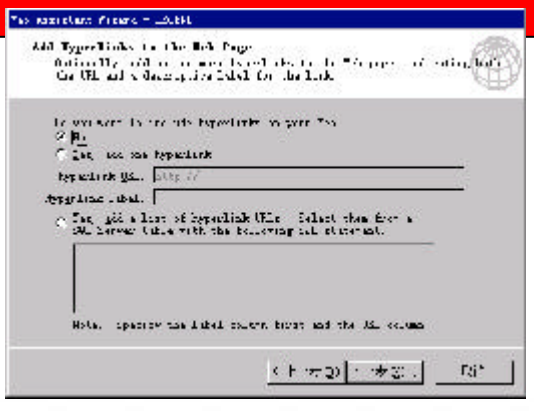
1



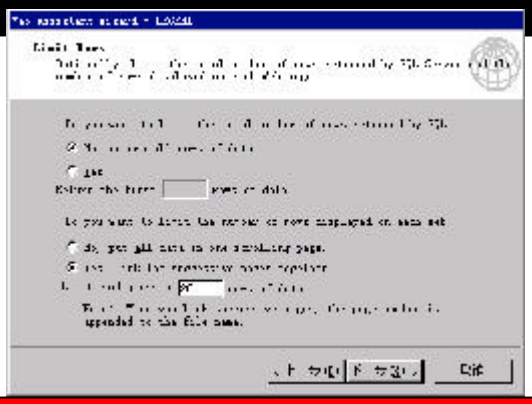
2 在下图所示对话框中，可以选定是否需要显示表格的字段名称，表格内的字体（有斜体、粗体等），以及设定是否需要为表格加上边框。确定后单击下一步继续。



3 在下图中，选定是否需要在网页中添加额外的 URL 链接以充实网页。可以设定不加链接，添加一行链接，从某个表格中选定添加一系列的链接列表。

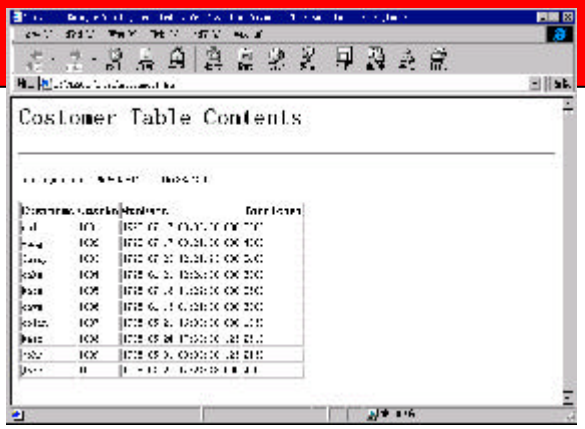


4



5 在下图所示的对话框中，可以在该向导中查看已经进行的设定。如果确定无误后，单击完成。否则，可以按上一步按钮，返回修改。

6 下图所示为在 IE4.0 浏览器中，打开新创建的网页的浏览效果。



7 通过本例，学会了如何使用 SQL Server Enterprise Manager 中的 Web 页面向导，将数据库中的数据以网页的形式出版。

返回总目录

第 三 章

数据库操作

本章导读

作为数据库管理系统的 Microsoft® SQL Server，数据库操作是它最基本的也是最重要的任务。

SQL Server 是以 Client/Server 结构为基础的。通常该结构的数据库系统都要至少包括两个组件：提供给客户端的用户接口；管理和存储数据的数据库结构。本章所要讨论的就是管理和存储数据的数据库结构。用户如何在该结构的基础上创建和管理自己应用的数据库和数据库对象。

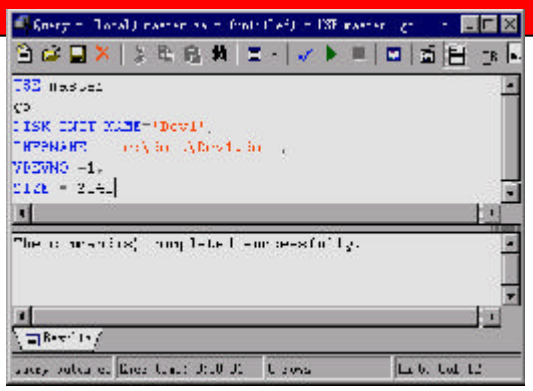
本章的主要内容包括：数据库设备；数据库操作；表格操作；索引操作；视图操作；存储过程操作；触发器操作。

例 数据库设备 添加设备 解

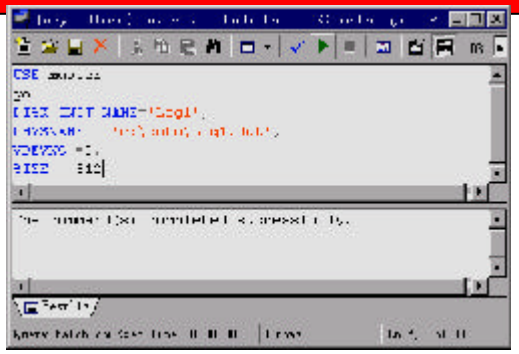
1

```
DISK INIT
NAME = 'logical_name',
PHYSNAME = 'physical_name',
VDEVNO = virtual_device_number,
SIZE = number_of_2k_blocks
[,VSTART = virtual_address]
```

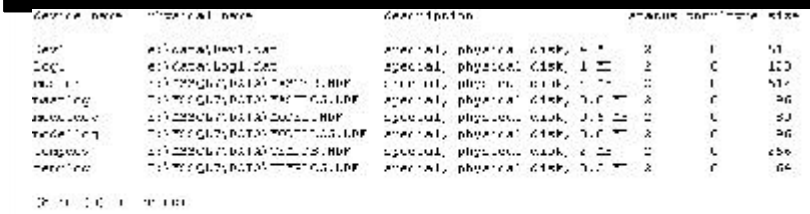
2 例 1：建立一个‘ Dev1 ’设备，大小为 4MB,文件存储位置为'c:\data\Dev1.dat' 预备存放数据库‘ My_db ‘ 的数据部分。在 Query Analyzer 中执行：



3 例 2：建立一个‘ log1 ’设备，大小为 1MB，用来存放‘ My_db ’数据库的日志记录文件。在 Query Analyzer 中执行：



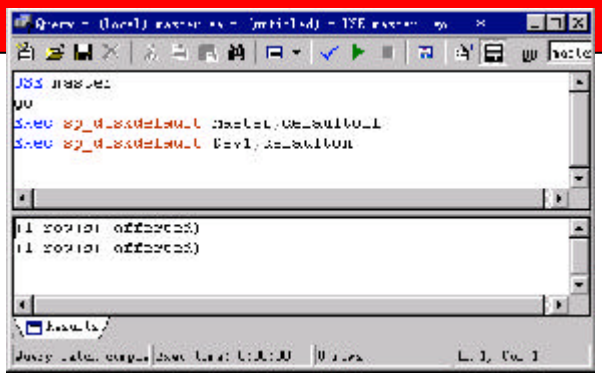
4



5 更改缺省设备,其句法如下图所示：

```
sp_diskdefault 'device_name', {defaulton|defaultoff}
  注:   devic_name : 设备名称;
        defaulton: 更改设备为默认设备;
        defaultoff:更改设备为非默认设备;

        defaulton 和 defaultoff二者选一
```

6 例：更改 Master 系统设备为非默认设备。设定 Dev1 为系统设备。



7 通过本例，我们学会了如何为数据库创建设备。以及如何设置设备为默认设备或者非默认设备。

例

数据库设备
扩增设备空间、删除设备

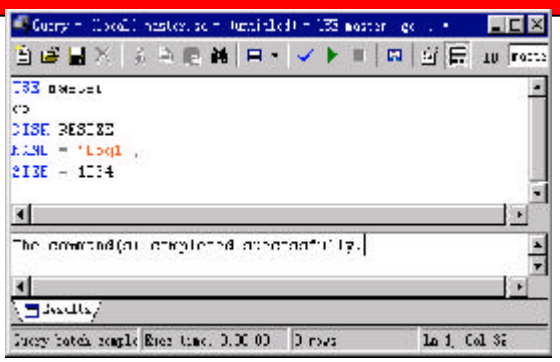
解

1

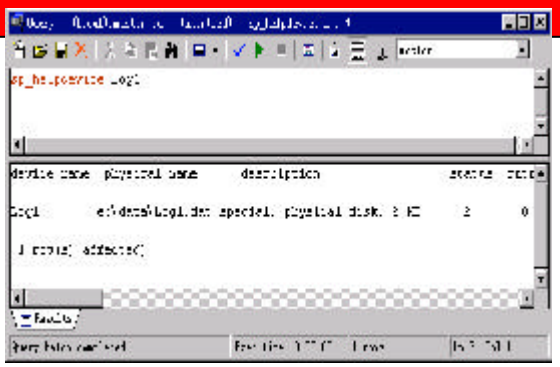
```
DISK RESIZE
NAME = 'logical_name',
SIZE = final_size
```

注：　logical_name ： 设备名称；
　　final_size　 ： 增大容量后的设备大小 单位是2K

2 例：增大上例中的 Log1 设备，原来大小为 1MB，增大至 2MB。在 Query Analyzer 中执行，如下图所示：



3 再用 sp_helpdevice 查看设备信息，发现 Log1 设备信息已被修改。如下图：



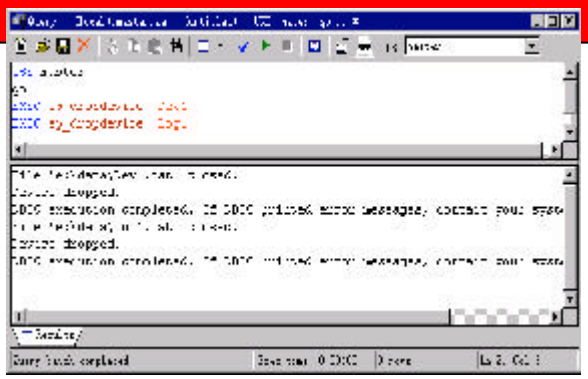
4

```
sp_dropdevice 'logical_device_name'
```

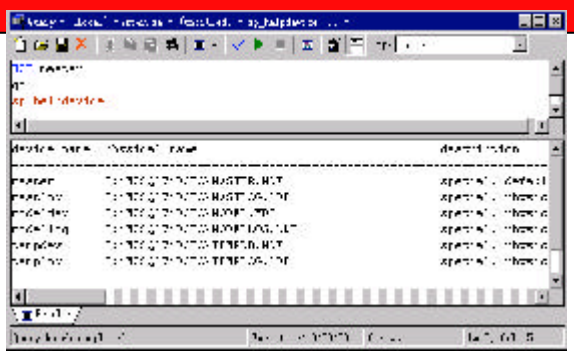
注： logical_device_name 位于删除的设备名称。

只有sa(系统管理员才有删除设备的权限。

5 例：删除‘ Dev1 ’、‘ Log1 ’设备。在 Query Analyzer 中执行 sp_dropdevice 语句，结果如下图所示：



6 执行完后，通过 sp_helpdevice 查看，发现‘ Dev1 ’、‘ Log1 ’两个设备的纪录都已被删除。



7 通过本例，学会了如何扩增设备的容量，以及如何在系统中删除设备。

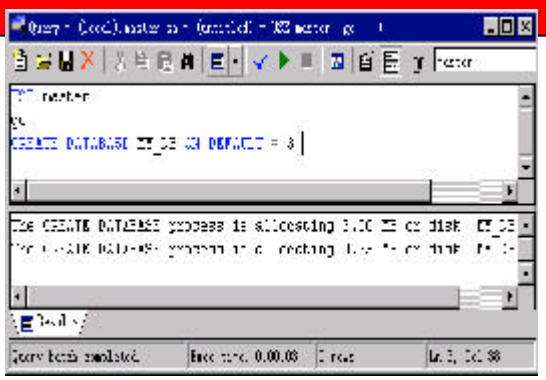
例 **数据库操作**
创建数据库 解

1

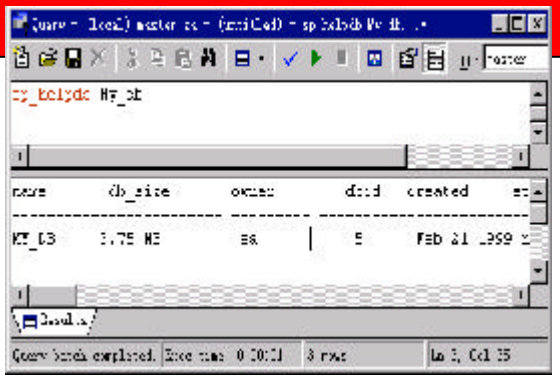
```
CREATE DATABASE database_name
       [ON DEFAULT|ON database_device]{ = size}
       [,database_device = size...]
       [LOG ON database_device = size ]
       [,database_device =size ]
       [FOR LOAD]
```

注：
database_name:建立的数据库名称
ON DEFAULT |ON database_device:选择使用默认设备还是指定的设备
LOG ON database_device = size :将日志文件存放在指定设备上
FOR LOAD :SQL Server会在设备上划分出空间，但并不立即建立数据库
而是要等到备份数据库存时才建立数据库结构

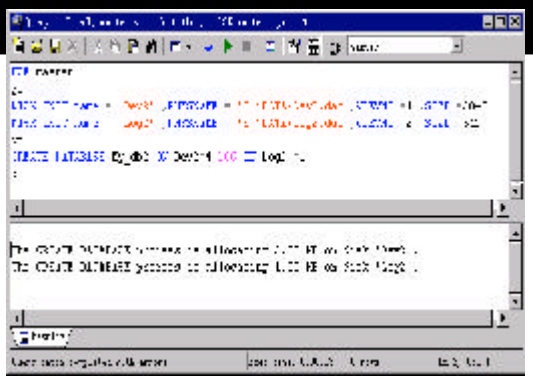
2 **例：建立数据库‘ My_db ’，将数据和日志文件都存储在默认设备上。如下图所示：**



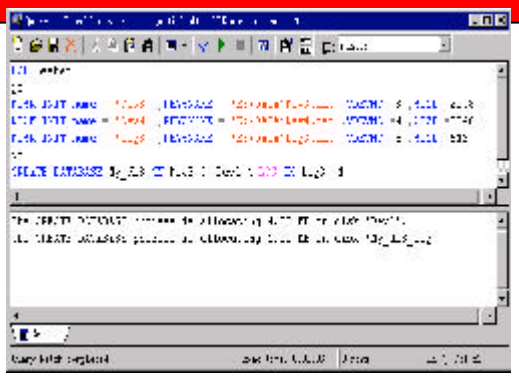
3 **可以通过 sp_helpdb 查看系统中的数据库信息。如下图所示：**



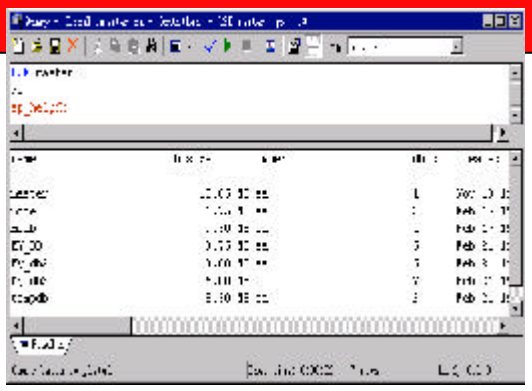
4



5 例三：建立数据库 My_db3,数据库的数据部分分别存放在设备‘ Dev3 ’、‘ Dev4 ’，日志文件存放在设备‘ Log3 上。’



6 通过 sp_helpdb 查询系统内所有数据库的信息。如下图所示：



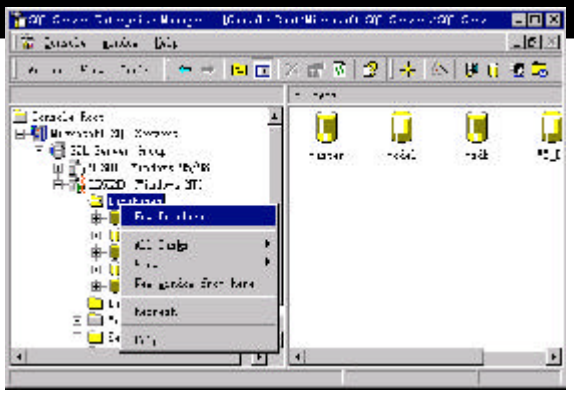
7 通过本例，学会了如何在 Query Analyzer 中通过执行 Transact-SQL 语句创建数据库。

例

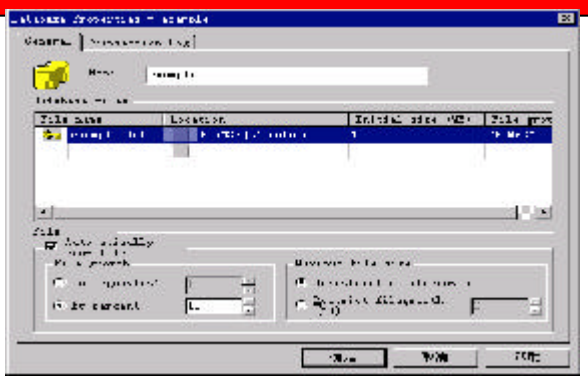
数据库操作
通过 Enterprise Manager 创建数据

解

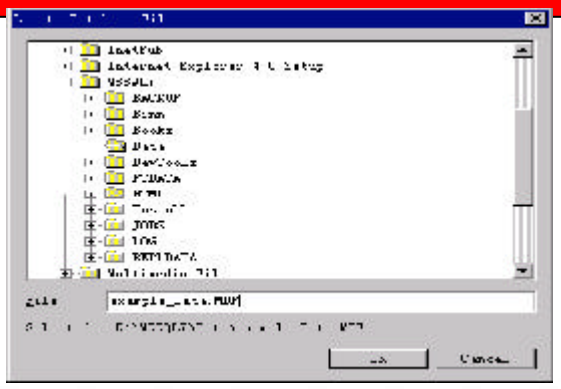
1



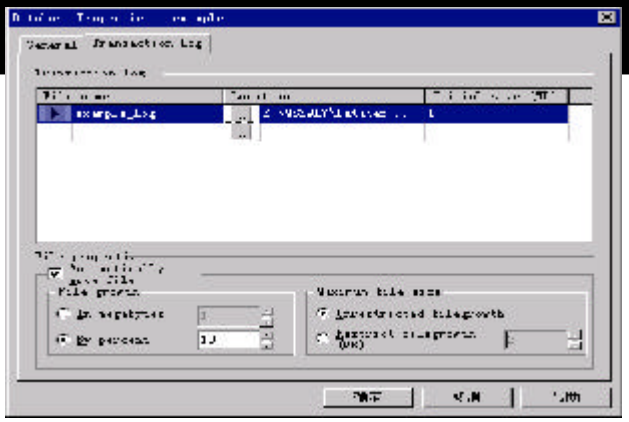
2 在弹出的对话框中输入数据库的名称， 如‘ example ’。在 database files 栏中可以输入数据库的文件名称， 选择文件的初始大小。



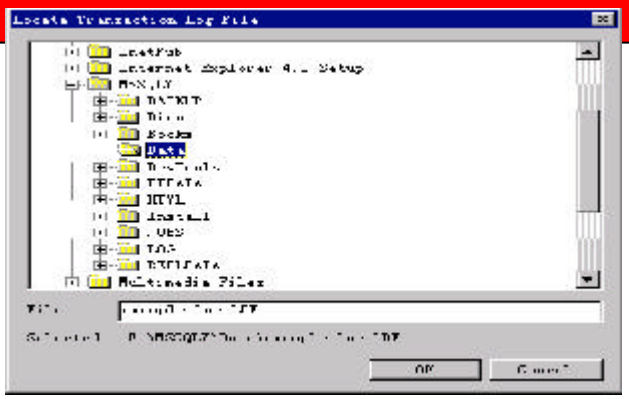
3 在 General 页面中，单击 Database files 中 Location 栏中的 … 按钮，可以设定存放数据文件的路径。弹出如下图对话框，在其中输入文件名。



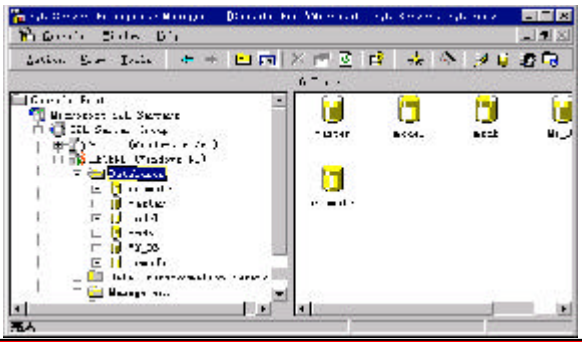
4



5 在 Transaction Log 栏中，可以输入日志文件的名称路径。设置日志文件的初始大小。单击 ... 按钮，可以打开对话框用于设定日志文件的路径。



6 所有的设置都设置正确后， 单击确定键，就完成了数据库的创建工作。在 Enterprise Manager 中浏览数据库时就能看到新的数据库，并可以对其进行操作了。



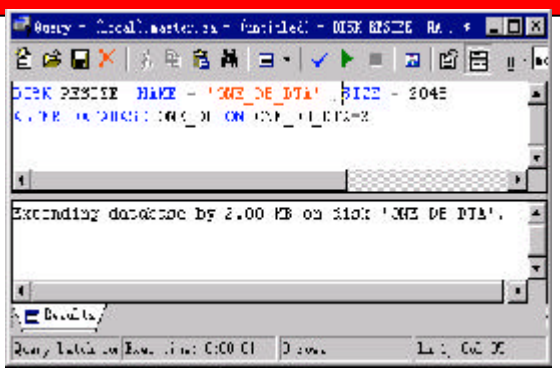
7 通过本例，学会了如何通过 Enterprise Manager 来创建数据库。

例 数据库操作 增加数据库容量 解

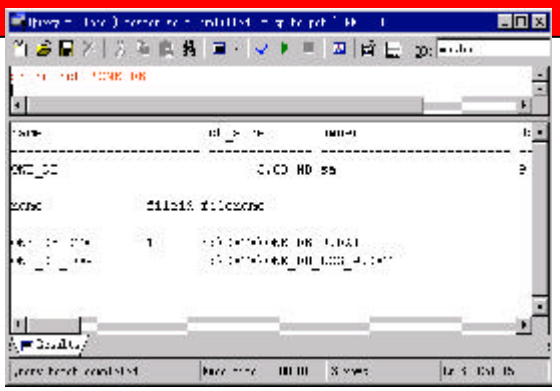
1

```
ALTER DATABASE database_name
        [ON DEFAULT | ON database_device ] { = size }
        [,database_device = size ...]
        [FOR LOAD]
```

2 **例 1：数据库 One_db 原来的大小为 2MB，想增加容量 2MB，执行如下 T-SQL 语句。**



3 **执行 sp_helpdb 查询执行结果如下图：**



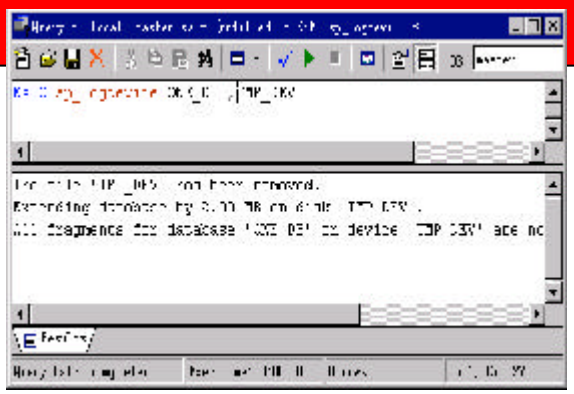
4

```
sp_logdevice database_name,device_name
```

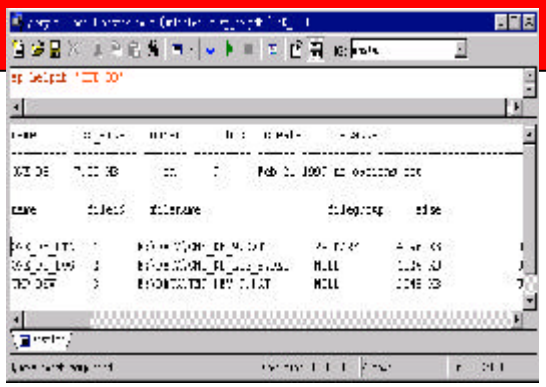
注：　database_name　为您要操作的数据库名称

　　device_name　　为您要更改的设备名称

5 TMP_DEV 为数据库的数据设备之一，下例语句将把该设备更改为存放日志文件。



6 使用 sp_helpdb 语句查询执行结果如下图所示：



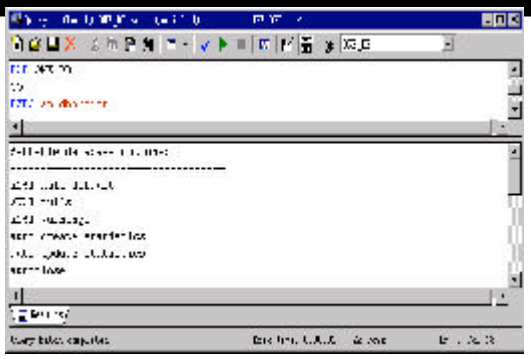
7 通过本例，学会了如何使用 T-SQL 语句增添数据库的容量。

例

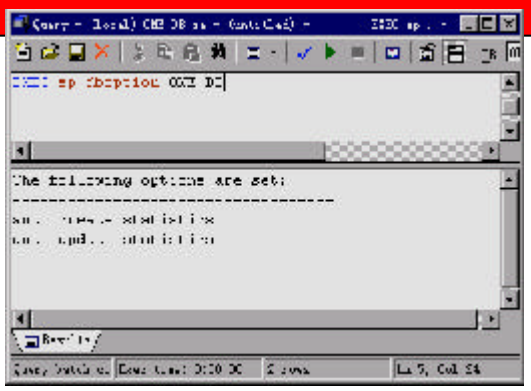
数据库操作
查看和设置数据库选项

解

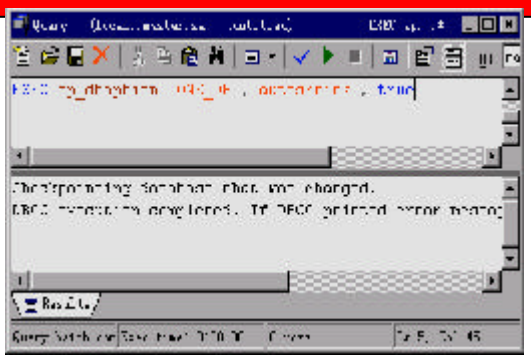
1



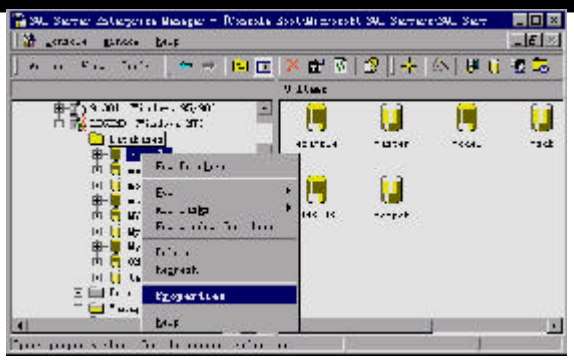
2 要显示某个数据库已经设置的属性，可以在 sp_dboption 后直接跟数据库的名称。如下图，显示数据库 ONE_DB 中已经设置的选项。



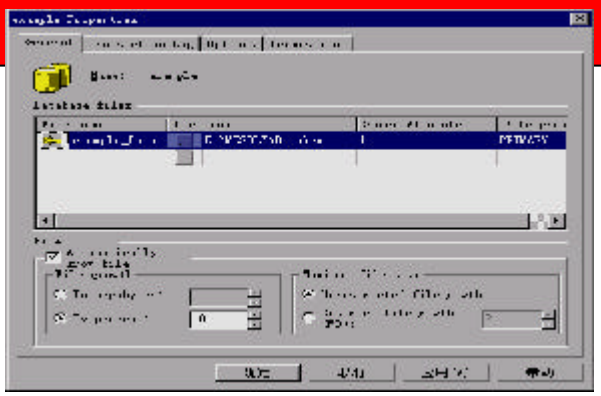
3 要在数据库中设置某个数据库选项为真或者为假，可以采用 sp_dboption database_name option_name TRUE|FALSE 的格式。



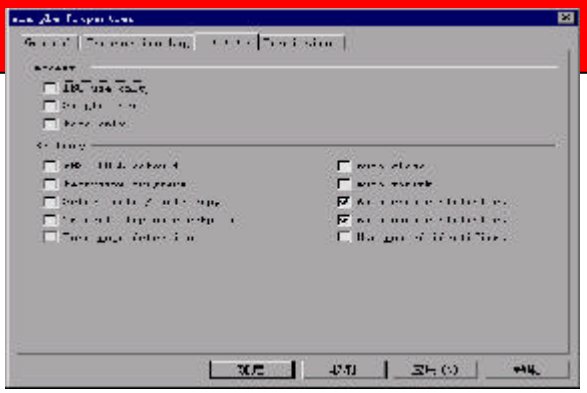
4



5 在弹出的数据库属性对话框中，选择 Options 页面。



6 在 Option 页面中可以直接设置要设置的数据库选项。如下图所示：



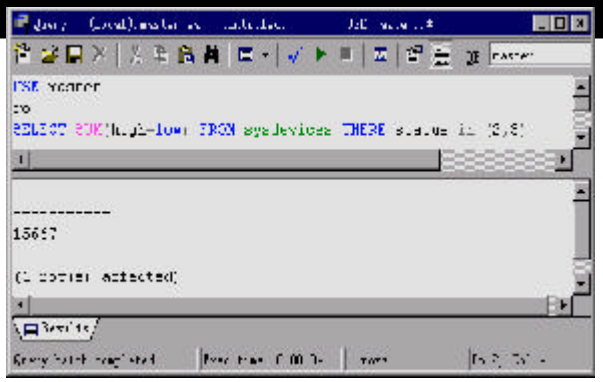
7 通过本例，学习了如何使用 T-SQL 中的 sp_dboption 语句和 Enterprise Manager 来设置数据库选项。

例

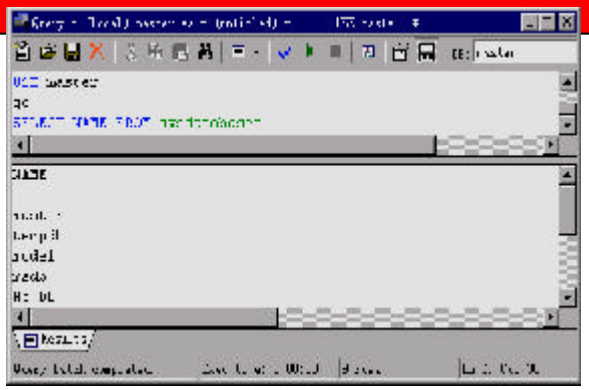
数据库操作
查看有关数据库的信息

解

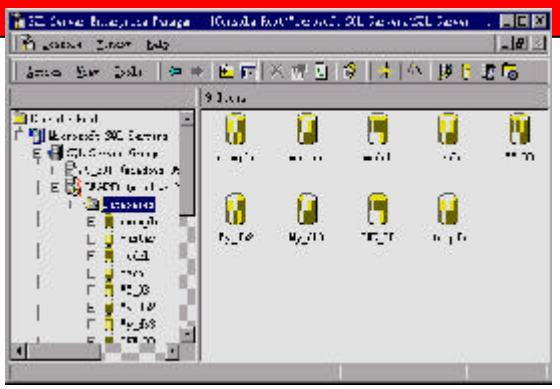
1



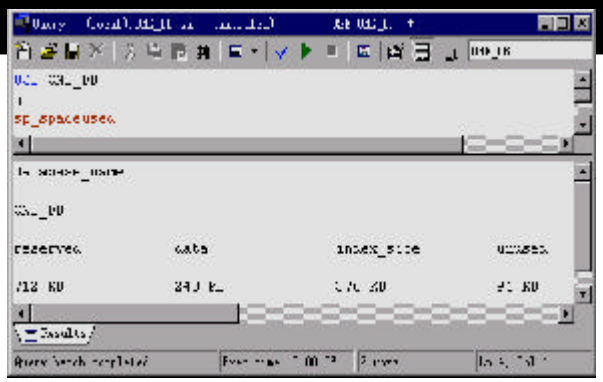
2 查看目前系统中有哪些数据库,在 Query Analyzer 中执行下述语句，返回结果是系统中的所有数据库名称。



3 上述语句与在 Enterprise Manager 中直接查看数据库信息时列出的数据库名称完全相同。

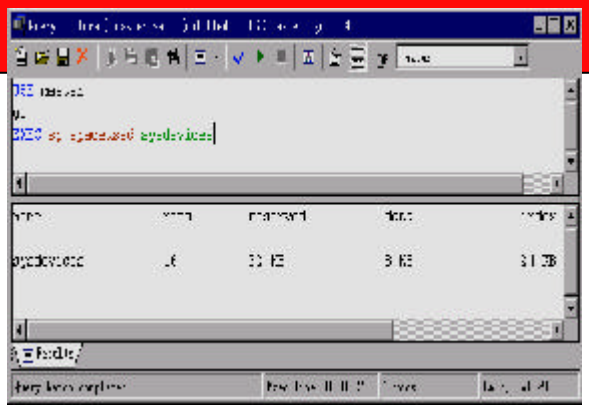


4



5 使用带有@updateusage 参数的 sp_spaceused 语句执行结果如下图：可以发现该结果与上图中的结果有所不同。

6 查看 master 系统数据库内，sysdevices 系统表格的空间分配情形。



7 通过本例，学会了如何使用 Query Analyzer 查看数据库的有关信息。实质上，熟悉 SQL Server 中的数据表格的内容和作用对于有效的使用 SQL Server 是非常有益的。

例

数据库操作
删除数据库

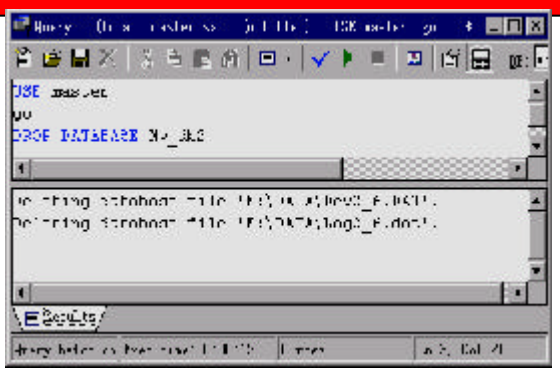
解

1

```
DROP DATABASE database_name [,database_name...]
```

注： database_name: 想删除的数据库名称

2 如在 Query Analyzer 中执行下述命令删除数据库 My_db2。执行的效果如下图所示：



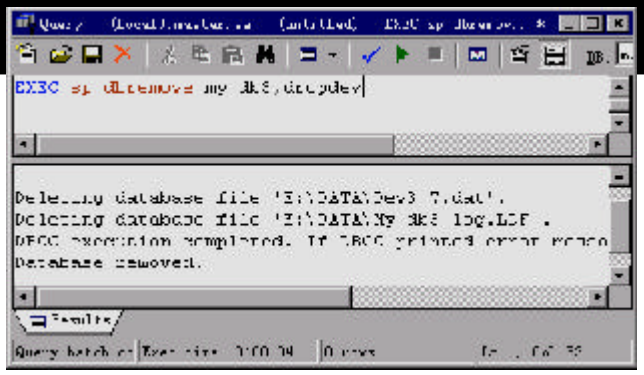
3 删除数据库的 T-SQL 句法 2 如下图所示：

```
sp_dbremove database_name [,dropdev]
```

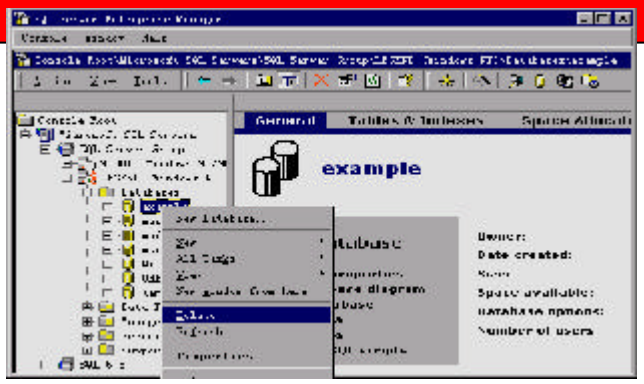
注： database_name：想删除的数据库名称

dropdev：打上此参数，假如在设备上只有一个数据库，又一并删除该设备在SQL Server中的定义，但仍需手动删除操作系统下的设备文件。

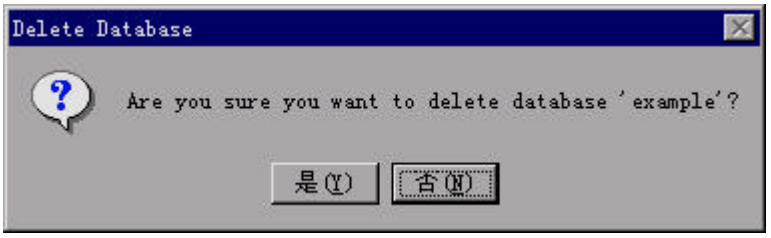
4



5 在 Enterprise Manager 中删除数据库：右键单击要删除的数据库，在弹出菜单中选择 Delete 功能.



6 选择 Delete 命令后，弹出确认对话框如下图所示，选择‘是（Y）’即可删除选定的数据库。



7 通过本例，学会了如何使用 Transact-SQL 语句或者 Enterprise Manage 删除数据库。

例

数据库操作
为数据库添加文件

解

1

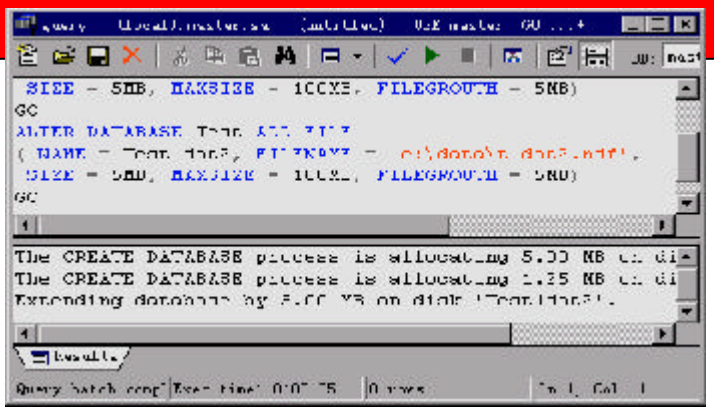
```
ALTER DATABASE database_name
ADD FILE
(NAME = logical_file_name
 [, FILENAME = 'os file name' ]
 [, SIZE = size]
 [, MAXSIZE = { max size | UNLIMITED } ]
 [, FILEGROWTH = growth_increment] )
```

注：
- database name：数据库名称
- logical_file_name：逻辑文件名
- os_file_name：实际的操作系统文件名称
- size：文件大小
- max_size：文件大小的最大值；UNLIMITED：文件大小不受限制
- groth_increment：文件每次增大的增量

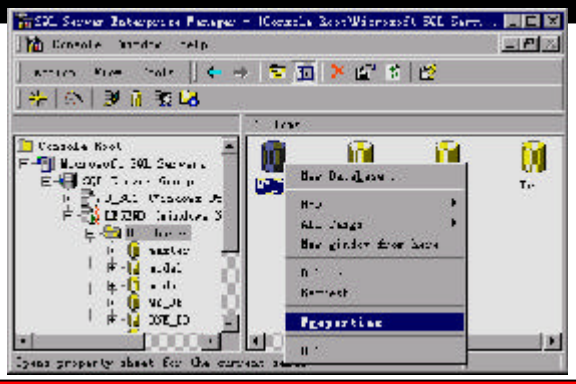
2 例：通过上述语句往数据库中添加文件。

```
USE master
GO
CREATE DATABASE Test ON
( NAME = Test1dat1,  FILENAME = 'e:\data\t1dat1.ndf',
 SIZE = 5MB, MAXSIZE = 100MB, FILEGROWTH = 5MB)
GO
ALTER DATABASE Test ADD FILE
( NAME = Test1dat2, FILENAME = 'e:\data\t1dat2.ndf',
 SIZE = 5MB, MAXSIZE = 100MB, FILEGROWTH = 5MB)
GO
```

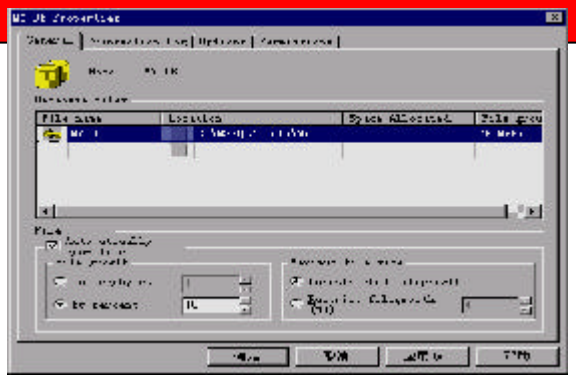
3 在 Query Analyzer 中执行上例，结果如下图所示：



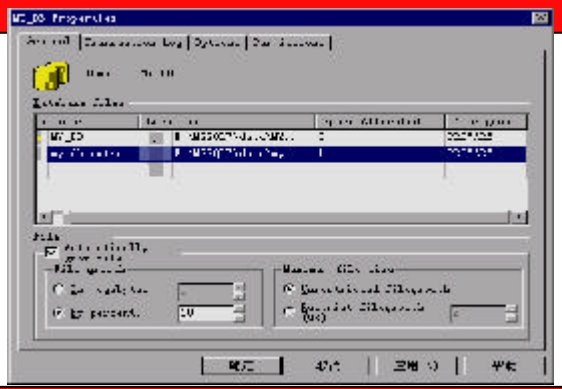
4



5 弹出的是数据库的属性对话框，在 General 页面中，可以设置数据文件属性，添加数据文件。要添加日志文件，选择 Transaction Log 页面。



6 鼠标单击 Database Files 中的最下面一行的空行，在此可以填写需要新增文件的属性，如文件名、初始大小、路径等。输入完毕按确认键就可以了。



7 通过本例，学会了如何使用 T-SQL 语句和在 Enterprise Manager 中为数据库增加文件。

例

数据库操作
添加文件组

解

1

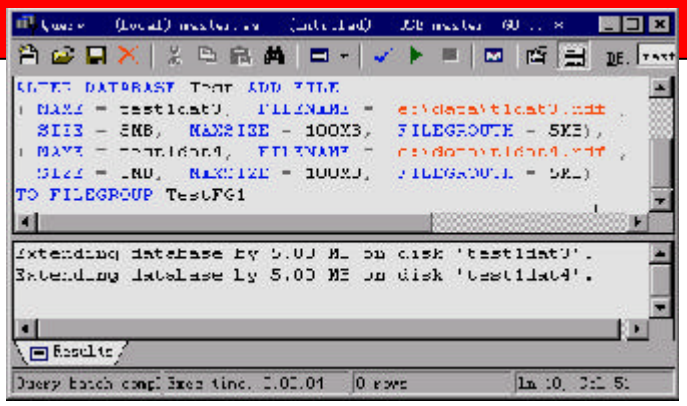
```
ALTER DATABASE database_name
ADD FILEGROUP filegroup_name

ALTER DATABASE database_name
ADD FILE
(NAME = logical_file_name
 [, FILENAME = 'os_file_name' ]
 [, SIZE = size]
 [, MAXSIZE = { max_size | UNLIMITED } ]
 [, FILEGROWTH = growth_increment] ) [, ...n]
TO FILEGROUP filegroup_name
```

注：　filegroup_name　:文件组名称
其余参数意义同上例。
第一句ALTER DATABASE用于在数据库中新建文件组
第二句ALTER DATABASE用于扩展数据库往新组文件组中加入文件

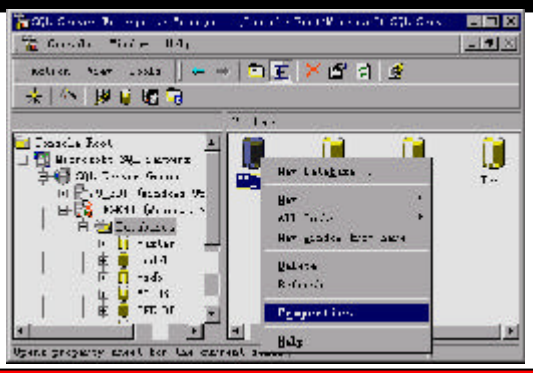
2 下图所示的例子在数据库中创建新的文件组后，再往其中添加文件。

```
USE master
GO
ALTER DATABASE Test
ADD FILEGROUP TestFG1
GO
ALTER DATABASE Test ADD FILE
( NAME = test1dat3,  FILENAME = 'e:\data\t1dat3.ndf',
  SIZE = 5MB,  MAXSIZE = 100MB,  FILEGROWTH = 5MB),
( NAME = test1dat4,  FILENAME = 'e:\data\t1dat4.ndf',
  SIZE = 5MB,  MAXSIZE = 100MB,  FILEGROWTH = 5MB)
TO FILEGROUP TestFG1
```

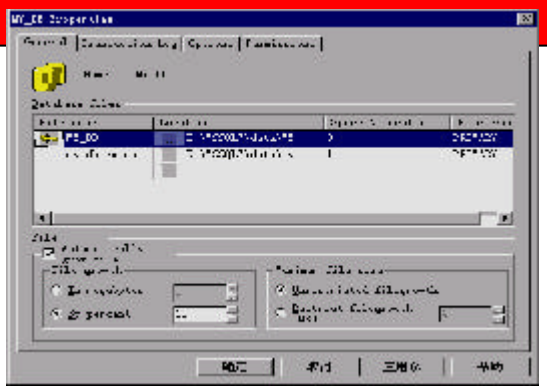
3 在 Query Analyzer 中执行，结果如下图所示：



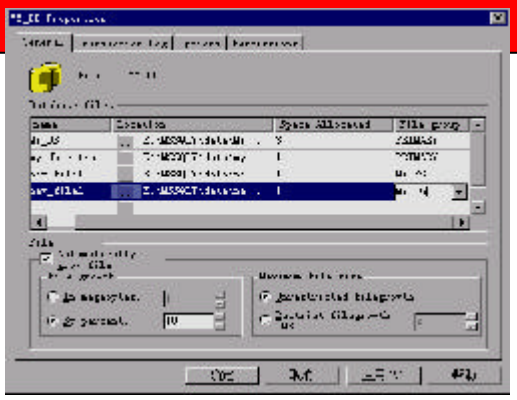
4



5 弹出的是数据库的属性对话框，选择 General 页面。



6 添加文件的方法与上例基本相同，只是注意要在 File Group 栏中输入的文件组名称。



7 通过本例，学会了如何使用 T-SQL 语句和 Enterprise Manager 在数据库中新建文件组，并且往该文件组中添加文件。

例

数据库操作
数据库的附接和分离

解

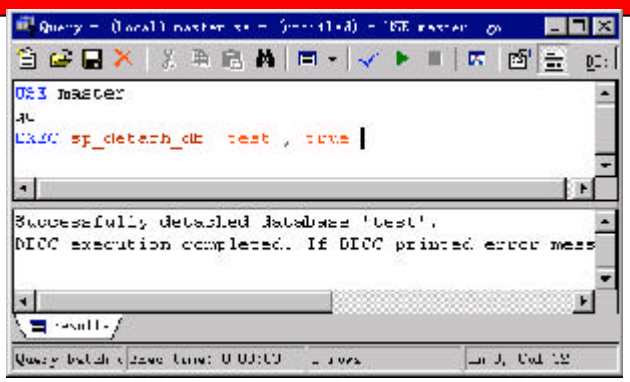
1

```
sp_detach_db [@dbname =] 'dbname'
    [, [@skipchecks =] 'skipchecks']
```

注：　dbname　　:为要分离的数据库名称，缺省值为NULL

　　skipchecks :为字符串，true 或者 false.
　　如果为true，那UPDATE STATISTICS将被跳过
　　如果为false，那将执行UPDATE STATISTICS过程
　　执行这个过程是在分离前进行数据库表格统计的更新

2 下图中在 Query Analyzer 中执行分离 Test 数据库的操作。在分离数据库前请先通过 sp_helpdb 或者 Enterprise Manager 记下该数据库的几个文件的存放位置。



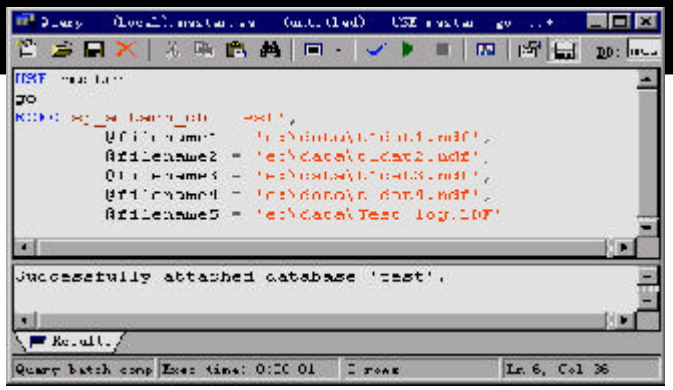
3 下图所示为向 SQL Server 服务器附接数据库的 T-SQL 语句的句法。

```
sp_attach_db [@dbname =] 'dbname',
    [@filename1 =] 'filename_n' [,...16]
```

注：　dbname　　:为要附接上的数据库的名称

　　@filename1 : 是数据库文件的带路径的文件名。它可以是最大260个字符的的字符串，缺省值为NULL。附接数据库最多可有16个文件。从@filename1到@filename16。这些文件中必须至少包含Primary文件。

4



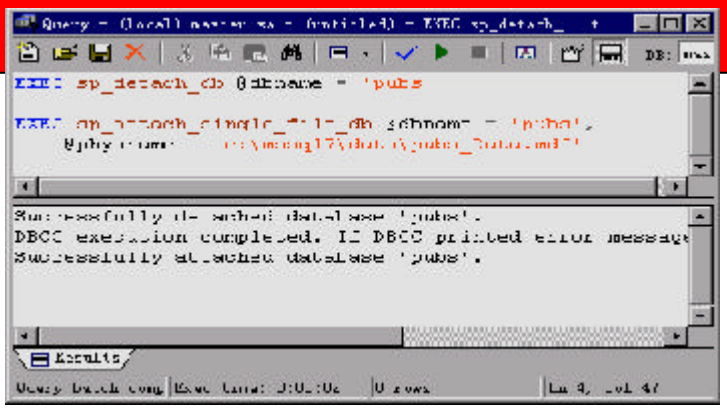
5 下图所示为附接单文件数据库的 T-SQL 语句句法。

```
sp_attach_single_file_db [@dbname =] 'dbname',
   [@physname =] 'physical_name'
```

注： dbname ：为要附接上的数据库的名称

@physname ：是数据库文件的带路径的文件名。它可以是最大260个字符的的字符串，缺省值为NULL。

6 在 Query Analyzer 中执行附接单文件数据库的结果如下图所示：



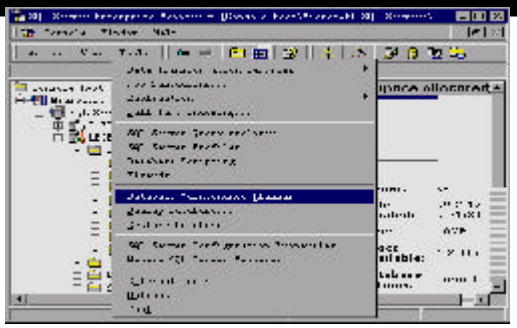
7 通过本例， 学会了如何在 SQL Server 中分离和附接数据库。采用这种方法复制和转移数据库是非常方便的。

例

数据库操作
数据库的维护向导

解

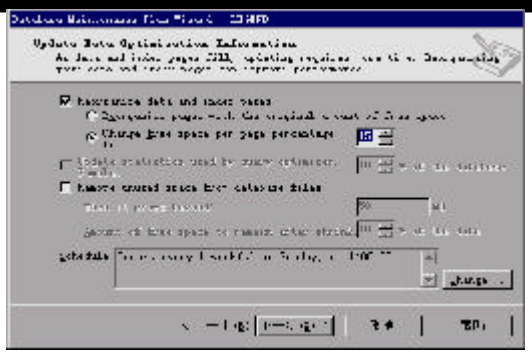
1



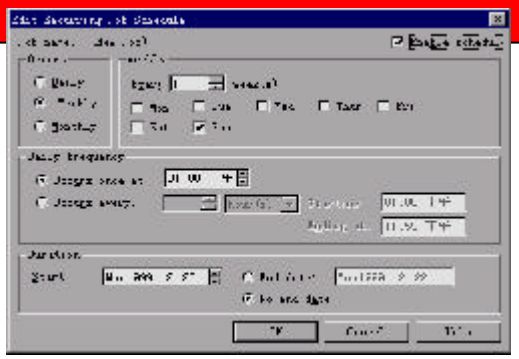
2 数据库的向导启动界面如下图所示，它介绍了数据库维护向导能完成的功能。

3 在接下来弹出的对话框中选择要进行维护的数据名称。有四种选项：所有数据库、所有系统数据库、所有的用户数据库、用户指定数据库。

4



5 在单击 Change 键后弹出的对话框中可以设置调度时间，如下图所示：



6 设置完更新数据优化信息后，进入设置数据一致性检查对话框。如下图所示：

7 后面将继续介绍数据库维护向导。

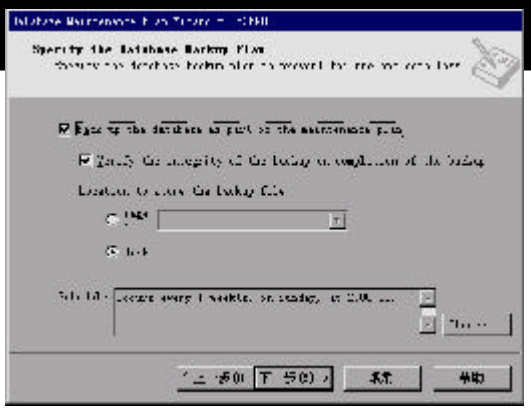
知识点

例

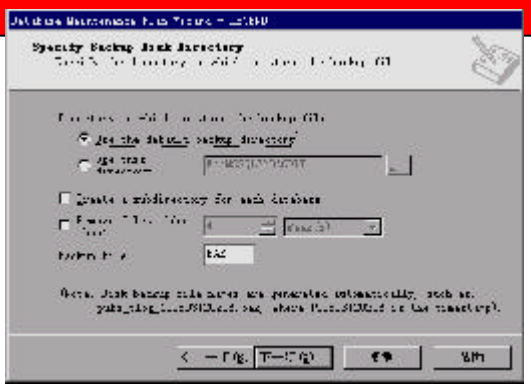
数据库操作
数据库的维护向导（续）

解

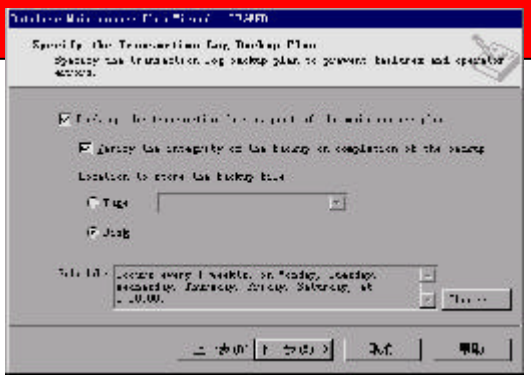
1



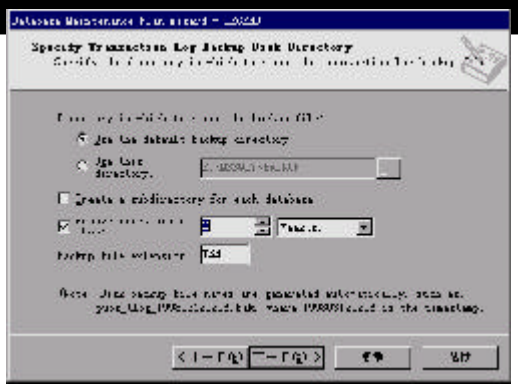
2 在指定备份磁盘目录的对话框中可以设置备份文件的存放路径，以及有关备份操作的选项。



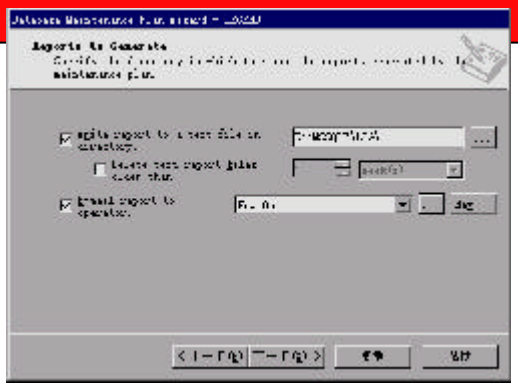
10 然后，可以设置日志文件的备份选项，如下图：



11



12 设置完日志文件，可以设置如何将此次任务的执行结果报告操作员。



13 最后一个对话框可以设置将任务执行结果记录在本地机上的msdb.dbo.sysdbmaintplan_history表格中，或者某个网络服务器上。

14 通过本例，学会了如何使用数据库维护向导来设置数据库维护的任务。

例

表格操作
设计用户数据类型

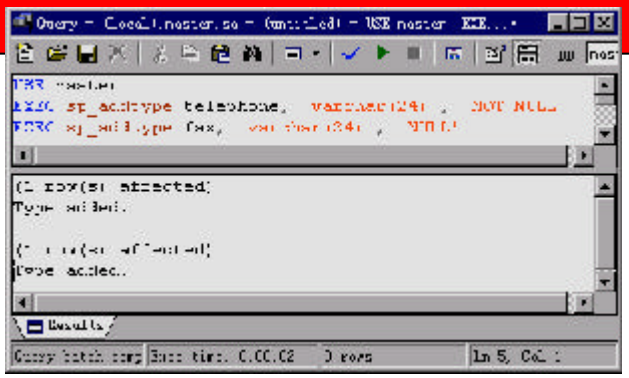
解

1

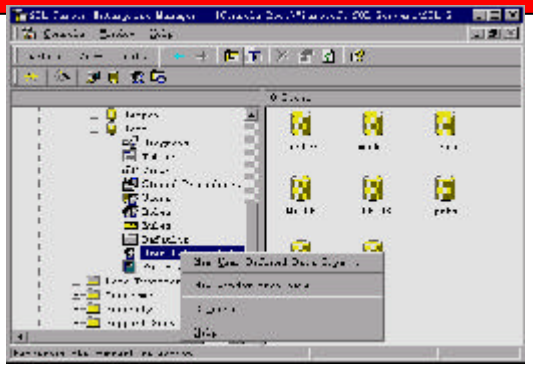
```
sp_addtype [@typename =] type,
    [@phystype =] system_data_type
    [, [@nulltype =] 'null_type']
```

注：
- type：用户自定义数据类型的名称
- system_data_type：作为用户自定义类型基础的系统数据类型
- null_type：指出数据类型是否允许为空

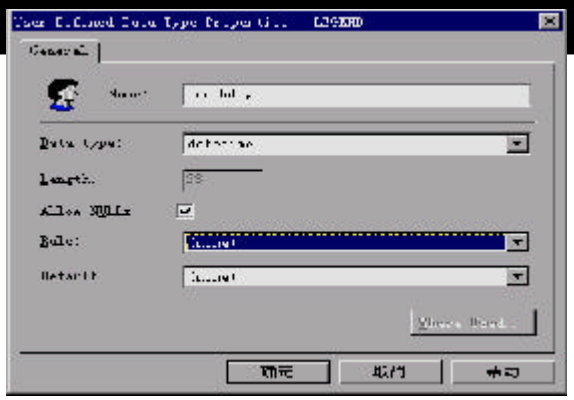
2 下图显示在 Query Analyzer 中增加 Telephone 和 fax 两种数据类型的结果。



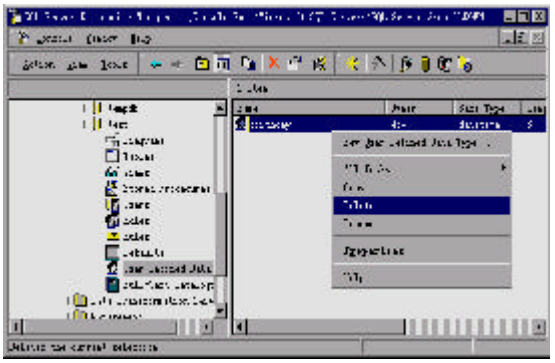
3 通过 Enterprise Manager 创建数据类型。在 Enterprise Manager 中展开服务器，展开 Database 项，展开要操作的数据库，鼠标右键单击 User Defined Data types，在弹出菜单中选择 new User Data type。



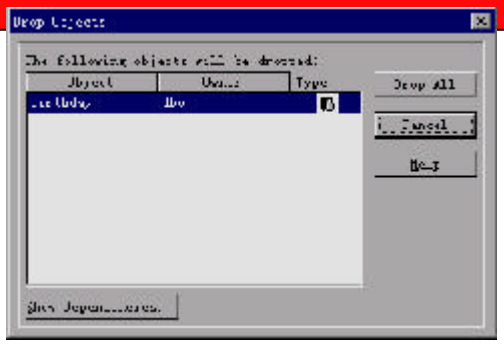
4



5 删除用户自定义类型的方法，也有两种。T-SQL 中的 sp_droptype typename；在 Enterprise Manager 中右键单击用户自定义的数据类型，在弹出菜单中选择 Delete。



6 在弹出的对话框中可以单击 Show Dependencies 查看数据库内是否有对象依赖此数据类型，如果没有，就可以按 Drop All 键删除该数据类型。



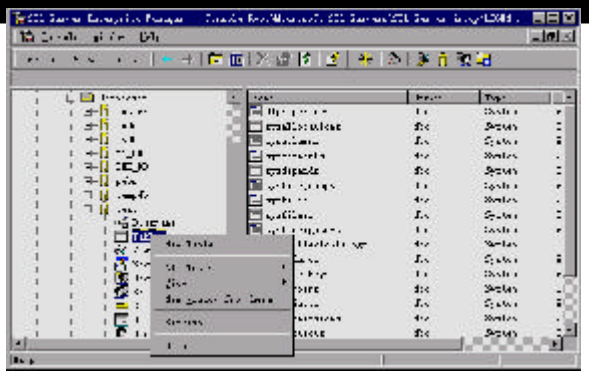
7 通过本例，学会了如何增加或者删除用户自定义的数据类型。数据类型是设计表格的基础。

例

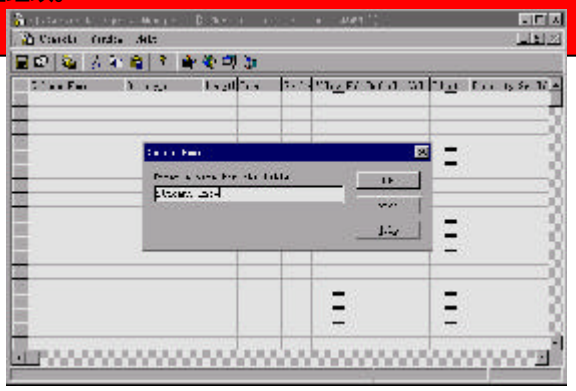
表格操作
创建表格

解

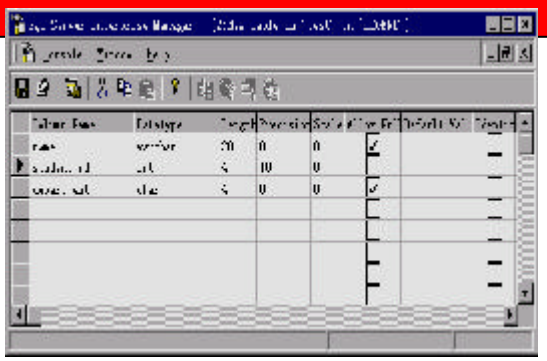
1



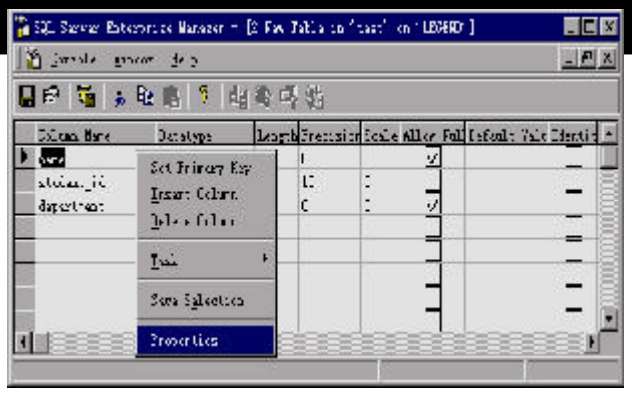
2 在弹出对话框中，输入新建表格的名称，如本例中的 Student_info。按 OK 按钮继续。



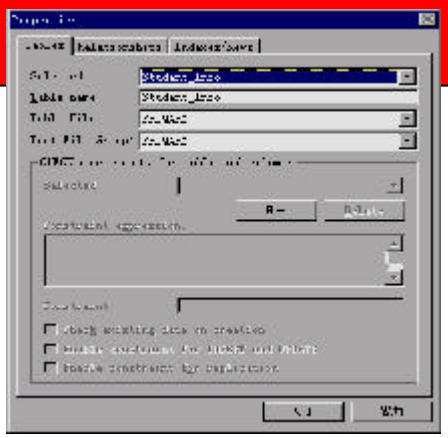
3 在下图中输入表格的各个字段（行）的定义，包括字段的名称、数据类型、长度等其他一些类型。



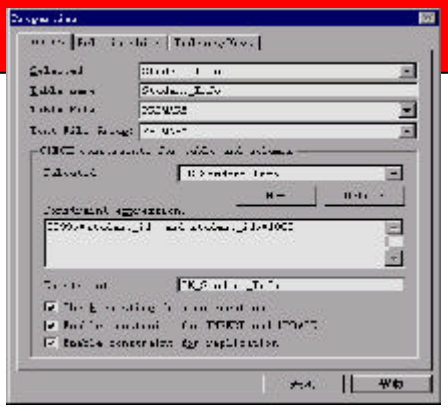
4



5 可以在属性对话框中选择表格存放的数据库文件组。



6 单击 New 按钮，可以添加数据合法性检查表达式，最后按确定键即可。



7 通过本例，学会了如何在 Enterprise Manager 中创建数据库表格。这也可以通过 T-SQL 语句 CREATE TABLE 完成。

例

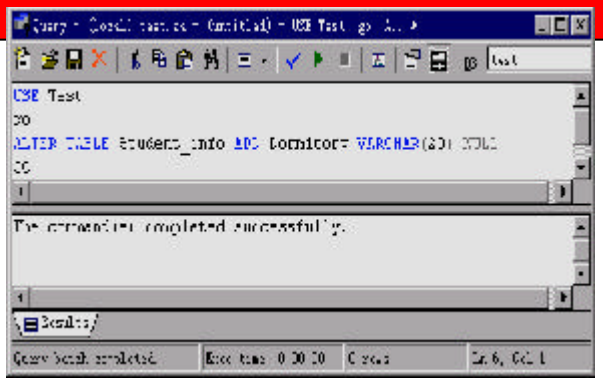
表格操作
增加、删除、修改字段

解

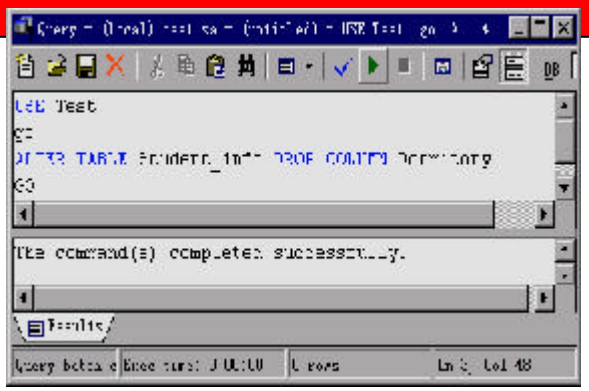
1

```
ALTER TABLE table
{   [ALTER COLUMN column_name
      { new_data_type [ ( precision [ , scale ] ) ]
          [ NULL | NOT NULL ]
        | {ADD | DROP} ROWGUIDCOL
      }
  ]
  | ADD
      { [ <column_definition> ]
        | column_name AS computed_column_expression
      }[ ,...n ]
  | [WITH {CHECK | NOCHECK}] ADD
      { <table_constraint> }[ ,...n ]
  | DROP
      { [ CONSTRAINT ] constraint_name
        | COLUMN column
      }[ ,...n ]
  | {CHECK | NOCHECK} CONSTRAINT
      {ALL | constraint_name[ ,...n ]}
  | {ENABLE | DISABLE} TRIGGER
      {ALL | trigger_name[ ,...n ]}
}
```

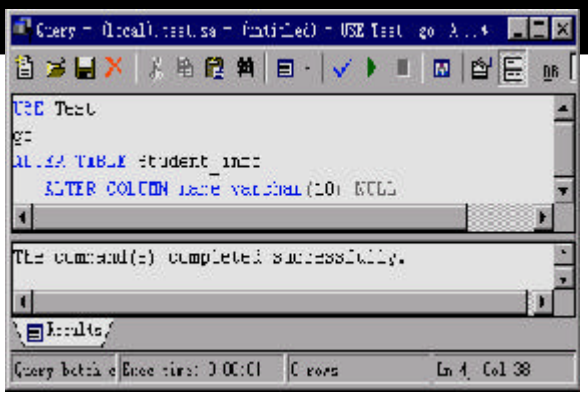
2 在 Query Analyzer 中执行 T-SQL 语句添加字段的结果如下图所示：



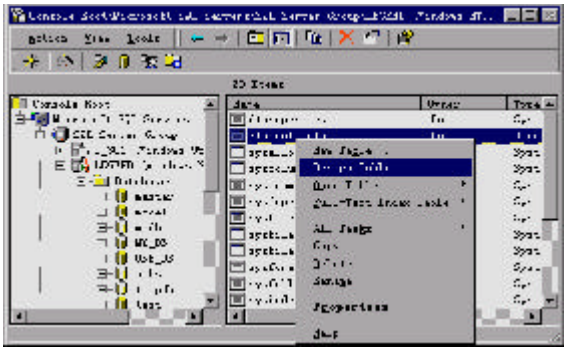
3 在 Query Analyzer 中执行 T-SQL 语句删除字段的结果如下图所示：



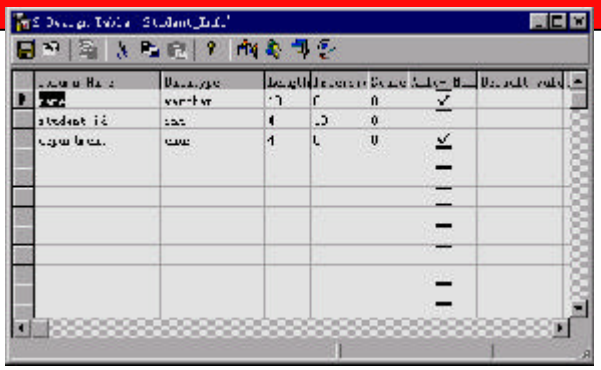
4



5 在 Enterprise manager 中也可以完成对已有表格字段的添加、删除和修改操作。在 Enterprise Manager 中展开所要操作的数据库中的表格项，右键单击要修改的表格，在弹出菜单中选择 Design Table 项。



6 在弹出的对话框中，可以用图形化工具完成添加、删除和修改表格字段的操作。如下图所示：



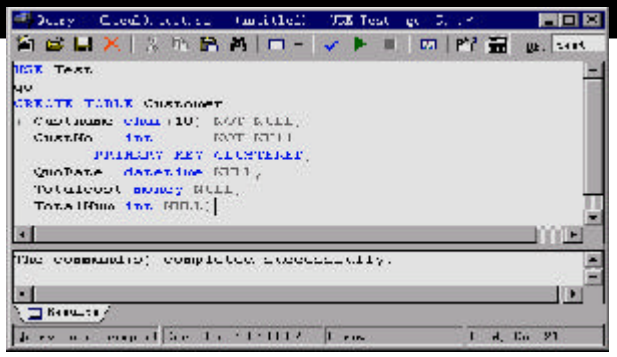
7 通过本例，学会了如何通过 T-SQL 语句或者通过 Enterprise Manager 完成添加、删除和修改已有表格的操作。

例

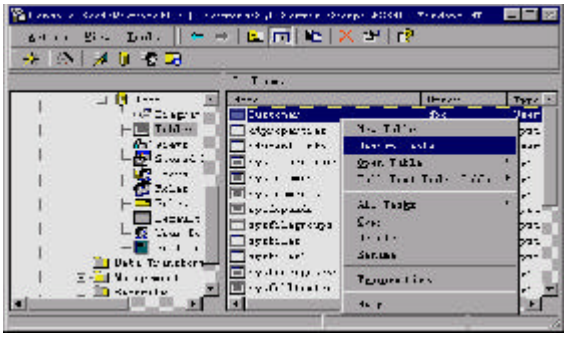
表格操作
添加、删除和修改主码

解

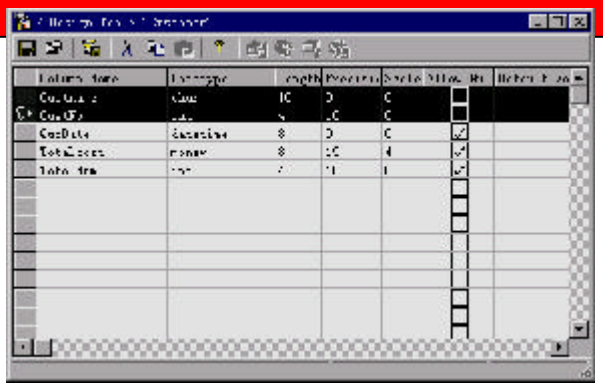
1



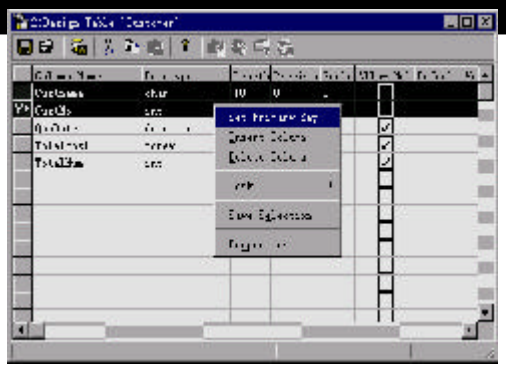
2 通过 Enterprise Manager 也可以完成添加、删除和修改主码的操作。如下图所示，在 Enterprise Manager 中右键单击要操作的表格，在弹出菜单中选择 Design Table。



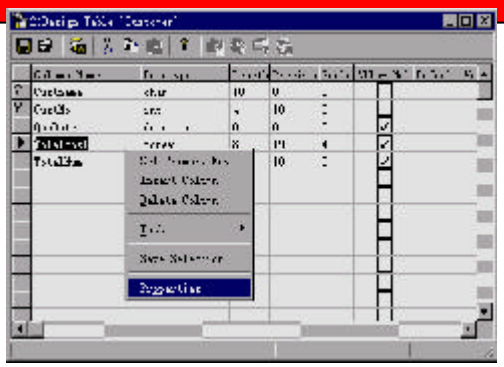
3 在表格的设计窗口中，选择要设定为主码的字段。如果有多个字段，按住 Ctrl 键后，用鼠标单击每个要选的字段。如下图所示。



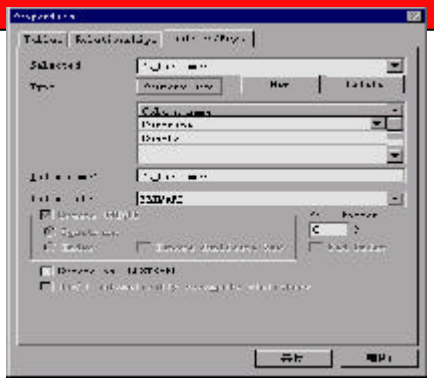
4



5 在 Enerprise Manager 中还可以删除已经设定的主码，是能够修改设定其他的字段为主码。右键单击某个字段，在弹出的菜单中选择 Properties。



6 在弹出的对话框中选择 Indexes/keys 页面，如下图。在 Selected 中选择 PK_Customer 表示主码的名称。单击 Delete 就可以删除了。



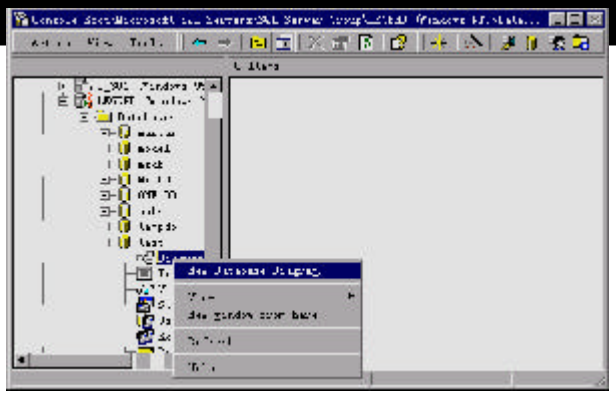
7 通过本例，学会了如何为表格创建主码，以及如何删除和修改表格的主码。

例

数据库视图
创建数据库视图

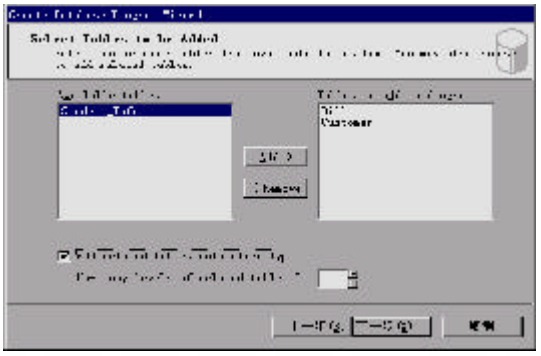
解

1

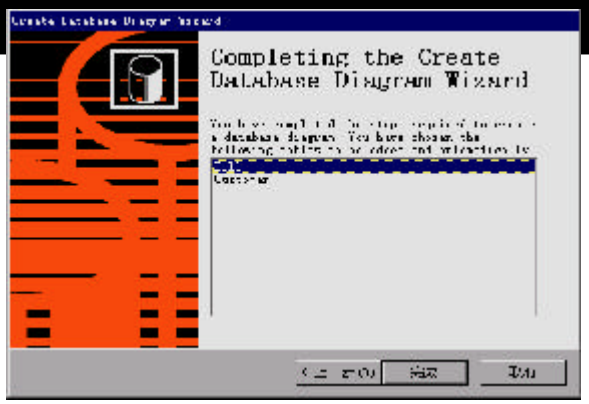


2 接下来弹出创建数据库视图向导的对话框如下图所示，它是对该向导功能的简述。选择下一步。

3 接下来选择要往数据库视图中加入的用户表格，在左边的 Available tables 中选择要加入的表格后单击 Add 键加入。下面的 Add related tables automatically 可以将与该表格相关的表格一并加入。

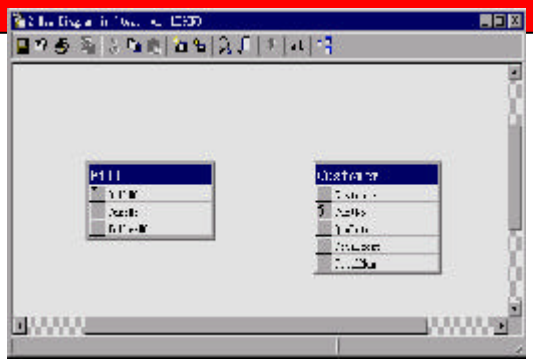


4



5 接着弹出对话框，表示表格已经被加入到视图中了。

6 接着在 Enterprise Manager 中可以看到打开的新建视图，如下图所示，其中两个子窗口是加入的表格。



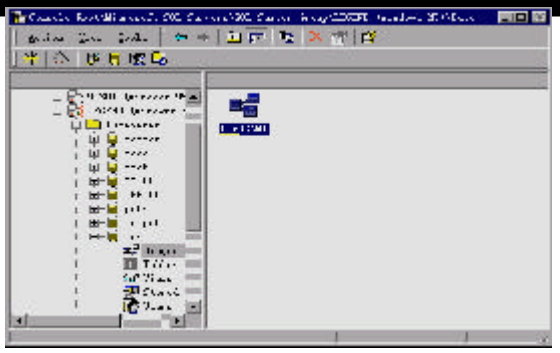
7 通过本例，学会了如何在 Enterprise manager 中创建数据库视图。后面将会看到数据库视图是非常有用的工具。

例

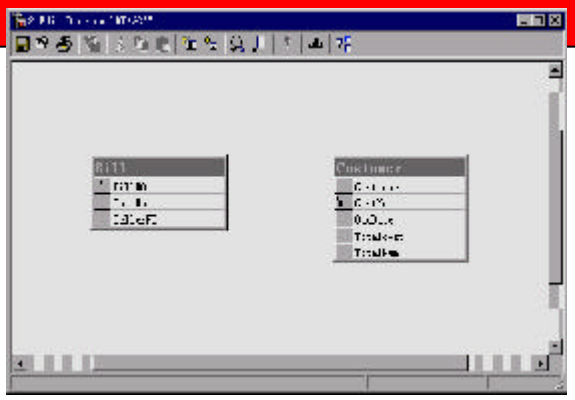
表格操作
添加外码限制

解

1

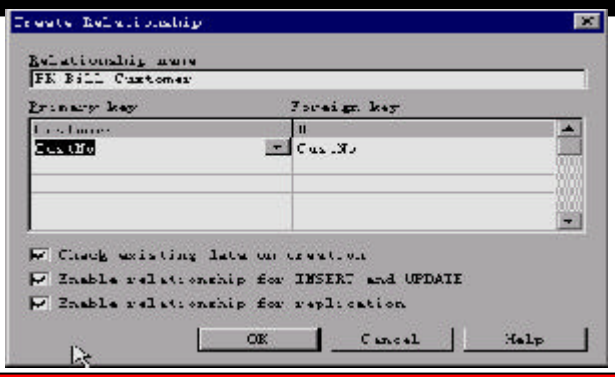


2 双击数据库视图图标后，打开的视图编辑窗口如下图所示：

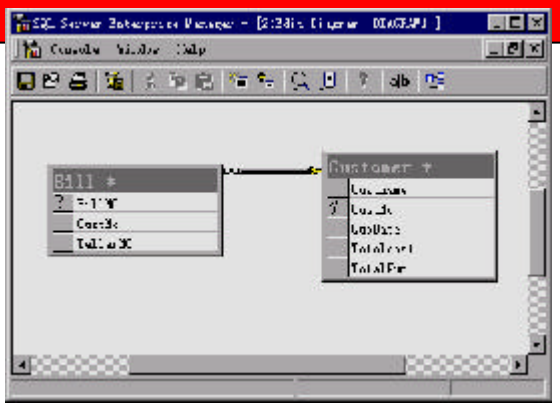


3 在上图所示的需要连接的两个表格中分别建立好主码和外码字段。单击主码字段，并拖动至需要建立连接的外码字段如下图所示：

4



5 单击 OK 键确认后，可以看到在两个表格之间建立一个连接，用箭头表示。这是该修改立刻影响数据库的内容，只是在视图中发生了改变。



6 当确定要修改数据库或者表格属性时，单击保存按钮 ，将会弹出对话框如下图。请确认是否要修改表格的外码属性，按下 Yes 按钮即可。

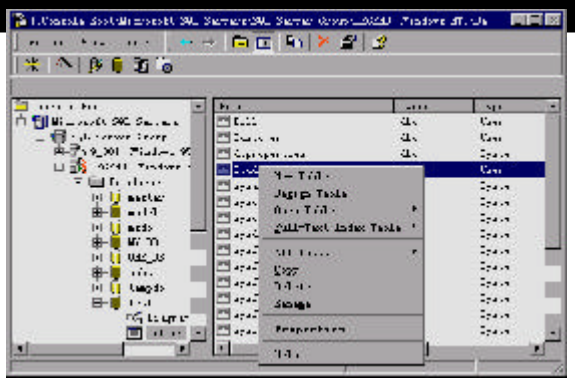
7 通过本例，学会了如何在 Enterprise Manager 中使用数据库视图在表格中建立连接，创建外码。该操作也可以通过 T-SQL 语句 CREATE TABLE 或者 ALTER TABLE 实现。

例

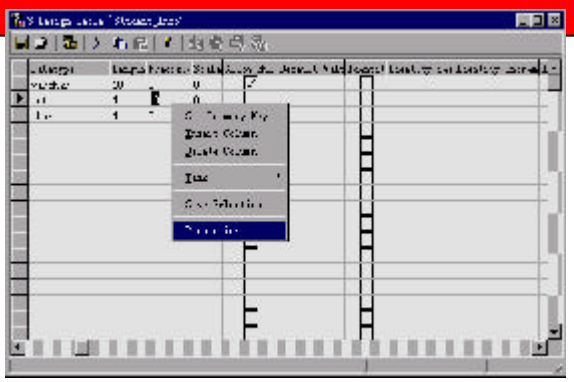
表格操作
创建和修改 UNIQUE 限制

解

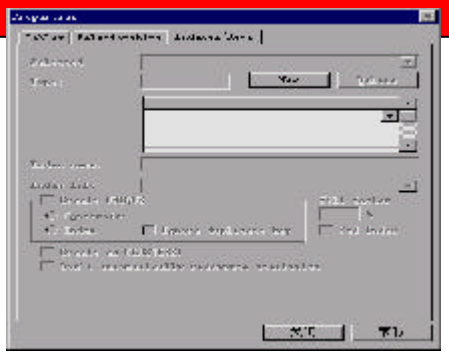
1



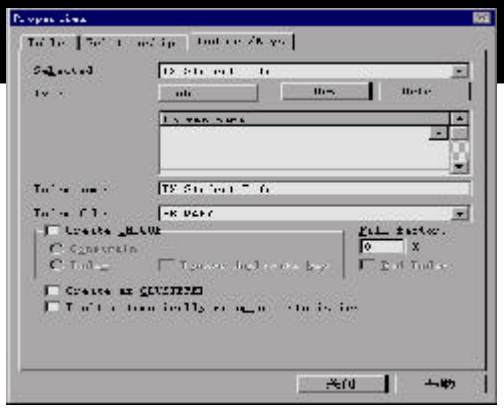
2 在表格的编辑窗口中右键单击某个子段定义，在弹出菜单中选择 Properties 项。



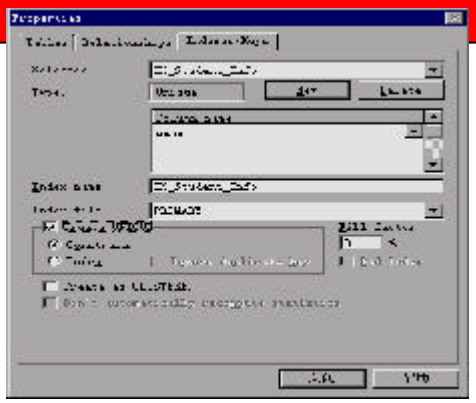
3 在弹出的对话框中选择 Indexes/Keys 页面， 如下图所示：



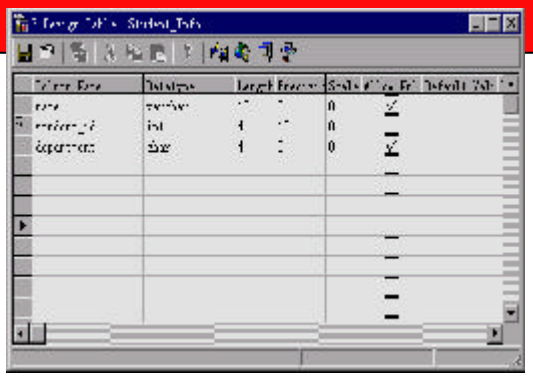
4



5 在 Column name 中选择字段名称，选择复选框 Create UNIQUE 为 Constrain 选项。然后按关闭键。



6 最后，可以在表格编辑窗口中选择保存，以在数据库上的上述修改生效。



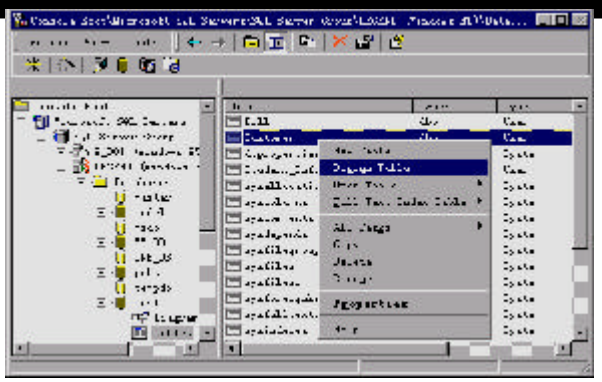
7 通过本例，学会了如何创建或者修改数据库表格的字段的限制。

例

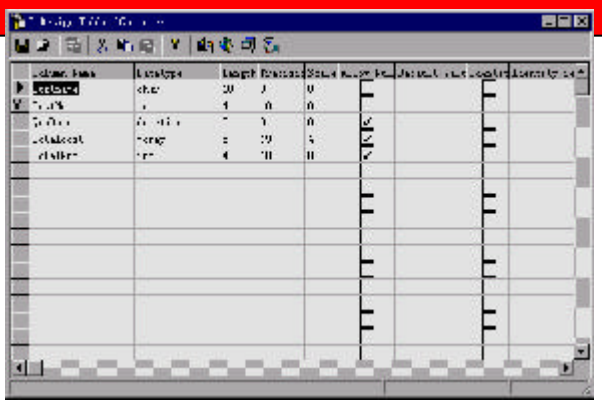
表格操作
创建 CHECK 限制

解

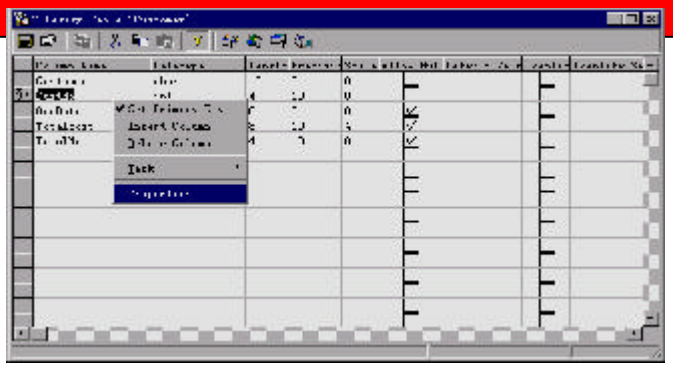
1



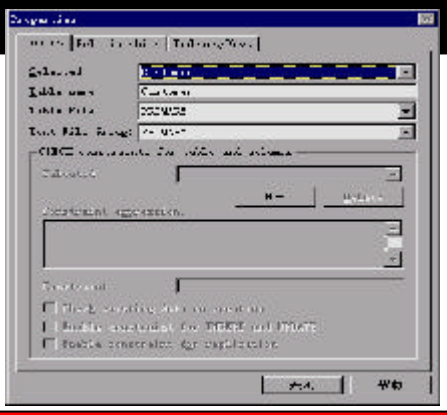
2 单击 Design table 后出现表格编辑窗口如下图所示：



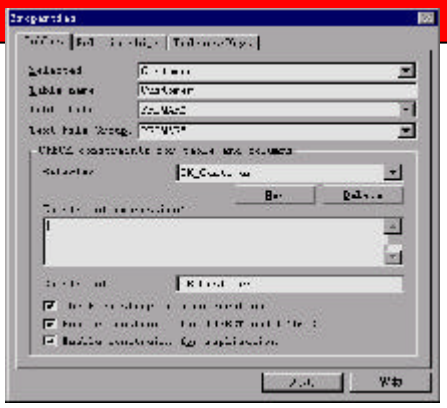
3 鼠标右键单击要设置 CHECK 限制的字段，在弹出菜单中选择 Properties，如下图所示：



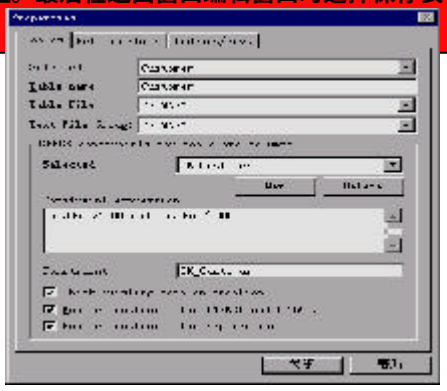
4



5 接着可以在 Constrain expression 栏中输入 CHECK 限制的表达式。如下图所示：



6 输入完 CHECK 限制对话框后，在 Selected 栏中输入该限制的名称。然后单击关闭按钮。最后在退出窗口编辑窗口时选择保存表格的修改。



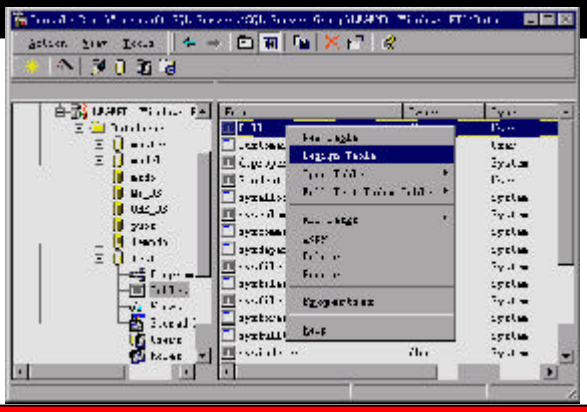
7 通过本例，学会了如何在 Enterprise Manager 中创建 CHECK 限制，以及如何设置 CHECK 表达式。

例

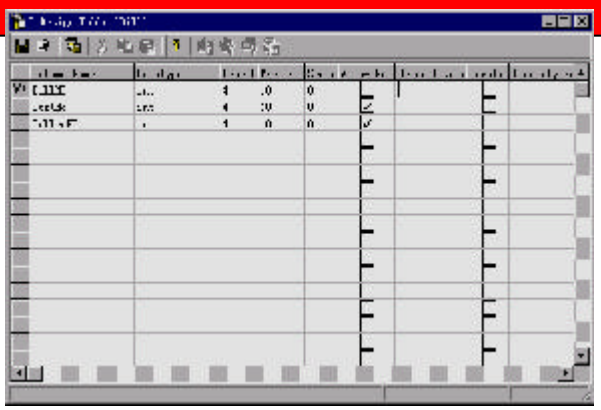
表格操作
设置字段缺省值和标识符字段

解

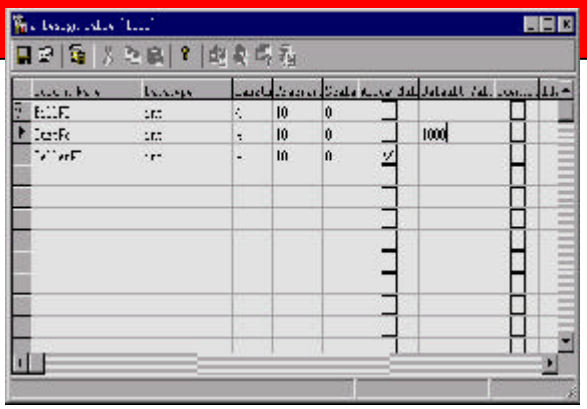
1



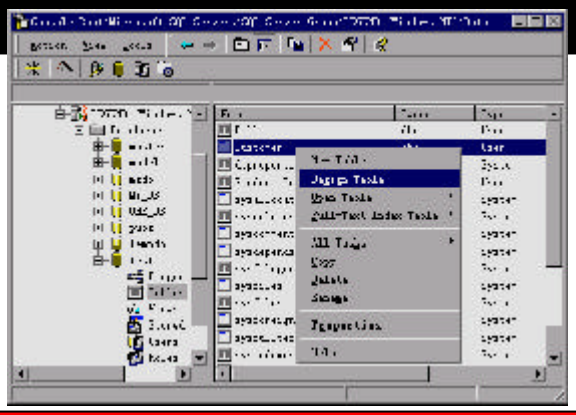
2 接着在 Enterprise Manager 中将会出现表格编辑窗口，选定要设置的字段后，找到 Default value 列，如下图光标所示：



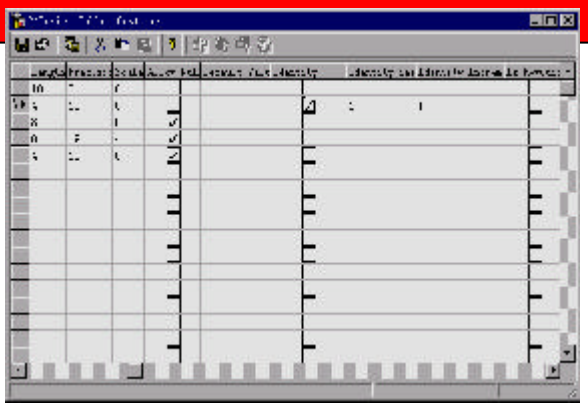
3 在 Default value 中输入该字段的缺省值如下图所示：



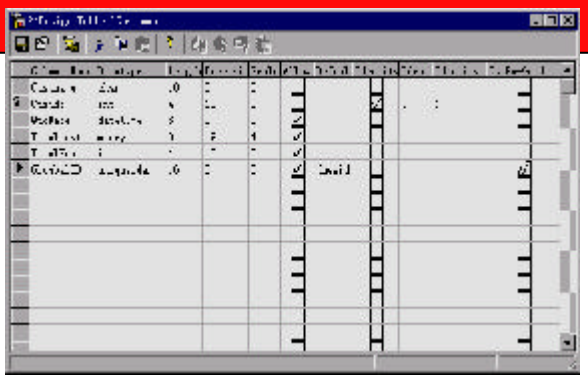
4



5 在要设定为标识符字段的字段行中找到 Identity 列，选定 Identity 复选框。还可以在 Identity Seed 中设定起始值，在 Identity Increment 中设定增量。



6 如图要创建全球唯一的标识符字段 gloabal unique identifier，先要设定该数据类型为 uniqueidentifier 类型，然后选中该字段 IS ROWGUID 复选框。



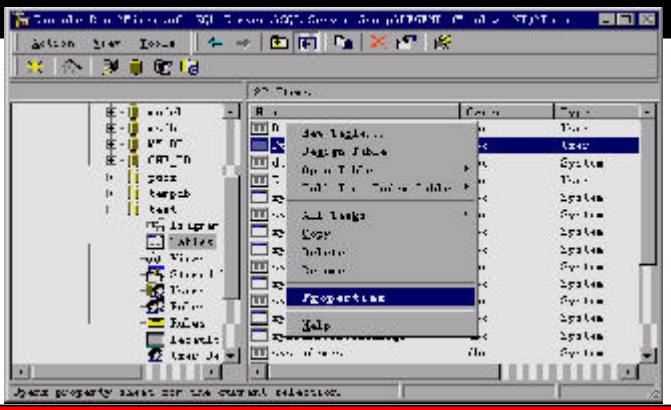
7 在本例中，学会如何在 Enterprise Manager 中创建和设置标识符字段和全球唯一的标识符字段。

例

表格操作
查看表格属性

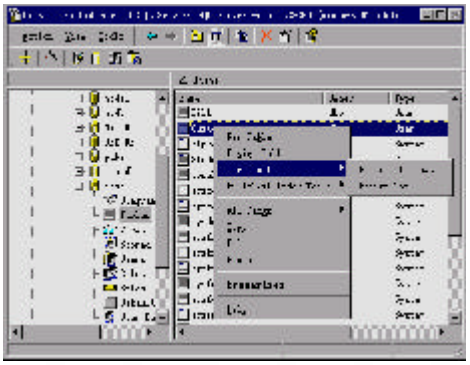
解

1

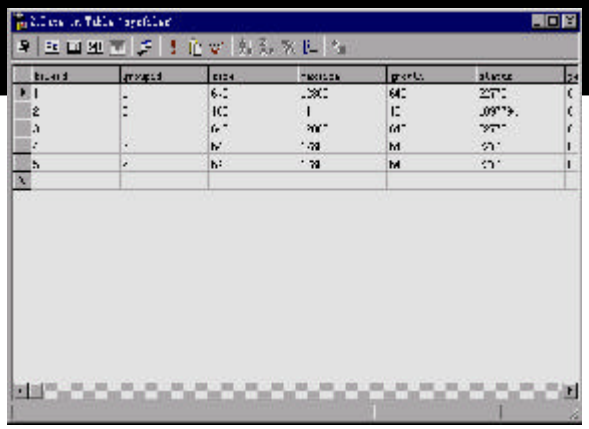


2 弹出对话框中，会显示该表格的定义，如各个字段的名称、类型、长度等属性。

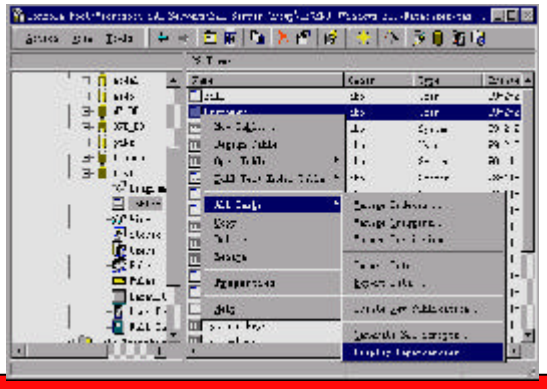
3 查看表格内存放记录的数据：在 Enterprise Manager 中，展开数据库，展开 Tables 项， 在右栏窗口中右键单击要操作的表格，在弹出菜单中选中 Open Table.



4



5 查看表格与其它数据库对象的相互依赖关系：在 Enterprise Manager 中，右键单击要查看的表格，在弹出菜单中选择 All tasks ー>Display dependency。



6 接下来的窗口中会显示依赖于此表格的对象（左栏），以及此表格依赖的对象（右栏）。

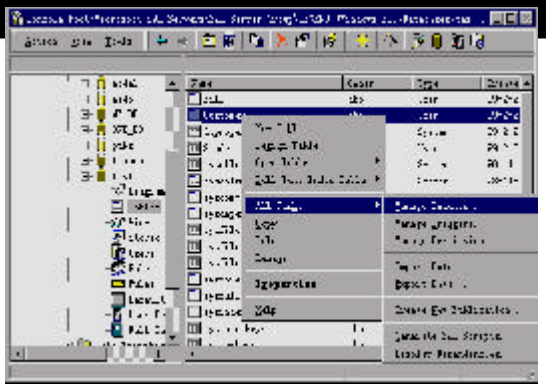
7 通过本例，学会了如何在 Enterprises manager 中查看表格的属性、定义，以及表格和其他数据库对象之间的关系。

例

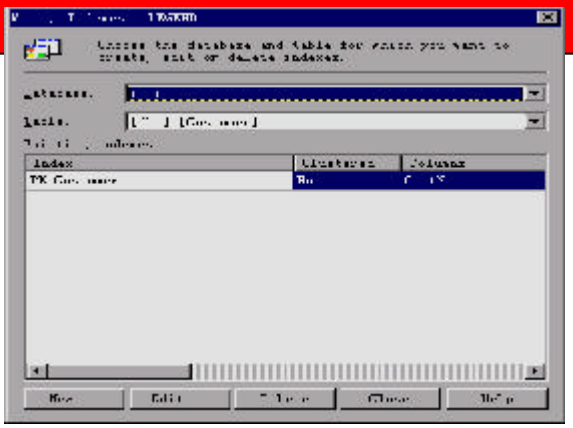
索引操作
创建索引

解

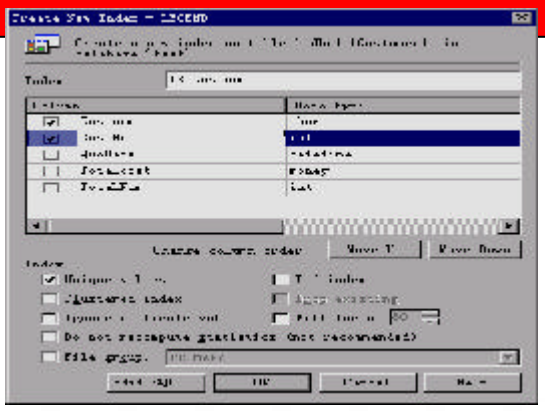
1



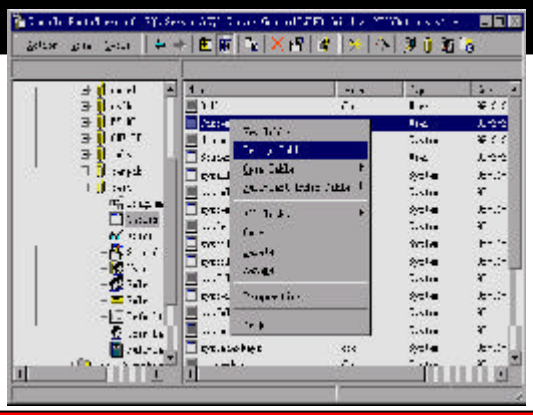
2 弹出管理索引窗口如下图所示，单击 New 按钮。



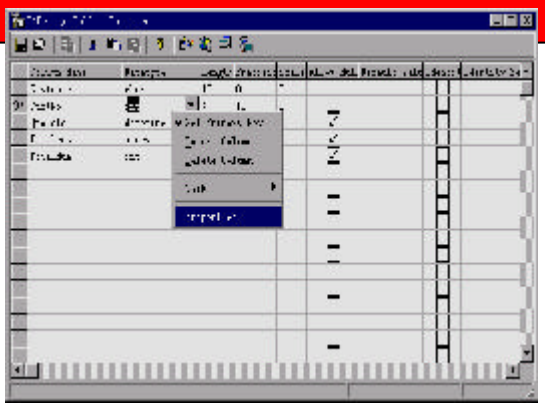
3 在弹出的创建索引窗口中设置索引的名称，选择建立索引的字段，设置索引的属性。如下图。设置完按 OK 按钮即可。



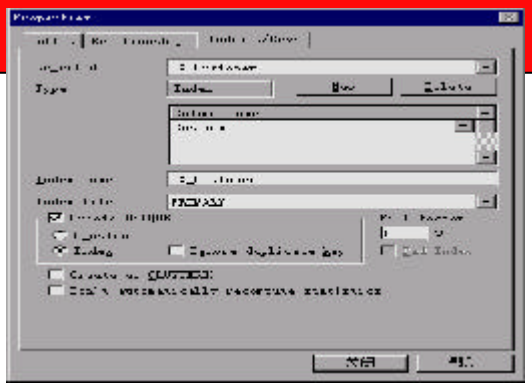
4



5 右键单击表格编辑窗口中的某个字段，在弹出的菜单中选择 Properties 项。



6 在弹出的属性对话框中选择 Index/keys 页面，按下 New 按钮。



7 通过本例， 学会了如何在 Enterprise Manager 中为表格创建索引。

例

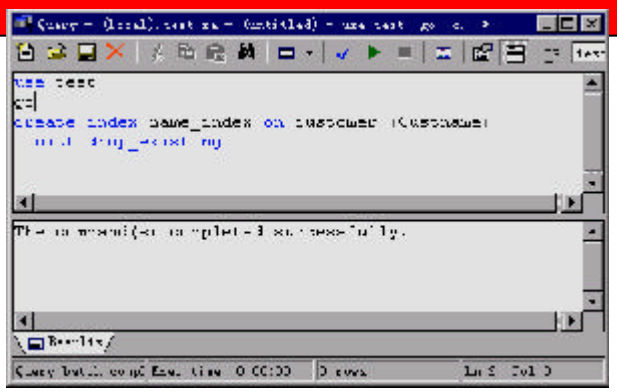
索引操作
重建索引

解

1

```
CREATE [UNIQUE] [CLUSTERED | NONCLUSTERED]
   INDEX index_name ON table (column [,...n])
[WITH
      [PAD_INDEX]
      [[,] FILLFACTOR = fillfactor]
      [[,] IGNORE_DUP_KEY]
      [[,] DROP_EXISTING]
      [[,] STATISTICS_NORECOMPUTE]
]
[ON filegroup]
```

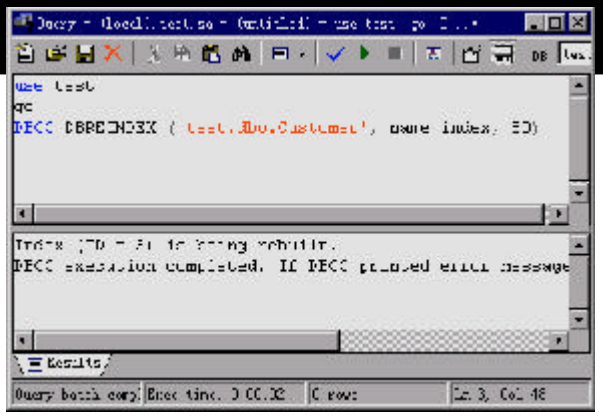
2 **在 Query Analyzer 中执行重建索引的操作如下图所示，其中索引名称必须是已经存在的索引名，drop_existing 分句用来指明重建索引。**



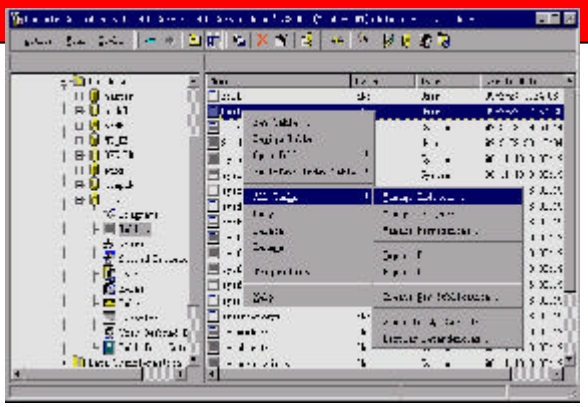
3 **一次重建多个索引的 T-SQL 语句如下图所示：**

```
DBCC DBREINDEX
   (['database.owner.table_name' [,index_name [,fillfactor ]]]
   ) [WITH NO_INFOMSGS]

注：'database.owner.table_name '：中是表格的名称，包括数据库名、表格
                    所有者、表格名称。其中前两项可以省略
              index_name   ：索引名称。
          fillfactor       ：装填因子。
          WITH NO_INFOMSGS ：结果不发任何消息。
```

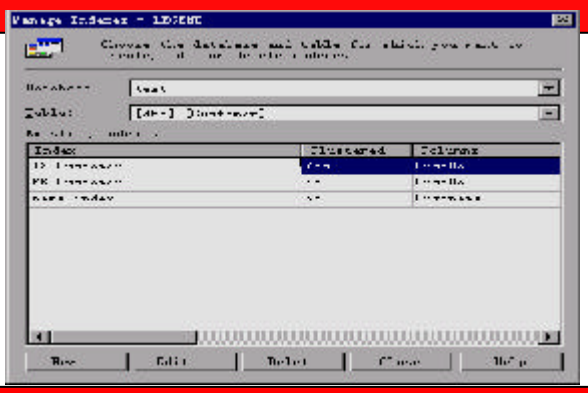
4



5 删除表格的方法：在 Enterprise Manager 中右键单击表格，在弹出菜单中选择 All tasks－》Manage Indexes。



6 在弹出的索引对话框中选择某个索引名称后，单击 Delete 按钮，再在弹出对话框中确认即可。



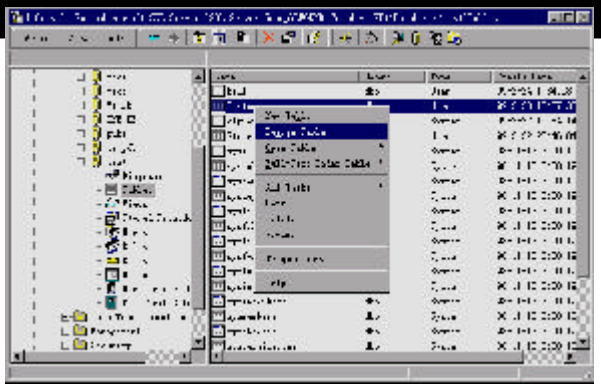
7 通过本例的学习，学会了如何重新创建索引，以重新组织数据库系统的存储。

例

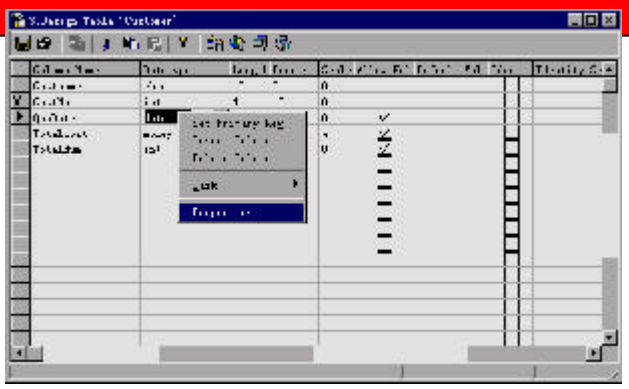
表格操作
重命名表格，查看表格信息

解

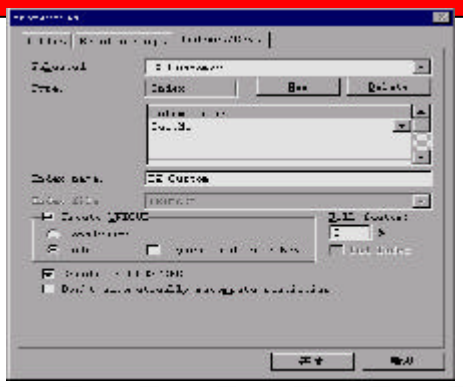
1



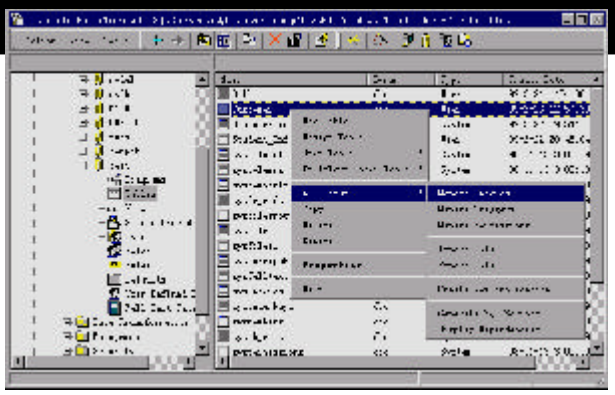
2 在表格的属性窗口中，右键单击某个字段，在弹出菜单中选择 Properties。



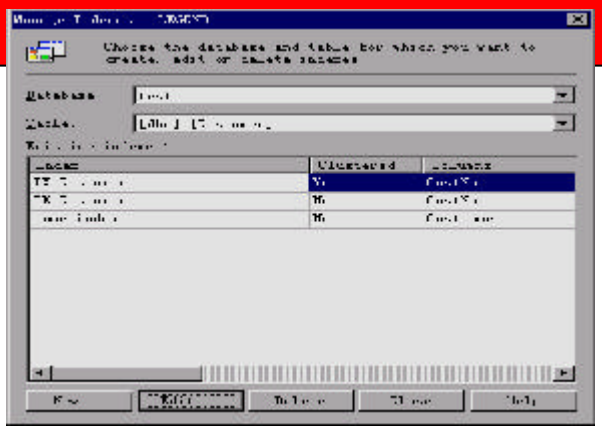
3 在弹出的对话框中，选择 Indexes/Keys 页面，在 Index name 中输入重命名的 Index 名称。



4



5 在弹出的对话框中，选择某个索引后，单击 Edit 按钮。



6 在弹出的索引属性对话框中，可以直接修改索引的属性。

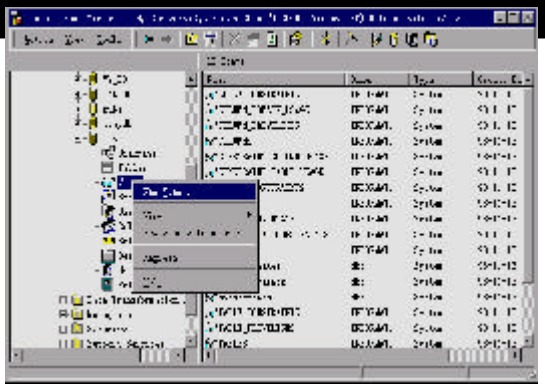
7 通过本例，学会了如何重命名索引，以及如何查看索引信息。

例

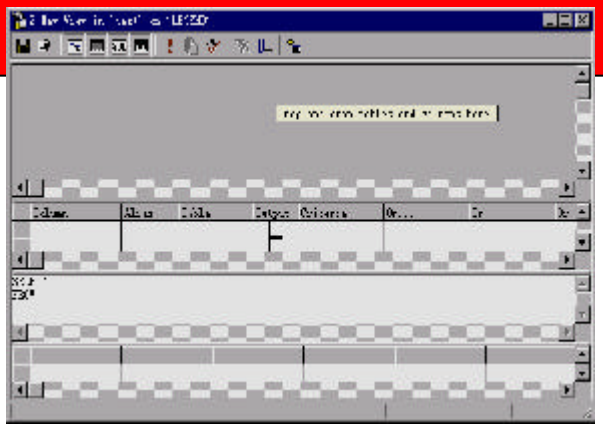
视图操作
创建视图

解

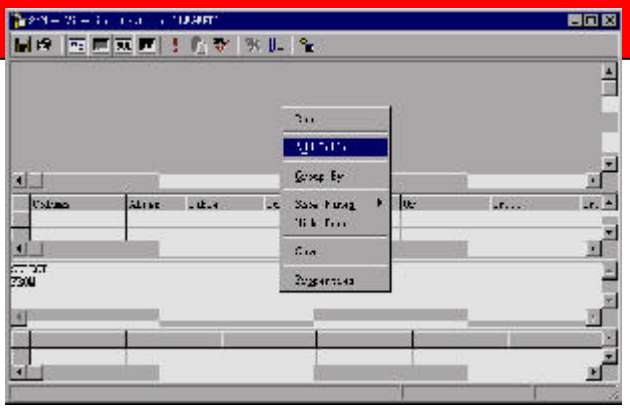
1



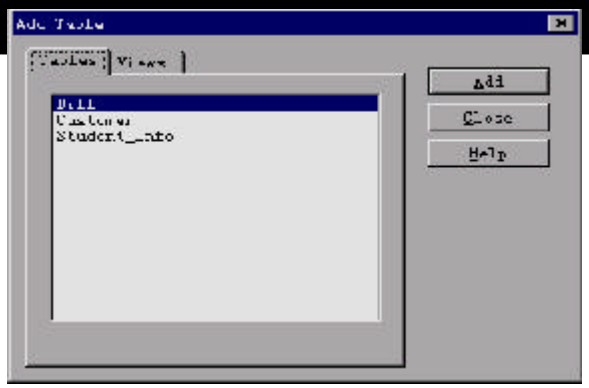
2 弹出的如下窗口可以定义视图。



3 右键单击图表窗格，在弹出菜单中选择 Add Tables。

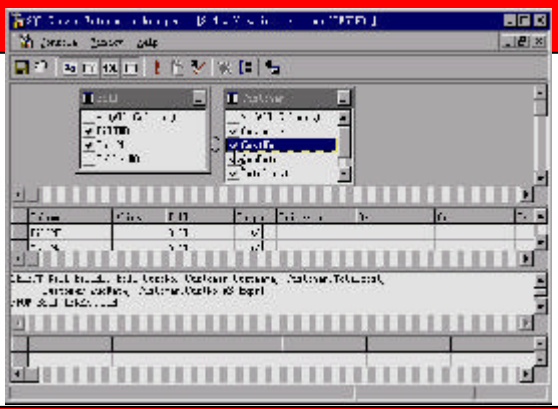


4



5 添加完表格和视图后可以在图表窗格中看到新添加的表格和视图。

6 在表格前字段复选框中单击，可以选中在视图中被引用的字段。



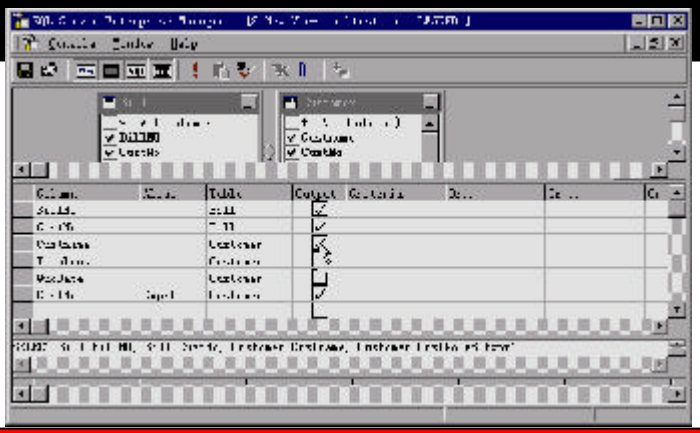
7 本例未完，下节续。

例

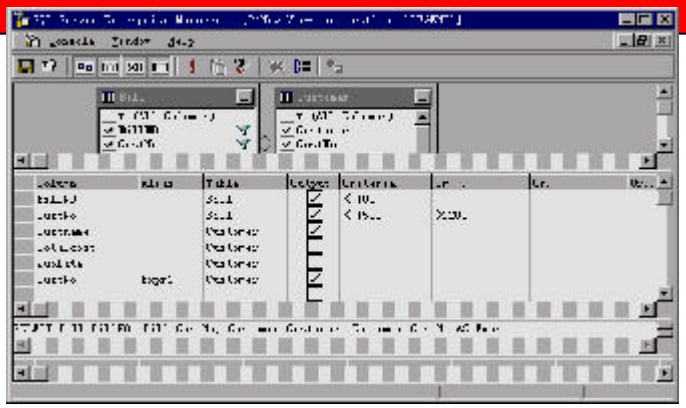
视图操作
创建视图（续）

解

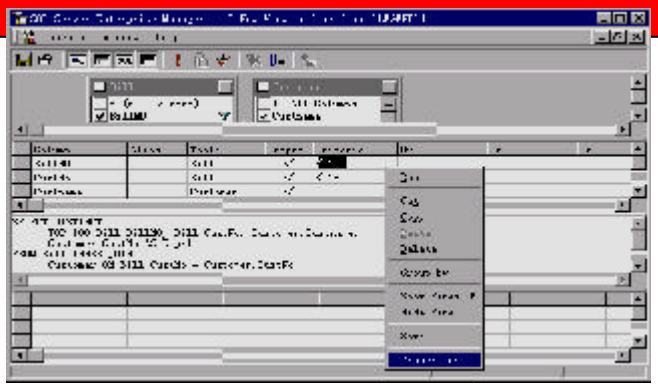
1



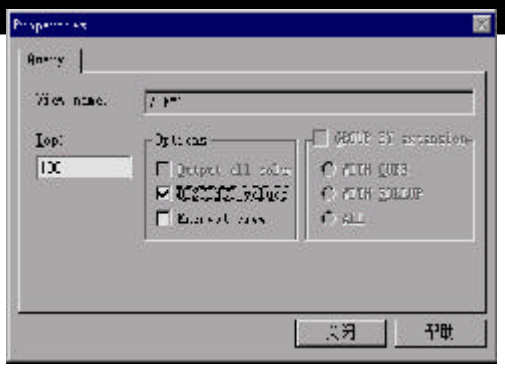
2 在 Criteria 中输入限制条件，这可以限制输出的记录满足该条件。在定义视图的查询语句中该限制条件对应 WHERE 子句。Or..可以用于添加或条件。



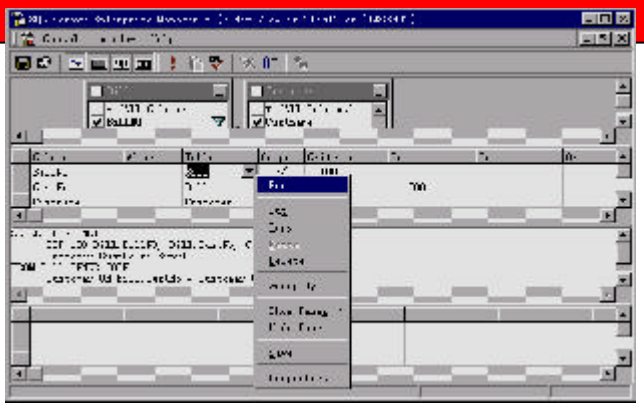
3 右键单击字段定义窗格，在弹出菜单中选择 Properties 项。



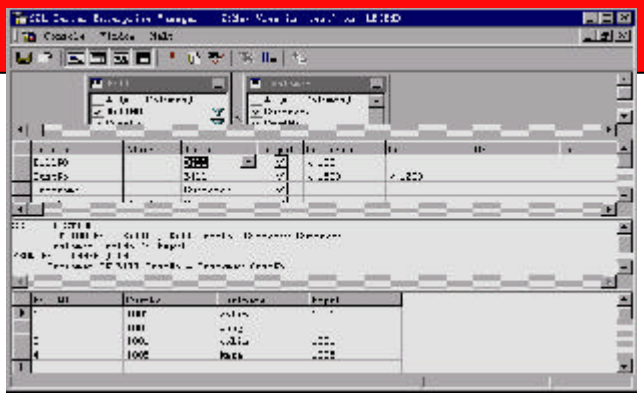
4



5 要运行并输出该视图结果，可以右键单击字段定义窗格，在弹出菜单中选择 Run。也可以单击 ! 按钮。



6 在最下面的输出窗格中将会显示按照中间的查询语句生成的视图内容。



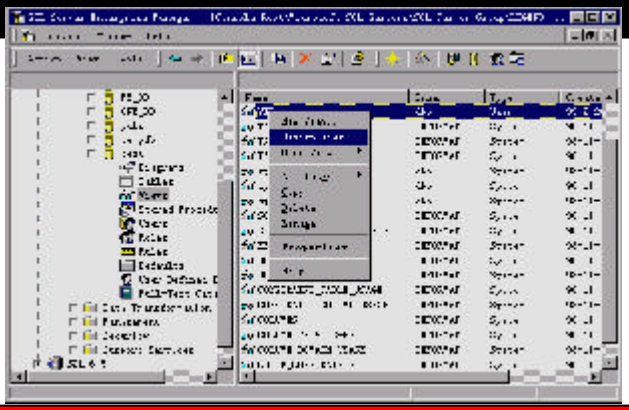
7 通过本例的学习，学会了如何创建视图，并且定义视图的一些属性。

例

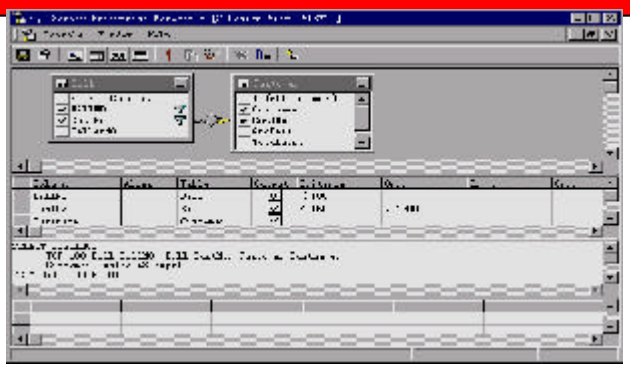
视图操作
修改和重命名视图

解

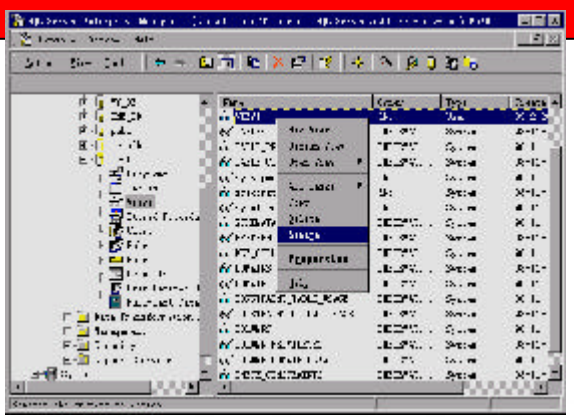
1



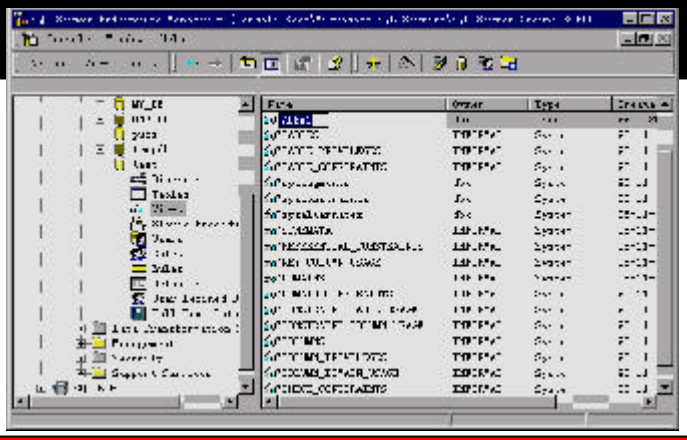
2 接着弹出的视图属性窗口与定义视图时的窗口相同，可以按照定义视图的方法修改视图的属性。



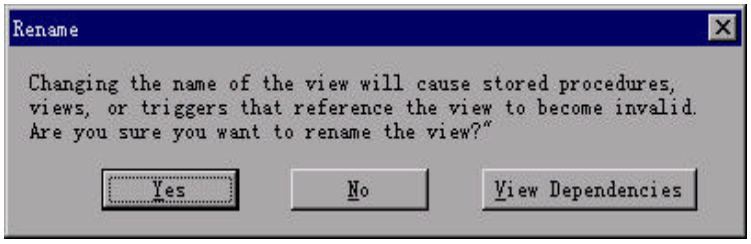
3 修改视图的名称：在 Enterprise Manager 中右键单击要重命名的视图，在弹出菜单中选择 Rename。



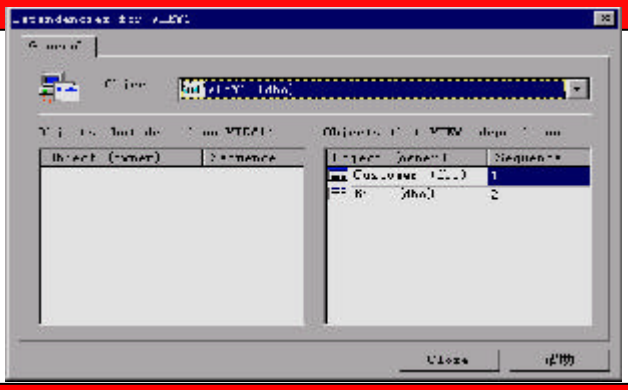
4



5 名称修改完后，将会弹出如下对话框，选择 Yes 就可以了，另外，单击 View Dependencies 可以查看与该视图有关的其他数据库对象。



6 单击 View Dependencies 按钮，可以查看与该视图有关的其他数据库对象。如下图：



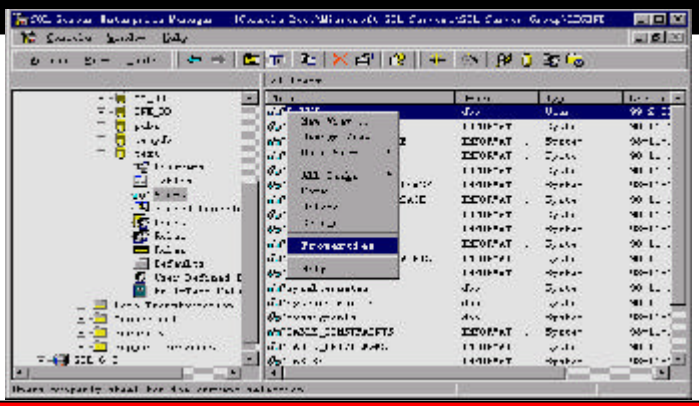
7 通过本例的学习，学会了如何在 Enterprise Manager 中修改视图的属性和名称。

例

视图操作
查看视图信息、删除视图

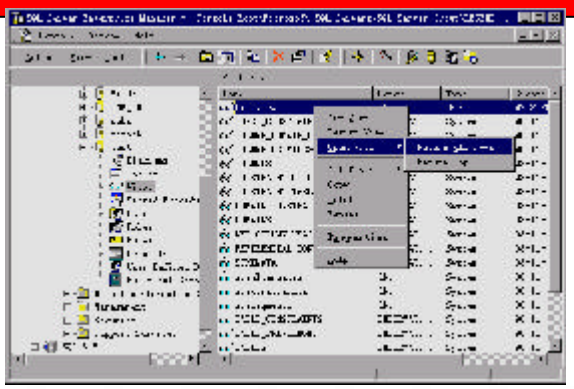
解

1

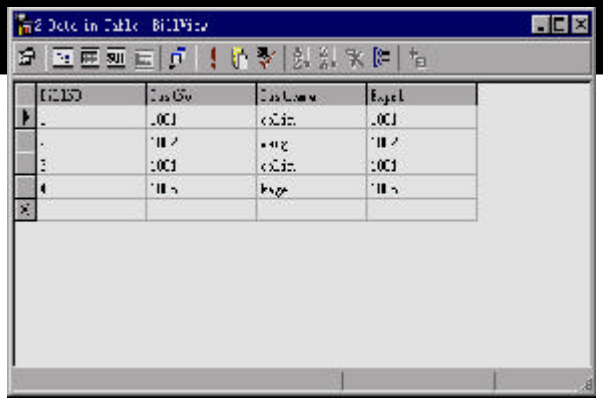


2 弹出的属性对话框显示了视图的定义。

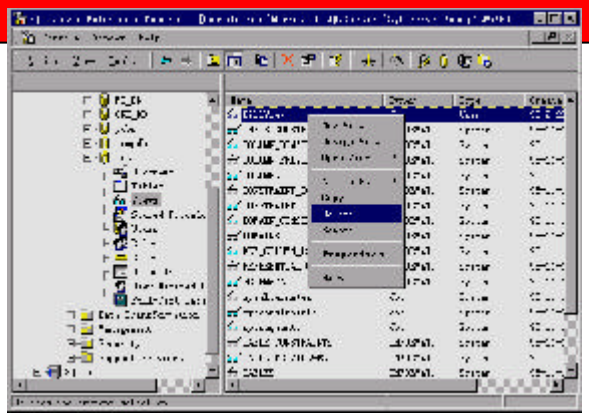
3 查看视图的输出数据：在 Enterprise manager 中，右键单击某个视图的名称，在弹出菜单中选择 Open View/Return all data （或者 Return Top）。



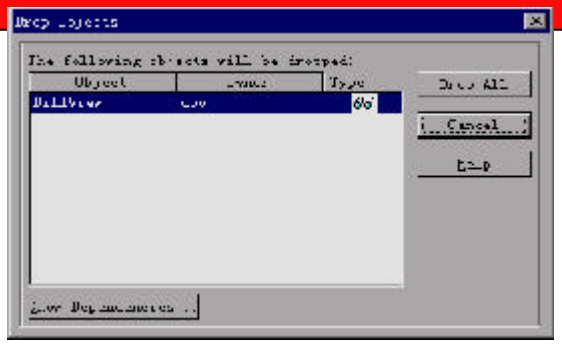
4



5 在 Enterprise Manager 中右键单击要被删除的某个视图名称，在弹出菜单中选择 Delete 命令。



6 在弹出的对话框中选择 Drop All 按钮完成删除操作。单击 Show Dependences 可以显示与该视图相关联的数据库对象。



7 通过本例的学习，学会了如何查看视图的定义、显示视图的输出数据以及如何删除视图。

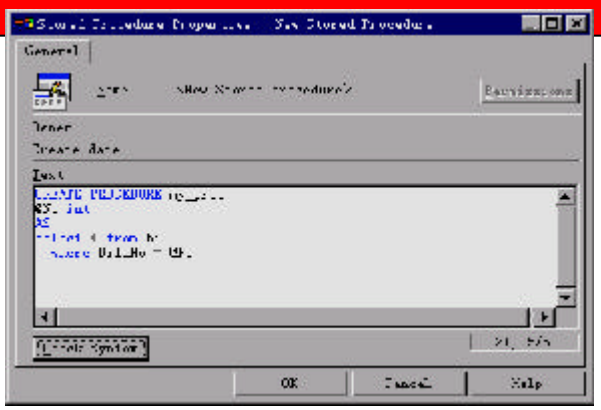
例

存储过程
创建存储过程

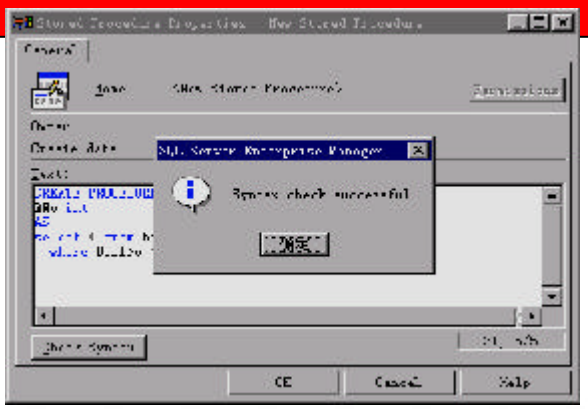
解

1

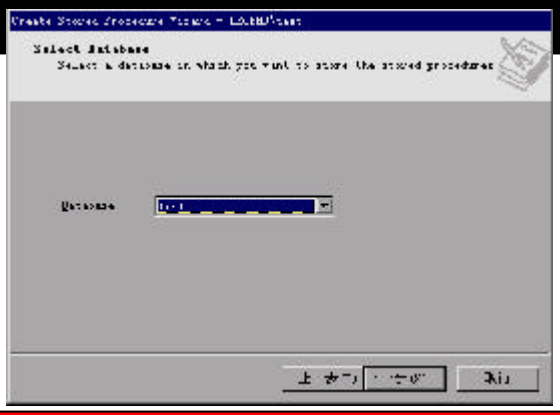
2 在 Text 框中，输入创建存储过程的 T-SQL 语句，如下图所示。单击 OK 即可完成创建操作。



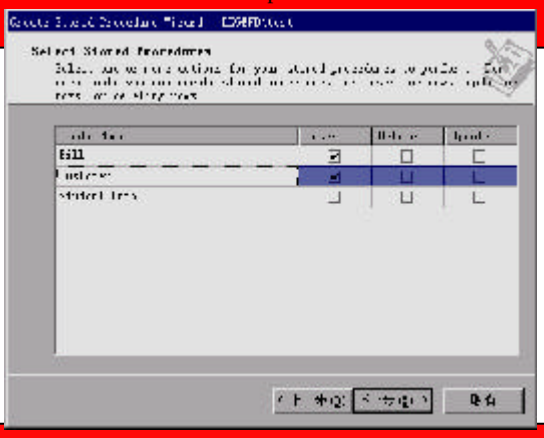
3 单击 Check Systax 按钮可以检查该语句的语法。如果语法正确，将会弹出如下对话框。



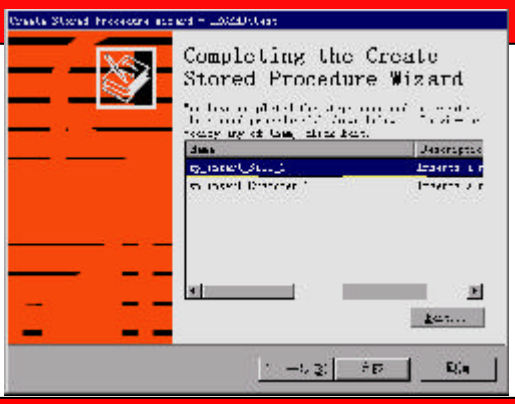
4



5 接着在对话框中选择要操作的表格，以及对表格进行何种操作。有插入（Insert）、删除（Delete）、更新（Update）三种操作。



6 最后的完成窗口中会显示对创建的存储过程的描述，可以单击 Edit 按钮编辑创建存储过程的 T-SQL 语句。



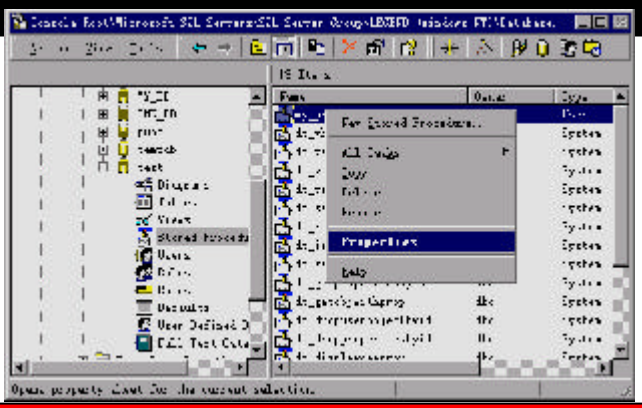
7 通过本例，学会了如何在 Enterprise Manager 中以及如何通过 Create Stored Procedure Wizard 创建存储过程。

例

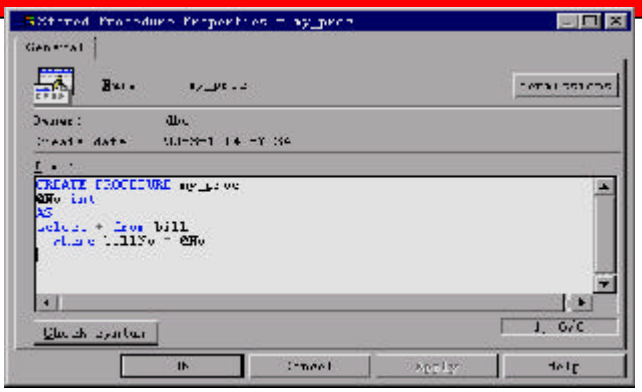
存储过程操作
修改或者重命名存储过程

解

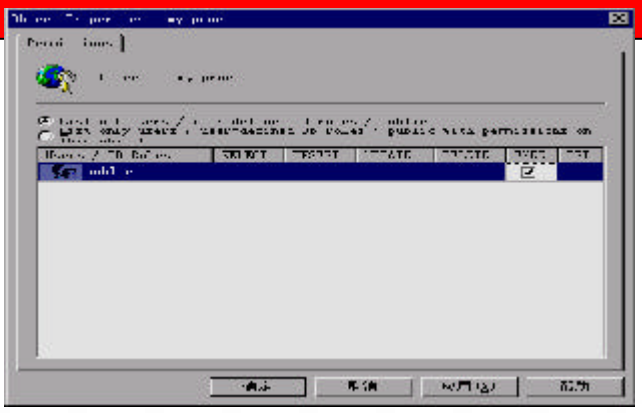
1



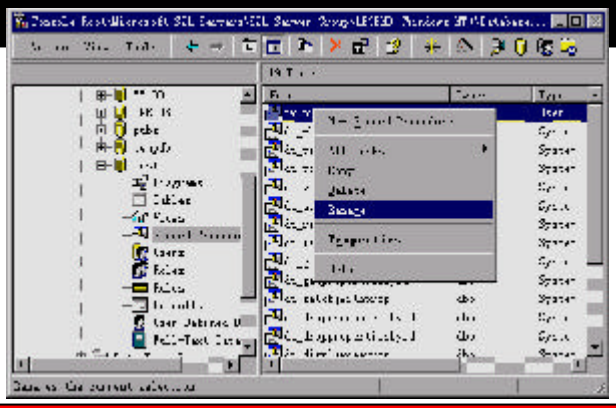
2 在弹出的存储过程属性窗口中，可以直接修改定义该存储过程的 T-SQL 语句。单击 Check Syntax 可以检查语法，单击 Permissions 可以修改用户执行该存储过程的权限。



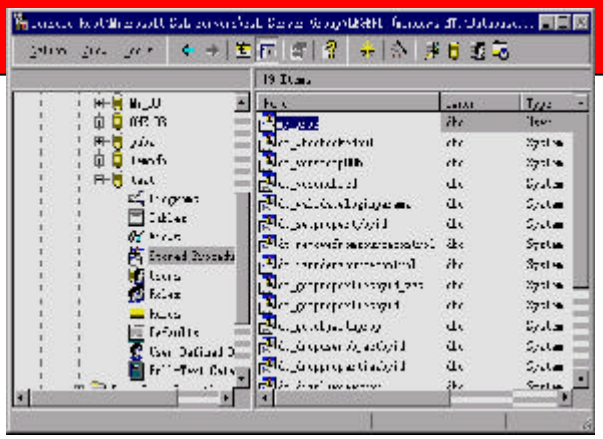
3 按下 Permissions 后，可以设定访问该数据库的用户对该存储过程是否具有执行的权限。



4



5 当存储过程名称变为可输入状态时，直接修改该存储过程的名称。



6 在弹出的确认对话框中选择“是(Y)”即可。

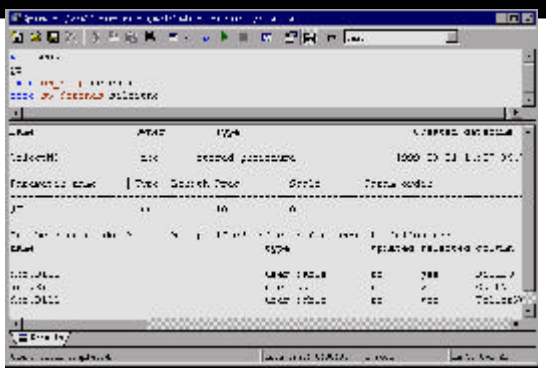
7 通过本例，学会了如何修改存储过程的属性和权限，以及如何重命名存储过程。

例

存储过程操作
查看和删除存储过程

解

1



2 在 Query Analyzer 中执行查询语句的结果如下图所示：

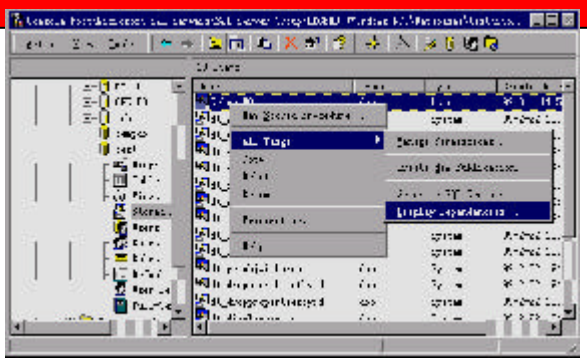
```
sp_help [[@objname =] name]
```

注：name为要查看的存储过程的名称。

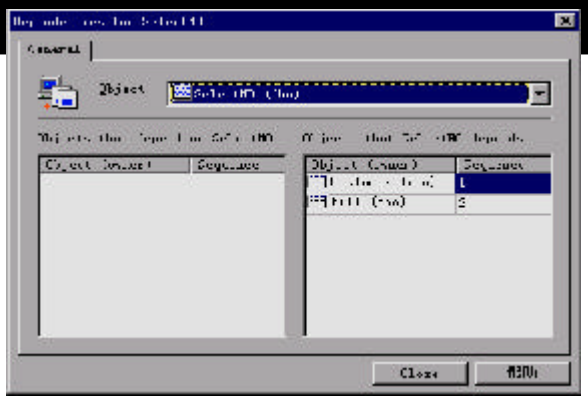
```
sp_depends [@objname =] 'object'
```

注：object为要查看依赖关系的存储过程的名称。

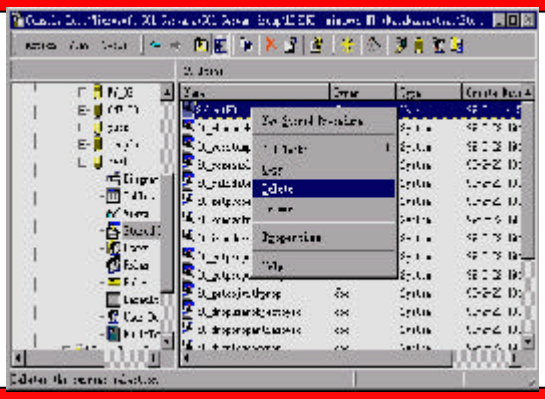
3 在 Enterprise Manager 中，右键单击要查看的存储过程的名称。在弹出菜单中选择 All tasks—》Display Dependences



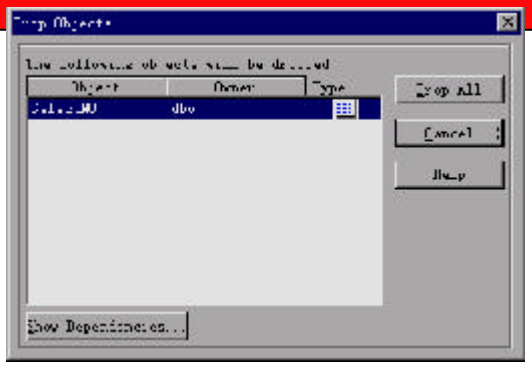
4



5 要删除对话框，在 Enterprise Manager 中，右键单击要删除的存储过程，在弹出菜单中选择 Delete 项。



6 在弹出对话框中，按下 Drop All 按钮，即可完成删除操作。按下 Show Dependences 可以在删除前查看与该存储过程有依赖关系的其他数据库对象名称。



7 通过本例，学会了如何查看存储过程的信息，存储过程与其他数据库对象的依赖关系以及如何删除存储过程。

例

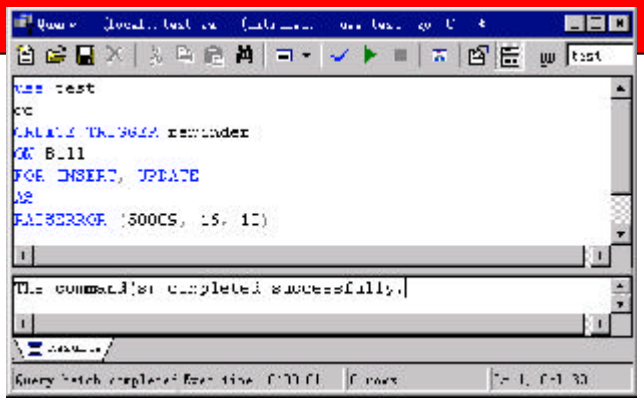
触发器操作
创建触发器

解

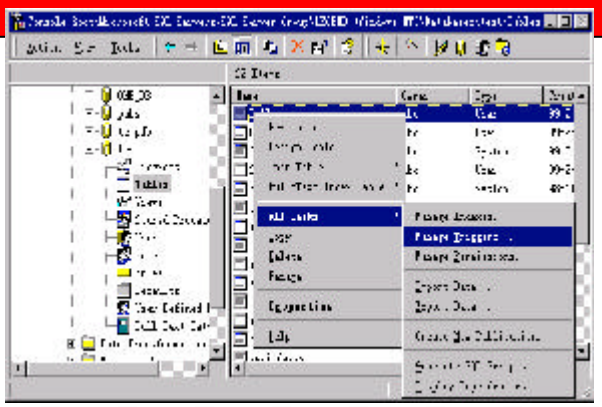
1

```
CREATE TRIGGER trigger_name
ON table
[WITH ENCRYPTION]
{
    { { FOR | AFTER | INSTEAD OF } { [DELETE] [,] [INSERT] [,] [UPDATE] }
        [WITH APPEND]
        [NOT FOR REPLICATION]
        AS
        sql_statement [ ...n]
    }
    |
    { FOR | AFTER | INSTEAD OF } { [INSERT] [,] [UPDATE] }
        [WITH APPEND]
        [NOT FOR REPLICATION]
        AS
        { IF UPDATE ( column )
            [ { AND | OR } UPDATE ( column ) ]
                [ ...n ]
        | IF ( COLUMNS_UPDATED ( ) { bitwise_operator } updated_bitmask )
            { comparison_operator } column_bitmask [ ...n ]
        }
        sql_statement [ ...n ]
    }
}
```

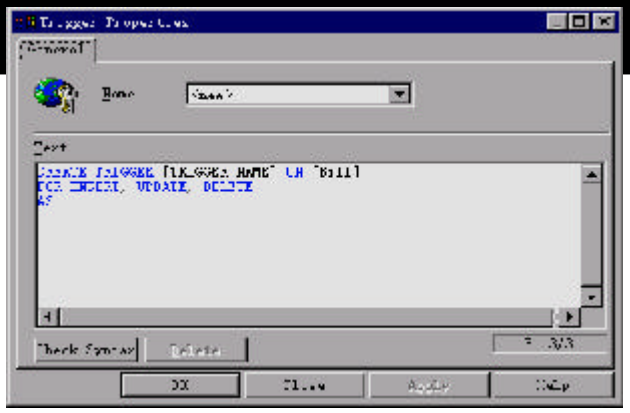
2 在 Query Analyzer 中执行创建触发器，其结果如下图所示：



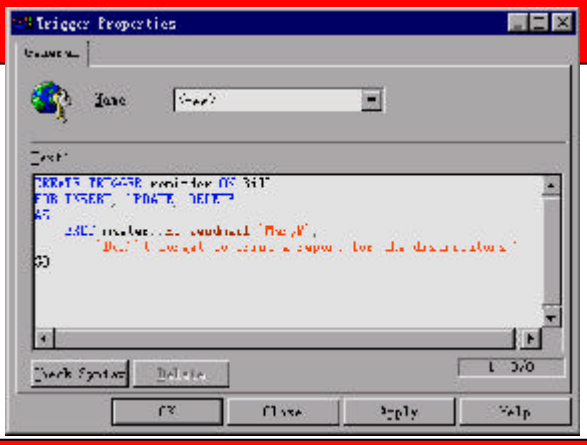
3 在 Enterprise manager 中右键单击要添加触发器的表格，在弹出菜单中选择 Manage Triggers。



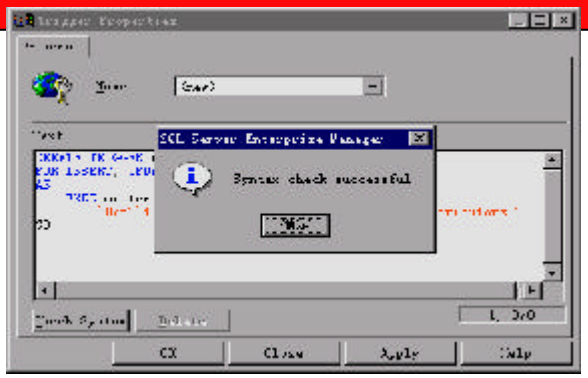
4



5 在 Text 栏中直接输入创建触发器的 T-SQL 语句，如下图所示：



6 单击 Check Syntax 可以检查 T-SQL 语句的语法正确性。如果语法正确将会弹出如下对话框。



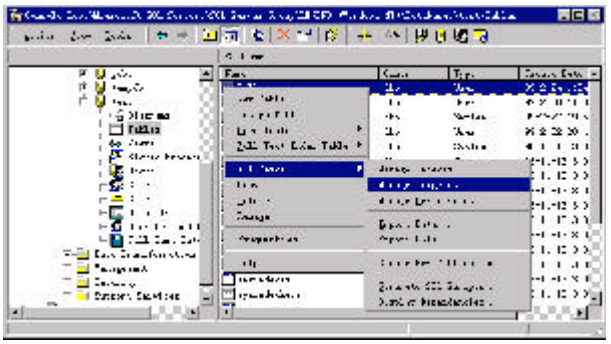
7 通过本例，学会了如何使用 T-SQL 语句或者在 Enterprise Manager 中创建触发器。

例

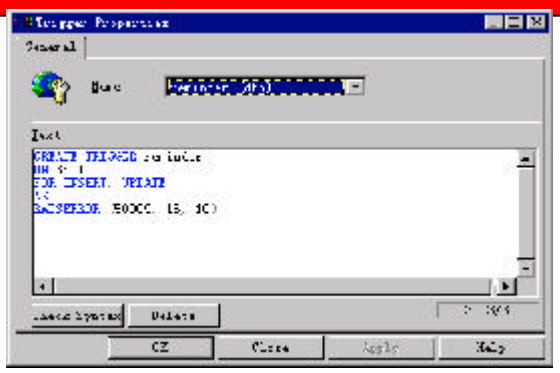
触发器操作
查看和删除触发器

解

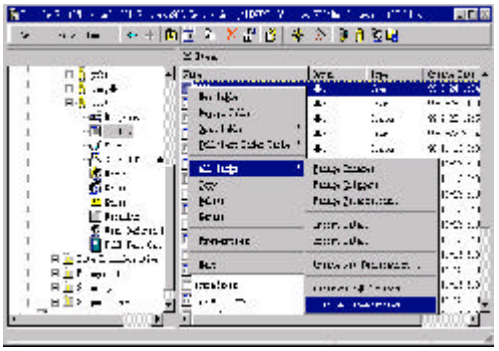
1



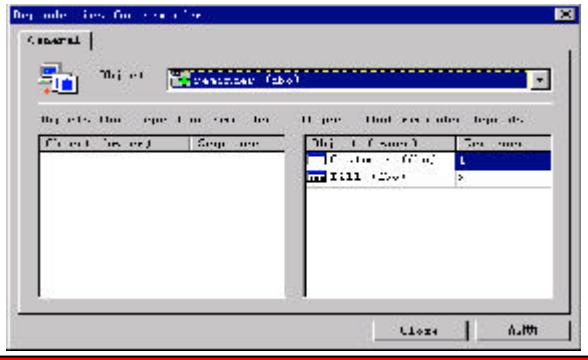
2 弹出触发器的定义对话框如下图，在 Name 选单中选择要查看的触发器名称，Text 中就会显示该触发器用 T-SQL 定义的语句。



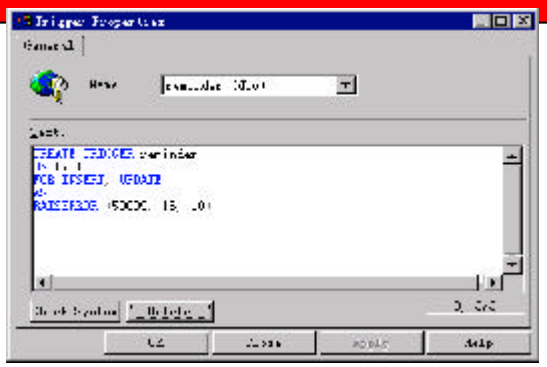
3 要查看与触发器有依赖关系的其他数据库对象：在 Enterprise Manager 中，右键单击要查看的表格，在弹出菜单中选择 All tasks －》 Display Dependences。



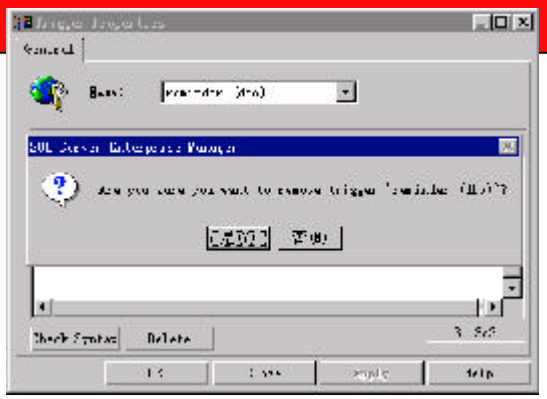
4



5 在图２所示的管理触发器的对话框中，按下 Delete 按钮可以删除选定的触发器。如下图。



6 弹出对话框要求确认是否真的要删除该触发器。单击“是（Y）”即可。



7 通过本例，学会了如何在 Enterprise Manager 中查看触发器的定义，查看与触发器相关的其他数据库对象名称，以及如何在 Enterprise Manager 中删除触发器。