

SQL SERVER 数据库自动备份软件的实现

廖辉传

(华东交通大学 信息工程学院 江西 南昌 330013)

摘要: 数据库备份是用于保障数据库信息的安全和稳定的重要技术手段. 文中提出了以 delphi 作为开发工具, 实现对 SQL Server 数据库自动备份及恢复的方法. 该方法简单实用, 在数据库系统中具有实际的应用价值.

关键词: 数据库备份; SQL Server; Delphi

中图分类号: TP 311.13

文献标识码: A

随着计算机技术的发展, 数据库技术已经成为计算机信息系统与应用系统的核心技术和重要基础, 从而使得数据库信息的安全性和稳定性就显得非常重要了. 由于人为错误、硬件故障、程序错误、计算机病毒、网络故障或自然灾害等原因而造成数据库被破坏或丢失的情况经常发生, 因此迫切需要数据库自动备份技术来保护数据库^[1]. 当数据库受到破坏或丢失时, 用户就可以使用数据库的备份文件对其进行恢复, 从而保障了数据库信息的安全和稳定.

SQL SERVER 是目前比较常用的数据库, 其企业管理器带有备份和恢复数据库的工具, 但是对于一般用户而言使用不是太方便. 因此, 备份工作的简化正是使用者所渴求的. 本文实现了自动备份 SQL Server 数据库的软件, 操作简单、实用, 解决了众多使用者的需求, 能备份多个 SQL 数据库, 可灵活设置备份周期和执行时间.

1 SQL SERVER 数据库备份与恢复方法

要实现备份技术就必须要了解 SQL SERVER 数据库备份的基本原理和相关的基础知识, SQL SERVER 2000 有以下三种备份和恢复的方法^[2]:

1) 通过企业管理器的数据库备份和还原功能.

2) 在查询分析器里执行 “sp_detach_db ‘dbname’, true” 分离数据库, 将 *.mdf (主数据文件) 和 *.ldf (日志文件) 拷贝出来, 当然也可以通过停止 SQL Server 服务来拷贝数据库文件, 即冷备份.

3) 在查询分析器里执行 “backup database ‘dbname’ to disk = 文件名” 来备份数据库, 执行 “restore database ‘dbname’ from disk = 文件名” 可还原数据库.

第一种方法是利用 SQL SERVER 2000 的企业管理器自带的数据库备份和恢复的功能, 可以直接使用, 第二种方法是通过在查询分析器执行数据库分离的 SQL 语句来实现的, 也可以通过拷贝文件来实现冷备份. 而第三种方法就是要在查询分析器里执行 SQL 语句来备份和还原数据库. 本文就是采用第三种方法的原理, 在 DELPHI 程序中使用 SQL 语句来实现数据的备份和还原功能.

2 系统的主要功能

数据库自动备份软件的主要功能包括手动备份、定时备份、数据库还原. 手动备份时用户可以随时对需要备份的数据库进行选择, 可同时对多个数据库进行备份. 定时备份是用户可事先设置好备份的时间和周期, 使得系统到时能自动备份, 能更好地

保障数据安全. 数据还原也是一个重要的功能, 当用户的数据库遭到破坏时能够及时地恢复已备份的数据. 为了使得程序更具实用性, 系统还需要在开机后自动在后台运行的功能. 最后还需要托盘功能, 为了让程序不占据桌面, 可将程序放入托盘区, 当用户需要手动备份时, 可以从托盘区图标唤醒程序.

3 系统的具体实现

1) 数据库备份功能的具体实现

在备份前, 用户需要从列表框中选择一个或多个数据库. 备份时, 程序将在用户指定的文件夹中自动创建以数据库名称命名的文件夹, 并将数据库文件备份到所创建的文件夹中, 备份后的数据库文件名按系统时间命名, 即为“年月日时分秒. SQ -”. 通过 TADOQuery 控件的 Sql. Text 属性设置备份命令, 用 ExecSQL 方法执行命令, 就实现了对数据库的备份. 主要代码如下:

```
//开始备份
for i:=0 to clstdb. Items. Count - 1 do
// clstdb 是数据库列表控件
begin
    if cLstDb. Checked[i] then
    begin
        sDir:= pa + cLstDb. Items[i] + '\';
        if not directoryexists ( sDir) then CreateDirectory
        ( PChar( sDir) ,nil); //创建文件夹
        //文件名等于 <年月日时分秒>
        bkName:= sDir + FormatDateTime ( 'yyyymmddh-
        hhmss',Now) + '.SQ -';
        panel1. Caption:= bkName;
        panel1. Refresh; //刷新 panel1
        bkStr:= Backup Database ' + cLstDb. Items[i] + ' to Disk = '~+
        bkName + '.SQ -'; //用于备份的 SQL 语句
        with adoQr do //adoQr 为 query 控件
        begin
            close;
            Sql. Text:= bkStr;
            ExecSQL;
        end;
    end;
end;
```

2) 定时备份的实现

定时备份前, 用户指定自动备份时间, 并启用定时器控件实时检查系统时间, 当系统时间到达用户指定的时刻则调用以上过程执行备份. 在启用定时

备份前, 需要判断用户是否选择了可备份的数据库, 是否指定了正确的备份路径, 是否指定了备份的时刻. 设计时可使用三个 TComboBox 控件用于显示用户设置备份的时间, 分别用来表示时、分、秒. 同时也要添加一个 CheckBox 控件, 用来选择定时备份.

选择定时备份以后就应该将时间的设置控件设置为不可用, 同时也要将手动备份控件和每周一次备份控件也设置为不可用. 最后只保留保存设置按钮和程序托盘按钮为可用, 其他的一律不可用. 用户设置好了备份时间之后, 就需要检查系统时间是否达到了用户所设置的时间. 可在窗体中添加一个 T-Timer 控件, 如果到了就执行备份, 如果还没有到就继续检测系统时间, 直到到了用户设置的时间才执行备份, 否则不执行备份.

定时备份的主要流程如下:

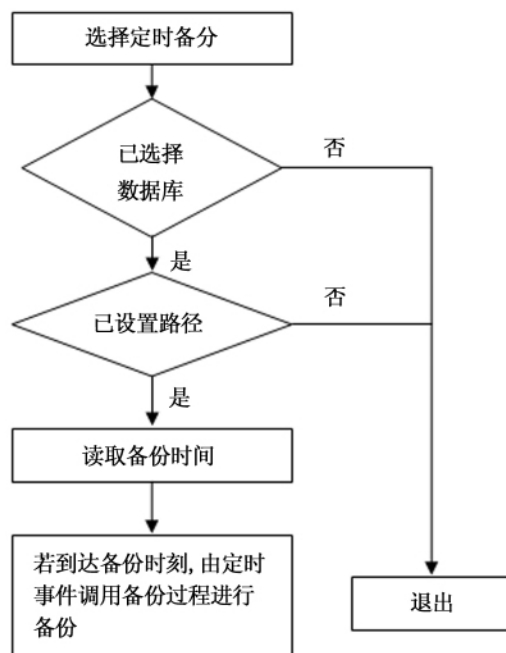


图1 定时备份功能流程图

主要代码(时间控件的触发事件):

.....

```
bktime:= ''; //bktime 用于存放用户指定的备份时刻
if length( cmbH. Text) = 1 then bktime:= bktime + 0;
bktime:= bktime + cmbH. Text; //获取指定的小时
if length( cmbM. Text) = 1 then bktime:= bktime + 0;
bktime:= bktime + cmbM. Text; //获取指定的分
if length( cmbS. Text) = 1 then bktime:= bktime + 0;
bktime:= bktime + cmbS. Text; //获取指定的秒
```

.....

```
cTime:= FormatDateTime( 'hhmmss',Now);
//监测系统时间
```

```

if cTime = BkTime then
begin
    timer1.Enabled: = false;
    SqlBackup(); //用于备份的事件 procedure ,主要代码
    见上一小节 3.1
    timer1.Enabled: = true;
end;
end;

```

3) 程序开机运行和托盘的实现

要实现定时备份,必须让程序一直处于运行中,那么就有必要让程序在开机后自动运行,在注册表“HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run”位置创建一个开机运行项,即可实现程序开机自动运行。

在启用定时备份后,程序只需要在后台工作,为了让程序不占据桌面,可以将程序放入托盘区,当用户需要手动备份时,可以从托盘区图标唤醒程序。为了实现程序的托盘,需要使用 notifyicondata 结构类型,并使用 API 函数 shell_notifyicon 实现托盘区图标的添加和删除,同时需要创建托盘区图标的单击事件。

让程序开机运行,就是让程序在电脑启动的时候就运行,同时 SQL Server 服务器也运行,但是因为有些电脑主机启动程序的速度很慢导致 SQL Server 服务器启动很慢,而备份软件的启动却很快,因而导致备份软件与 SQL Server 服务器连接失败。为了解决这个问题,就必须在程序中添加一个刷新程序,刷新程序是在 SQL Server 服务器以后运行的,一般是在备份软件运行 20 秒之后。这样就可以确保备份软件与 SQL Server 服务器的成功连接。那么也就需要再添加一个 TTimer 控件来倒计时,即在倒计时 20 秒之后激发刷新事件。

4) 数据库还原功能的设计

要进行数据库还原,首先必须要选择需要进行还原的数据库,其次是要选择还原设备(备份文件)。设计时需要添加两个 SpeedButton 按钮和一个 OpenFileDialog 控件,还有一个 CheckBox 复选框,复选框用于选择数据库还原功能。选择还原设备的控件为 SpeedButton3,其单击事件的具体代码如下:

```

procedure TForm1.SpeedButton3Click(Sender: TObject);
//选择还原设备文件
begin
    OpenFileDialog1.Execute;
    Panel1.Caption: = OpenFileDialog1.FileName;
    pa: = Panel1.Caption;

```

```

end;

```

选择好还原文件以后,下一步就可以实现具体的还原功能了。其实数据库的还原功能,主要是在程序中执行这样一条 SQL 语句来还原数据库的,在查询分析器里执行“restore database ‘dbname’ from disk = 文件名”可还原数据库。在本软件的设计中,需要在相应的 button 控件中添加单击事件来实现该功能,主要代码如下:

```

try
    for j: = 0 to clstdb.Items.Count - 1 do
    begin
        if cLstDb.Checked[j] then
        begin
            bkName1: = pa;
            bkStr1: = 'restore Database ' + cLstDb.Items[j]
            + 'from Disk = ' + bkName1 + '
            with adoQr do
            begin
                close;
                Sql.Text: = bkStr1;
                ExecSQL;
            end;
        end;
    end;
    cLstDb.ItemIndex: = -1;
    MessageBox(handle, '数据库还原成功!', '提示', MB_
    ICONINFORMATION + mb - Ok);
except
    cLstDb.ItemIndex: = -1;
    MessageBox(handle, '数据库还原失败!', '错误', mb -
    IconWarning + mb - Ok);
end;

```

其实数据库还原的运行环境要求很严格,因为要对数据库进行还原,必须要保证被还原的数据库不能被其它程序或设备使用,这样才能保证数据库的正确恢复和还原。

3 用户界面运行效果

以上介绍了数据库备份软件的几个主要的功能及其设计方法,由于篇幅关系,还有一些设计细节尚未提及,比如数据库的连接、备份信息的保存、程序的独自运行等。本软件所有功能在 Delphi 7.0 + SQL Server 2000 环境下经测试备份和还原功能均能正常工作,最终的软件运行界面如图 2 所示。



图2 软件运行界面

从图2可以看到,用户可以选择一个或多个需要备份的数据库,并在下方选择备份文件存放的位置,然后单击“手动备份”按钮即可在指定位置生成



图3 运行结果

备份文件(备份时会在指定目录下自动生成一个与数据库同名的目录),图3是备份SQL SERVER自带的数据库 Northwind 的运行效果。

4 结束语

SQL SERVER 是目前比较常用的数据库,其企业管理器虽然自带备份和恢复数据库的工具,但是对于一般用户而言使用时有一定难度,容易让人望而生畏。本系统完成之后,可以大大简化SQL Server 数据库的备份工作,可以有效地保护好数据库信息的安全和稳定,在数据库应用系统中具有一定的实用价值。

参考文献:

- [1] 苗雪兰,刘瑞新,宋会群. 数据库系统原理及应用教程[M]. 北京: 机械工业出版社, 2004.
- [2] 宋振会. SQL Server 2000 中文版基础教程[M]. 北京: 清华大学出版社, 2005.
- [3] 启明工作室. Delphi + SQL Server 数据库应用系统开发与实例[M]. 北京: 人民邮电出版社, 2005.
- [4] 刘瑞新. Delphi 程序设计教程[M]. 北京: 机械工业出版社, 2005.

Implementation of SQL SERVER Database Automatic Backup

LIAO Hui - chuan

(School of Information Engineering, East China Jiaotong University, Nanchang 330013, China)

Abstract: The database backup is an important way for the safety and stability of database information. This paper describes the method to backup and restore SQL SERVER database under the environment of Delphi. Easy and practical, the method has great practical value in the application of database system.

Key words: database backup; SQL Server; Delphi

(责任编辑: 王建华)