需要全本请访问 www.ertongbook.com

第 11 章　许可管理

许可的概念在 SQL Server 中非常重要，使用许可用以指定授权用户使用的数据库对象以及对这些对象可以执行的操作，如果用户没有被明确地赋予对数据库中某个对象的访问权限，就不能访问该对象。

在 SQL Server 数据库系统中，许可有多种层次，可以分配许可给单个用户或者用户自己建立的角色，也可以增加到 Widow NT　组上。另外，要注意，分配许可给一个组或者一个角色要比分配给单个用户容易。

本章主要介绍以下内容：

- 许可类型；
- 验证许可；
- 授予许可；
- 否定许可；
- 回收许可。

11.1　许可类型

在 SQL Server 系统中，有三种许可类型：语句许可、对象许可和预定义许可。

11.1.1　语句许可

语句许可通常授予那些需要在数据库中创建对象或者修改对象、执行数据库和事务日志备份的用户。这是 SQL Server 中功能最强大的许可，如果给用户授予语句许可，用户便具有了创建对象的权利，可以使用以下语句。

- CREATE DATABASE：该许可允许用户在服务器上创建数据库。
- CREATE DEFAULT：该许可允许用户在当前数据库上创建对象。
- CREATE PROCEDURE：该许可允许用户在当前数据库中创建存储过程。
- CREATE RULE：该许可允许用户在当前数据库中创建规则。
- CREATE TABLE：该许可允许用户在当前数据库中创建表。
- CREATE VIEW：该许可允许用户在当前数据库中创建视图。
- BAKUP DATABASE　：该许可允许用户创建一个具有访问权限的数据库的备份。
- BAKUP LOG　：该许可允许用户创建一个具有访问权限的数据库的事务日志备份。

11.1.2　对象许可

对象许可一般分配给数据库层次上的对象，并允许用户访问和操作数据库中已经存在的对象，如果没有找到对象许可，将不能用数据库中的任何对象。对象许可有以下几种：

- SELECT：该许可授予数据库中某个特定表的用户，如果具有这种许可，就能访问、

存储该表上的数据。

■ INSERT：该许可授予数据库中某个特定表的用户，如果具有这种许可，就能在表中增加数据。

■ UPDATE：该许可授予数据库中某个特定表的用户，如果具有这种许可，就能修改数据库中的数据。

■ DELETE：该许可授予存储过程的用户，如果具有这种许可，就可以删除该存储过程。

■ EXECUTE：该许可授予存储过程的用户，如果具有这种许可，就可以运行该存储过程。

■ REFERENCES：这是一种特殊的许可，允许用户使用一个主键/外键关系把两个表连接起来。

11.1.3 许可状态

许可有三种状态：禁止、撤消和授予，禁止许可是许可的最高级别，一旦许可对用户禁止，用户就不能访问数据库对象，即使在另外的级别授予许可。比如，Tom 是 Authors 组中的成员，Authors 角色在 Sal 表上被授予 SELECT 许可，而 Tom 在 Sal 表上被撤消许可，所以 Tom 便不能访问 Sal 表。

撤消就是删除以前分配给用户许可的禁止或者授予状态，但是另一个级别上的授予或者禁止不受影响，比如：Tom 是 Authors 组中的成员，Authors 角色在 Sal 表上被授予 SELECT 许可，而 Tom 在 Sal 表上被撤消了 SELECT 许可，但是此时 Tom 仍然能够访问 Sal 表，如果要完全撤消 Tom 对该表的访问，必须对 Tom 禁止这个许可或者在 Authors 角色中撤消 SELECT 许可。

授予可以删除禁止或者撤消许可。如果一个许可在任何其他级上被禁止，用户将不能使用该功能；如果许可在另一级上被撤消，用户仍然能够使用该功能。比如，Tom 是 Authors 组中的成员，Authors 角色在 Sal 表上被授予 SELECT 许可，如果 Authors 角色在 Sal 表上撤消 SELECT 许可，Tom 仍然能够访问 Sal 表。

11.2 验证许可

针对每一个数据库，管理员可以为用户指定执行某些操作的许可。当用户执行某个操作的时候，系统首先进行许可检查，如果有许可，可以执行该操作，如果没有许可，不允许执行该操作，并且返回错误信息。SQL Server 系统按照如下步骤验证许可是否有效：

（1）当用户执行某项操作的时候，该操作所涉及的 Tranact-SQL 语句就会被发送到 SQL Server 系统中。

（2）当 SQL Server 系统接收到这些 Transact-SQL 语句的时候，检查用户是否具有执行这些语句的许可。

（3）如果用户没有执行 SQL 语句相应的许可，SQL Server 系统不执行该操作并返回错

误信息；如果用户有相应的许可。SQL Server 系统将执行相应的操作。

验证许可的流程如图 11-1 所示。

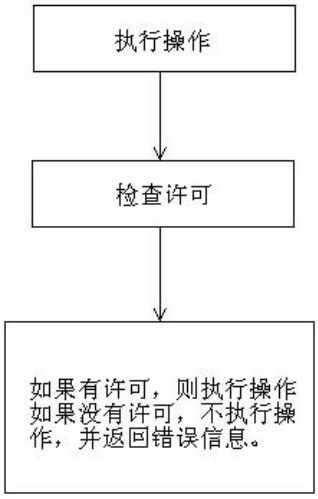


图 11-1　验证流程图

11.3　管理许可

11.3.1　授予许可

可以使用 SQL Server Enterpri Manager　或者 Transact-SQL 语句授予许可，当授予许可的时候，应该考虑以下因素。

- 只能在当前数据库中授予许可。
- 默认情况下，sysadm 、db-owner 和 db-surityadm　的成员和数据库对象所有者具有授予许可的权限。

使用 SQL Server Enterpri Manager　授予许可的操作步骤如下：

（1）运行 SQL Server Enterpri Manager　并连接到使用许可的服务器上。

（2）展开服务器树，显示服务器上的对象目录，然后单击数据库文件夹左边的“ + ”，显示服务器上的所有数据库。

（3）单击 pubs 数据库文件夹左边的“ + ”号，显示 pubs 数据库的所有对象目录，单击 Table　文件夹，在右边的窗格内找到 sal　表，单击鼠标右键，选择所有任务，在所有任务子菜单中，单击 Manage Permons　，打开如图 11-2 所示的 Objet Properti　对话框。

（4）如果要禁止许可，在与用户对应的方框上单击两下，直到看到一个红色的 X，如果要授予用户许可，在对应的方框上单击一下，直到看到一个勾号。

（5）完成许可分配后，单击“ 确定 ”按钮，SQL Server 将分配这些许可。

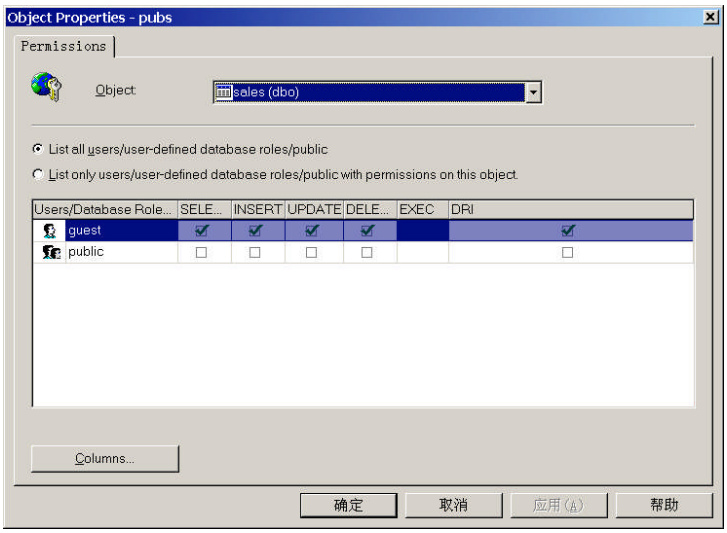


图 11-2　Objet Properti　对话框

另外，也可以在 SQL Query Analyzer 中运行 SQL 语句来分配许可。常用的分配许可的语句有 GRANT、DENY 和 REVOKE，下面讲述 GRANT 语句的语法形式。

语句许可的语法形式如下：

GRANT{ALL|statet[　，…n]}TO surity_acunt[　，…n]

对象许可的语法形式如下：

```
GRANT
{
{ALL[PRIVILEGES]|perm[　，…n]}
[(cum[　，…n])]ON{tabl|vi}
|ON {Stored_produre|exteded_produre}
}
TO surity_acunt[　，…n]
[AS {group|ro}]
```

下面是一个分配许可的例子。程序中先把 SELECT 许可赋予角色 Authors，然而，又对 Tom 用户进行了额外授权，这样，Tom 就拥有对表 sal　的全部许可。

```
USE pubs
GO
GRANT SELECT
ON sal
TO Authrs
GO
GRANT To
ON sal
TO To
GO
```


需要全本请访问 www.ertongbook.com

下面的例子把语句许可 CREATE VIEW 和 CREATE PROCEDURE 授权给其他用户和角色：

```
USE pubs
GRANT CREATE VIEW
TO To ，[Authrs]
USE
GRANT CREATE PROCEDURE
TO to ，[Authrs]
```

11.3.2　否定许可

有时候可能需要限制某些用户或者角色的访问权限，这时可以通过否定许可来达到这一目的。否定许可相当于。

- 删除以前授予用户或者角色的许可；
- 禁止从另外一个角色中继承的许可；
- 确保用户或者角色不能从任何其他角色中继承许可。

在否认许可的时候，需要考虑以下因素：

- 只能在当前数据库中否认许可；
- 默认情况下，sysadm ，db_owr ，db_surityadm 的成员和数据库对象所有者具有否定许可的权限。

可以使用 DENY 语句否认许可，DENY 语句的语法形式如下：

否认许可语句的语法形式如下：

```
DENY {ALL | statet [ ,.] | TO surity_acunt[,.]
```

否认对象许可的语法形式如下：

```
DENY{ALL|satet[ ，…n]|TO surity_acunt[ ，…n]}
{[(cum[ ，…n])]ON{tabl|vi}
|ON{tabl|vi}[( cum[ ，…n])]
|{Produre|exteded_produre}}
TO surity_acunt
```

下面的例子否认对象许可。先为 Authors 角色授予 SELECT 许可，然后，又否认了 Tom 的许可，从而 Tom 就没有访问 Sal 表的许可。

```
USE pubs
GO
GRANT SELECT
ON sal
TO Authrs
GO
DENY To
ON sal
TO To
```

```
GO
```

下面的例子否认了 Tom 创建视图和创建存储过程的许可。

```
USE pubs
DENY CREATE VIEW
TO To
USE
DENY CREATE PROCEDURE
TO To
```

11.3.3 收回许可

收回许可可以禁止使用授予的许可或者否认许可。在收回许可的时候，需要考虑以下因素。

- 只能在当前数据库中收回许可。
- 默认情况下，sysadm 、db_owr 、db_surityadm 的成员和数据库对象所有者具有收回许可的权限。
- 收回许可本质上就是删除系统表 sysprotets 中的内容，这部分内容是由授予许可和否认许可创建的。

收回许可可以使用 REVOKE 语句，REVOKE 语句的语法形式如下：

收回许可的语法形式如下：

```
REVOKE{ALL|statet[ ，…n]||FROM surity_acunt[ …n]
```

收回对象许可的语法形式如下：

```
REVOKE[GRANT OPTION ON]
{ALL[PRIVILEGES]|ON{tabl|vi}
|ON{stored_produre|exteded_produre}
}
FROM surity_acunt[ ，…n]
[AS{group|ro}]
```

下面是一个收回对象许可的例子，从多个用户收回对象许可。

```
USE pubs
REVOKE SELECT，INSERT
ON sal
FROM To Jo
```

下面是一个收回语句许可的例子，从 Tom 收回 CREATE TABLE 语句许可。

```
REVOKE CREATE TABLE
FROM To
```

11.4　小　结

本章介绍了许可的概念、类型、系统验证许可的过程以及许可的三种状态。通过本章的学习，应掌握以下内容。

- 使用许可可以指定授权用户可以使用的数据库对象以及对这些对象可以执行的操作
- 在 SQL Server 系统中，有三种许可类型：语句许可、对象许可和预定义许可。
- 创建数据库或者创建数据库中的其他内容所需要的许可类型称为语句许可。
- 使用数据或者执行存储过程所需要的许可称为对象许可。
- 可以使用 GRANT 语句授予许可。
- 可以使用 DENY 语句否定许可。
- 可以使用 REVOKE 语句收回许可。

第 12 章　数据库备份

计算机的广泛应用，给人们的生活带来很大方便。但是，也带来了一些负面的影响，比如计算机犯罪和计算机病毒的出现。一个对计算机依赖性很强的公司，可能由于黑客攻击或者计算机病毒等原因，引起计算机的崩溃，造成数据的丢失，使公司陷于瘫痪之中。造成的损失也是无法弥补的。为了防止灾难性事故和不可避免的损失，数据库的备份工作就是一项非常重要的系统管理工作。

本章主要介绍下列内容：

- 备份的概念；
- 何时进行备份；
- 备份准备工作；
- 如何执行备份；
- 备份类型。

12.1　概　述

数据库备份就是制作数据库结构和数据的拷贝，以便在数据库遭到破坏时能够恢复数据库。备份就是一项数据库维护的系统管理工作。

在 SQL Server 2000 中，数据库的备份是动态的。也就是说，在执行数据库备份的时候，用户也可以访问数据库，并对数据库进行操作。在备份数据库的时候，SQL Server 执行以下操作：

- 把数据库所有的数据页写到备份介质中；
- 记录最早的事务日志记录的日志序列号；
- 把全部的失误日志记录写到备份介质中。

制作备份数据库需要一定的许可。在 SQL Server 系统中，应该有专门的管理员负责数据库的备份工作。只有下列角色才能备份数据：

- 固定的服务器角色 sysadmin；
- 固定的数据库角色 db_owner；
- 固定的数据库角色 db_backupoperator。

另外，管理员也可以授权某些角色或用户执行数据库备份。

执行备份后就把需要备份的数据写到了备份介质上。备份介质是备份内容的载体。在 SQL Server 数据库管理系统中，有以下 3 种备份介质：

- 硬盘文件；
- 磁带；
- 命名管道（Named Pipe）。

磁盘文件是最普遍的备份介质，既可以是本地文件，也可以是网络文件。使用磁盘作为备份介质时，磁带只能安装在本地 SQL Server 系统上。命名管道实际上是利用第三方软件包执行数据库备份。

在备份数据库的时候，应该将系统的全部信息都备份下来，以便当系统发生故障时，能够将系统恢复到发生故障之前的状态。一般情况下，需要备份两方面内容：一是备份记录系统信息的系统数据库；二是备份记录用户数据的用户数据库。系统数据库主要包括 master、msdb 和 model 数据库，主要记录 SQL Server 系统和用户数据库信息。用户数据库存储了用户数据。这些数据非常重要，是备份的主要内容。

12.2　准备备份

在什么时候备份数据库，对于系统数据库和用户数据库是不同的，本节主要介绍什么时候备份数据库以及如何进行备份前的准备工作。

12.2.1　备份系统数据库

master 数据库包含了 SQL Server 系统中有关数据库的全部信息，如果创建了用户自定义对象，master 数据库就被修改，此时，就应该备份 master 数据库。这样，当 master 数据库遭到破坏时，就可以恢复原来的系统和用户数据库。如果没有 mater 数据库的备份，可以使用重建系统数据库的命令行程序 rebuildm.exe 重建全部系统数据库。

另外，在执行某些 Transact-SQL 或系统存储过程的时候，master 数据库也会被修改。因此，在执行了以下操作之后，应该备份数据库。

- 修改事务日志的系统存储过程 sp_logdevice。
- 增加或删除服务器的系统存储过程。
- 创建、修改、删除数据库。例如执行 CREATE DATABASE、ALTER DATABASE、DROP DATABASE 命令等。

msdb 存储了有关工作、警报和操作员等 SQL Server Agent 服务的信息。如果修改了 msdb 数据库，就应该对该数据库进行备份。如果没有 msdb 数据库的当前备份，就必须重建所有系统数据库，然后创建每一个工作、警报和操作员。

model 数据库存储了用户数据库的默认配置信息。如果修改了 model 数据库，就应该对该数据库进行备份。

12.2.2　备份用户数据库

用户数据库的备份工作是数据库管理的一项常规工作，应该有计划地备份用户数据库。此外，在创建数据库和索引之后，也应该执行用户数据库的备份工作，这个不是必须的，但是，在数据库遭受破坏之后，如果已备份了有索引的数据库，在执行恢复数据库操作的时候，就可以大大节省恢复进程的时间。如果在索引创建之后，没有备份数据库，只备份事务日志，在恢复事务日志的时候，SQL Server 系统还必须重建索引。创建索引的时间很长，甚至比恢复完全数据库备份所需要的时间更长。

可以使用下面的语句清除事务日志：

```
BACKUPLOG WITH TRUNCATE_ONLY
```

清除了事务日志后，事务日志中就不包含数据库活动和操作的记录，从而不能用来恢复数据库，所以，在清除事务日志之后，应该备份数据库。

另外，如果执行了不写入事务日志的操作，也应备份数据库。因为在执行了这种操作之后，事务日志中不再包含数据库活动的记录，不能用来恢复数据库。

不写入事务日志的操作包括下列语句：

- SELECT INTO 语句和命令行实用程序；
- BACKUPLOG WITH NO__LOG 语句；
- WRITETEXT 或者 UPDAETEXT 语句。

一般情况下，数据库的备份是动态的。但是，如下操作会影数据库的备份：

（1）使用 CREATE DATABASE 或者 ALTER DATABASE 语句创建或修改数据库。

（2）创建索引。

（3）加载数据或使用 WRITETEXT、SELECTINTO、UPDATETEXT 语句。

在执行数据库备份的时候，尽量避免以上操作。

12.2.3 执行备份前的准备

在执行备份之前，应该制定一个完整可行的备份计划，以保证数据库系统的安全。为此，需要考虑以下方面。

- 确定备份的内容。备份的内容包括系统库中的数据和用户数据库中的数据。在执行备份的时候，一定要将备份的内容完整地备份。
- 确定备份的频率。备份频率就是每隔多长时间执行一次备份。可以根据系统恢复的工作量和系统活动的事务来确定备份的频率。
- 确定备份存储的地方。备份应该异地存放。
- 确定备份存储期限。对于一般的数据，可以确定比较短的存储期限，对于重要的数据，则需要确定一个比较长的期限。
- 确定备份的介质。

12.3 执行备份

在备份数据库之前，首先要创建备份文件。在 SQL Server 2000 中，有多种备份数据库的方法。下面分别介绍。

12.3.1 创建临时性备份文件

使用 BACKUP DATABASE 语句可以创建临时性备份文件。如果所要创建的数据库备份文件不会长久使用，就可以创建临时的备份文件。

创建临时性备份文件有两种方法：

需要全本请访问 www.ertongbook.com

- 使用 SQL Server Manager；
- 使用 BACKUP DATABASE 语句。

在创建临时性文件的时候，必须指定介质类型、路径名称和文件名称。BACKUP DATABASE 的语法形式如下：

BACKUP DATABASE{database_name|@database_name_var}

TO<backup_file>[，…n]

其中，backup_file 是

{{backup_file_name|@backup_file_name_var}|[DISK|TAPE|PIPE]}

={temp_file_name|temp_file_name_var}

下面这段代码在磁盘上创建了一个临时备份文件，并把 master 数据库备份到该备份文件中。

```
USE master
BACKUP DATABASE northwind TO DISK='G:\Temp\Temp.bak'
```

12.3.2　创建永久性备份文件

创建永久性备份文件的时候，应该考虑以下因素：

- 应该指定备份文件的逻辑名称和物理名称；
- 在 master 数据库的表 sysdevices 中，创建永久性备份文件的逻辑名称和物理名称；
- 一个数据库最多可以创建 32 个备份文件。

创建永久性备份文件有两种方法：

- 使用 SQL Server Manager；
- 使用系统存储过程 sp_addumpdevice。

1．使用 SQL Server Manager 创建永久性备份文件

操作如下：

（1）运行 SQL Server Manager，打开服务器，选择 Backup，单击鼠标右键，弹出一个快捷菜单，如图 12-1 所示。



图 12-1　备份文件菜单

（2）在该快捷菜单中，选择 New Backup Device 菜单项，弹出如图 12-2 所示的 Backup Device Properties-New Device 对话框。

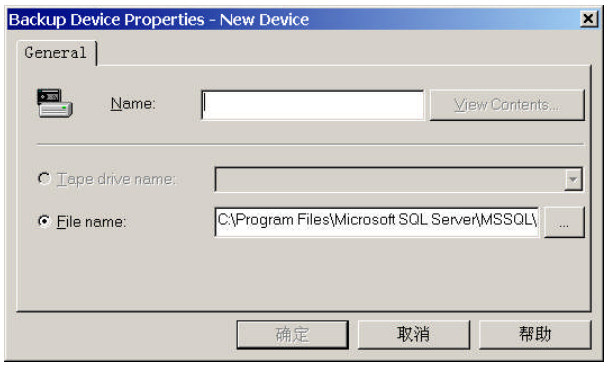


图 12-2　New Device 对话框

（3）在该对话框中，输入备份文件的逻辑名称和物理名称。

（4）输入完成后，单击“ 确定 ”按钮。

2．使用系统存储过程 sp_addumpdevice 创建永久性备份文件

系统存储过程 sp_addumpdeviec 的语法形式如下：

sp_addumpdevice{'device_type'}

[，'logical_name'][，'physical_name']

[，{{controller_type|'device_status'}}]

下面的例子创建了一个逻辑名称为 logical_name，物理名称为 physicak_name 的永久性备份文件：

USE master

EXEC sp_adumpdevice'tape'，'logical_name'，'\\physical_name'

12.3.3　使用多个备份文件进行备份

在 SQL Server 2000 中，可以进行并行备份，并行备份允许 SQL Server 系统同时向多个备份文件进行操作，这些文件存储了一个并行的备份集。

使用多个文件进行备份可以大大节省数据库备份的时间，其语法形式如下：

BACKUP DATABASE{database_name|@database_name_var}

TO<backup_file>[，…n]

[[,][NAME={backup_set_name|@backup_set_name_var}]]

其中，<backup_file>是

{backup_file_name|@backup_file_name_evar}|

{DISK|TAPE|PIPE}={'temp_file_name'|@temp_file_name_var}

使用多个备份文件进行备份的时候，应该注意以下方面。

- 备份集中可以混合使用永久性文件和临时文件。
- 在同一次备份操作中，所有的备份设备必须是同一种介质。
- 如果格式化备份集中的一个成员，其他成员中的数据会变得无效。

在 BACKUP DATABASE 语法形式中，NAME 选项给出备份集的名称。NAME 选项将多个备份文件与一个备份集的成员互相联系起来。如果使用多个备份文件，就应该使用 NAME 选项。

12.3.4　使用 BACKUP 语句

备份操作有以下 3 种方法：

- 使用 SQL Server Enterprise Manager；
- 使用 Transact-SQL 语句；
- 使用备份向导。

无论使用哪一种方法创建备份，都涉及到一些常规选项，在使用 Transact-SQL 语句的时候，涉及到的选项如下：

```
BACKUP DATABASE{database_name|@database_name_var}
TO<backup_file>[，…n]
[WITH
[[，]FORMAT]
[[，][INIT|NOINIT]
[[，]RESTART]
]
```

各个选项的含义如下：

（1）INIT 和 NOINIT 选项

如果使用 INIT 选项，则 SQL Server 系统将覆盖掉备份设备上除标题之外的任何信息，如果使用 NOINIT 选项，则 SQL Server 系统将备份附加在某一个已经存在的备份文件或备份集合上。默认情况下，SQL Server 系统备份内容附加在文件上。

（2）FORMAT 选项

使用 FORMAT 选项可以覆盖备份文件的内容，使用 FORMAT 选项可以执行以下操作。

- 覆盖所有介质和备份文件的内容。
- 在备份操作的所有文件上写入新的标题信息。

✍ 使用 FORMAT 选项要非常谨慎，如果不小心格式化备份集中的一个成员，整个备份都会变成不可用。

（3）RESTART 选项

如果选择 RESTART 选项，SQL Server 系统将从中断点重新启动备份操作。

12.3.5　使用备份向导进行备份

使用备份向导进行备份的操作如下：

（1）运行 SQL Server Enterprise Manager。

（2）单击 Tools 菜单，选择 Wizards 菜单项，弹出如图 12-3 所示的对话框。

（3）单击 Management 旁边的“ ＋ ”，展开 Management，双击 Backup Wizard，启动备份向导，如图 12-4 所示。

（4）单击“ 下一步 ”按钮，出现如图 12-5 所示的选择数据库对话框。

（5）在数据库下拉式列表中选择需要备份的数据库。

（6）选择完成后，单击“ 下一步 ”按钮，弹出如图 12-6 所示的键入备份名称和注释对话框。



图 12-3　选择向导

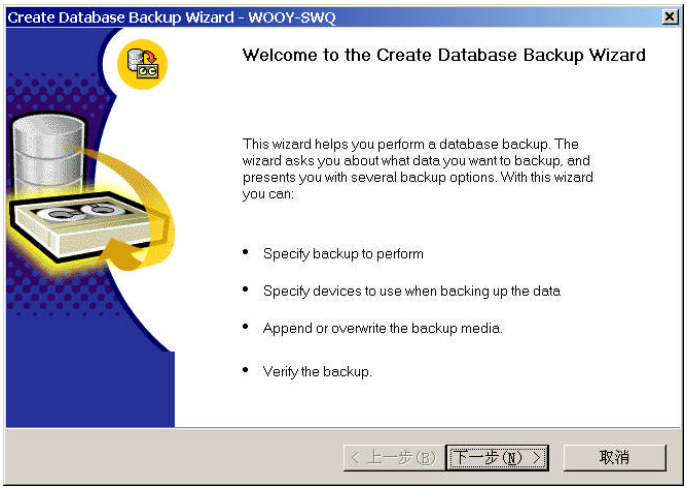


图 12-4　备份向导欢迎对话框

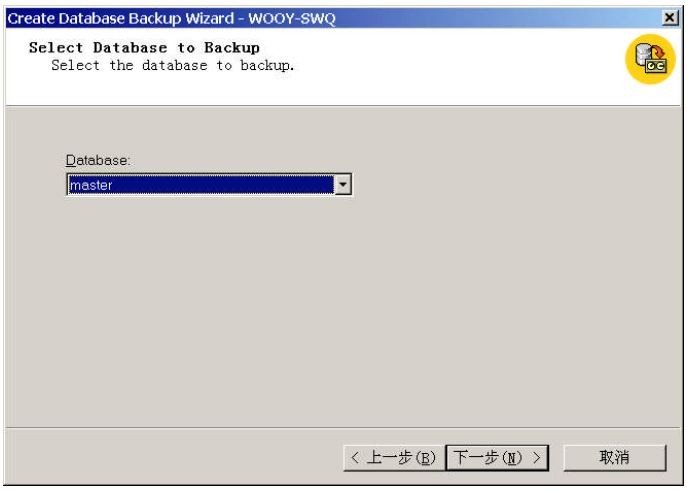


图 12-5　选择数据库对话框

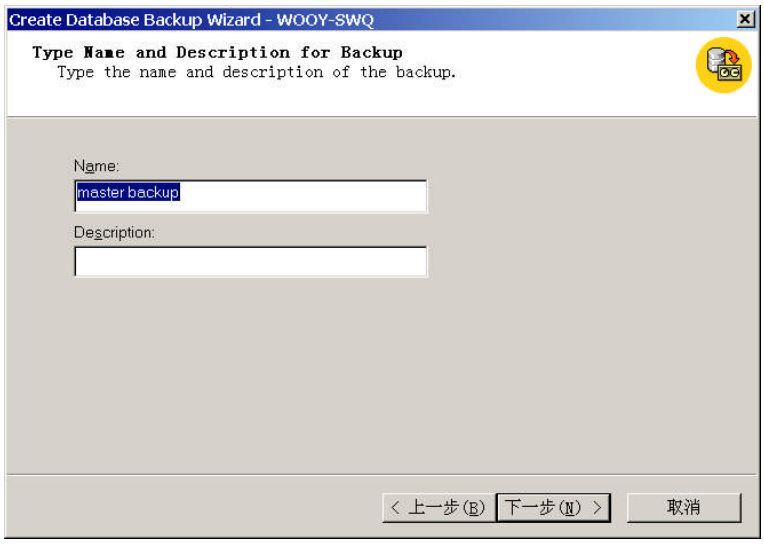


图 12-6　键入备份名称和注释

（7）键入备份的名称和注释信息，单击“ 下一步 ”按钮，弹出如图 12-7 所示的选择备份类型对话框。

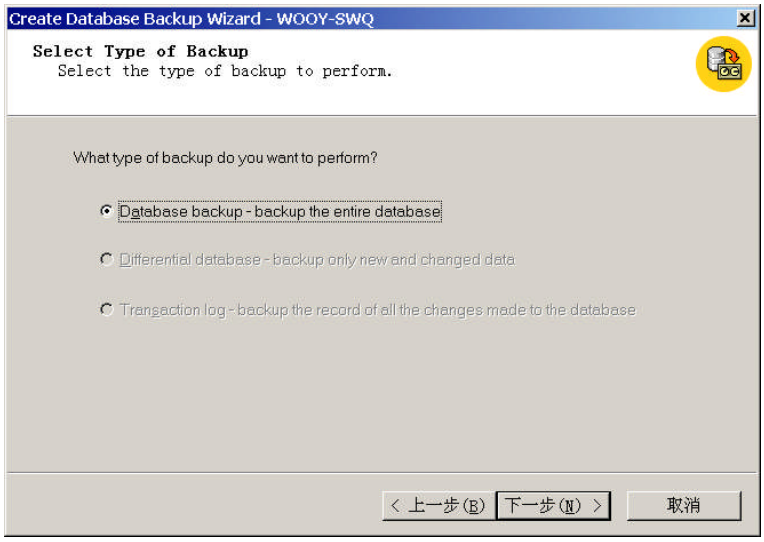


图 12-7　选择备份类型

（8）选择备份类型，然后单击“ 下一步 ”按钮。弹出如图 12-8 所示的选择备份设备并设置属性对话框。

（9）键入备份文件名称并选择备份设备，并选择是添加在原备份文件之后还是覆盖掉以前的数据。

（10）选择完成后，单击“ 下一步 ”按钮，弹出如图 12-9 所示的备份计划安排对话框。

（11）设置完相关的信息后，单击“ 下一步 ”按钮，弹出如图 12-10 所示的完成备份对

话框。在该对话框中，显示了备份信息。

执行完以上操作后，就成功地创建了备份。

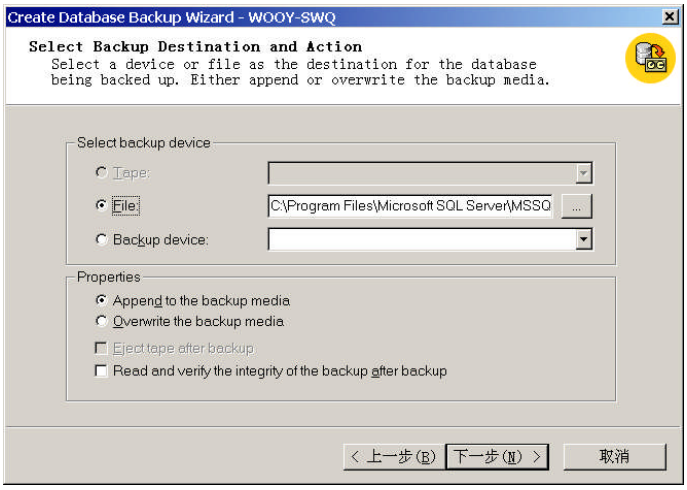


图 12-8　选择备份设备并设置属性

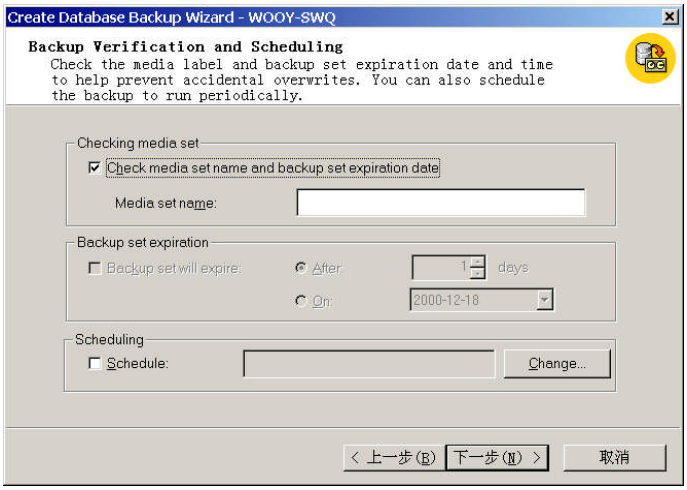


图 12-9　备份计划安排

需要全本请访问 www.ertongbook.com