

# 一种基于触发器技术的 SQL Server 数据库实时备份方案\*

## Real-time Backup Solution for SQL Server Database Based on Trigger Technique

周舟

ZHOU Zhou

(广西建设职业技术学院城市建设与交通工程系, 广西南宁 530003)

(Department of Urban Construction and Traffic Engineering, Guangxi Polytechnic of Construction, Nanning, Guangxi, 530003, China)

**摘要:**基于触发器技术的 SQL Server 数据库实时备份方案使用双机备份方式,根据触发器自动执行数据备份的代码和关系数据库存储数据变化前后的状态检索、数据变化痕迹,可以快捷地实现数据恢复,具有很强的实时性和直观性。

**关键词:**数据库 备份 触发器

**中图分类号:**TP309.3 **文献标识码:**A **文章编号:**1002-7378(2009)03-0210-03

**Abstract:** SQL Server database real-time backup based on the trigger techniques is double server backup method. Database automatically backup by running the special procedure deployed in the trigger and realizes database recovery rapidly according to the data states and the modification tracks stored in relation tables. It is shown that the proposed solution is real-time and intuitive.

**Key words:** database, backup, trigger

在 SQL Server 系统中,备份与恢复是应对数据灾难的有效方法。一般而言,数据备份与恢复常用 3 种方法<sup>[1~3]</sup>: (1)使用企业管理器提供的图形化向导; (2)使用系统提供的 Backup/Restore 命令; (3)使用数据文件的分离/附加。然而,这几种方法都是非实时的,缺陷很显而易见。首先,在两次备份的时间间隔上,若定时时间太长,在最后一次成功备份与发生故障之间丢失的数据就可能越多;定时备份时间太短,备份所需的时间代价太大,特别是数据库的规模较大时更是如此。其次,尽管前两种方法可以进行增量式备份和恢复,虽然能够减少备份和恢复时间,但是对备份数据的维护和数据恢复增加了困难,而且增量备份只能是阶段性的,无论数据库备份时

间长短,在两次备份之间增删改的数据均无法备份和恢复。

我们注意到,在 SQL Server 中,触发器被定义为一种因数据的 INSERT、DELETE 或 UPDATE 事件触发而自动执行的存储过程,具备基于事件驱动的即时响应和自动执行的能力<sup>[1]</sup>。SQL Server 允许为任何给定的 INSERT、DELETE 或 UPDATE 3 个事件的任意组合创建多个触发器。从 SQL Server 2000 开始,系统提供 AFTER 和 INSTEAD OF 两种类型的触发器,依附于特定的表或视图且由唯一的名称标识,INSTEAD OF 类型的触发器将产生两个逻辑表<sup>[4]</sup>: inserted 和 deleted。inserted 和 deleted 由系统自动创建和撤销,与触发器所依附的表或视图具有相同的属性结构,触发器工作时以这两个特殊的临时表存放受到影响的数据行的副本,实际上是事务日志的视图。3 种数据变化触发事件及其产生的临时表遵循如下规则:如果只产生 deleted 而不产生 inserted,则是 DELETE 事件;只产生 inserted

收稿日期:2009-04-13

作者简介:周舟(1976-),女,硕士,讲师,工程师,主要从事暖通及电气信息化技术研究。

\* 广西社科基金项目(06FTQ002)资助。

的INSERT事件;此外,同时产生deleted和inserted的事件是UPDATE事件,在事务操作上等同于先执行DELETE再执行INSERT。

鉴于此,我们提出一种新的备份方案,在不抛弃传统的定时备份的方法的前提下,使用SQL Server系统的触发器机制,将数据的增删改实时记录下来,实现对数据的增量式备份。

### 1 基于触发器的双机备份过程

#### 1.1 模型

基于触发器的双机数据备份模型如图1所示。在该模型中,采用双机备份的方法来保证其可靠性,数据源端和备份机分别部署在两台物理隔离的服务器上。

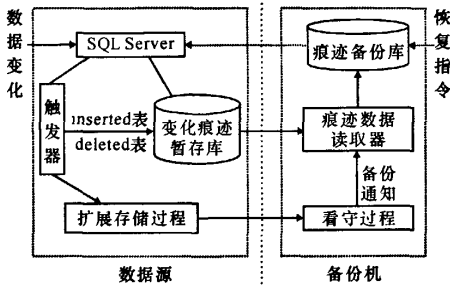


图1 基于触发器的双机数据备份模型  
→:操作指令;⋯:恢复指令,——构成部分。

#### 1.2 流程

在图1的模型中,数据备份由数据变化事件触发,备份操作的发起是事件驱动而非以定期轮询进行的,具有实时性特征,如果网络正常,对数据变化的记录几乎是瞬间完成的。以1个数据表的实时修改备份为例,假设源表为infos,暂存库中对应表名为tmp\_infos,备份机上对应表名为bk\_infos;为了使用触发器实时自动备份数据,在infos中设计INSTEAD OF类型的触发器为Backup\_TR,数据源服务器名为Tutors,备份机名为bk\_Tutors,其数据备份的流程如下:

(1)一旦数据变化事件发生,依附于数据表的触发器将数据变化前后状态写入暂存库。用SQL Server的触发器规则判断数据变化事件(INSERT、UPDATE或DELETE),写入操作的程序片断为:

```
IF EXISTS (SELECT name FROM sysobjects
WHERE name='Backup_TR' AND type=
```

```
"TR')
DROP TRIGGER Backup_TR
GO
CREATE TRIGGER [Backup_TR] ON dbo.
infos FOR UPDATE,DELETE,INSERT
AS
DECLARE @bFlag char(10)
IF NOT EXISTS(SELECT * FROM inserted)
SET @bFlag='删除'
INSERT tmp_infos SELECT *,getdate(),
suser_sname(),@bUpdate FROM deleted
ELSE
IF NOT EXISTS (SELECT * FROM deleted)
SET @bFlag='添加'
INSERT tmp_infos SELECT *,getdate(),
suser_sname(),@bUpdate FROM inserted
ELSE
SET @bFlag='更改前'
INSERT tmp_infos SELECT *,getdate(),
suser_sname(),@bUpdate FROM deleted
SET @bFlag='更改后'
INSERT tmp_infos SELECT *,getdate(),
suser_sname(),@bUpdate FROM deleted
GO
```

(2)触发器通过扩展存储过程通知备份机备份数据修改前后的状态。

(3)痕迹数据读取器将暂存库的数据读取并写入痕迹备份库,其过程很简单,并清除暂存库。使用如下语句:

```
INSERT bk_infos select * FROM Tutors..
tmp_infos
DELETE FROM Tutors..tmp_infos
```

执行上述语句只需保证备份机bk\_Tutors具有对Tutors数据库的读写权限。

经过上述3个步骤后,数据源的变化将被备份到备份机的痕迹备份库中,由于痕迹备份库为关系数据库,数据提取方便,一旦需要进行数据恢复,可以从中检索出源库修改前的数据状态并进行覆盖性恢复。由于备份时,已经记下数据变化的事件,可以通过检测备份库里的bFlag标志决定采取的恢复操作,如表1所示。

表1 通过检测标志恢复数据

bk_infos.bFlag 值	数据恢复操作
删除	将 bk_infos 的当前数据插入 infos
添加	删除 infos 中与 bk_infos 有主外键关联的数据
更改前	bk_infos 的当前数据更新 infos

此外,使用 SELECT 语句检索备份库并按时间排序,可得某条数据被修改过程的序列堆栈,便于用“后进先出”的出栈机制执行多级数据恢复。

## 2 基于触发器的双机备份的两个关键部件

### 2.1 数据源

数据源即需要备份的数据库服务器,包含数据库管理系统(DBMS)以及依附于数据表的触发器、扩展存储过程和数据变化痕迹暂存库。数据变化痕迹指每一次在数据库中执行插入、更新或删除之后,依次保留数据被修改前后的数据及修改者和修改时间等状态信息,以及由此所形成的记录序列。实时备份关心的是数据库变化瞬间的前后状态,所实时备份的数据也就是这些痕迹信息。痕迹信息可由触发器产生的 deleted 和 inserted 直接得到,由于网络故障可能导致备份机得到的备份信息不完整,在数据端设置暂存库记录为被备份机完全读取的数据变化痕迹,暂存库里的痕迹信息在被备份机读取后自动删除。

在这里,触发器完成两个工作:一是将数据变化事件发生前后的数据副本及修改者、修改时间、修改事件(INSERT、DELETE 或 UPDATE)和发出修改事件请求的客户端 IP 地址等信息写入本机的变化痕迹暂存库;二是通过扩展存储过程向备份机告知数据修改事件,通知备份机读取信息。

扩展存储过程是 DBMS 可调用在 SQL 服务进程内部执行而独立于 DBMS 的外部进程(DLL),本质上是一些自定义的 API 函数,用来实现与备份机的看守进程进行通信,使看守进程能够感知数据变化,并发出数据备份指令。

### 2.2 备份机

备份机设置一个痕迹备份库、一个看守进程和一个痕迹数据读取器。

痕迹备份库与数据源端的痕迹暂存库具有一样的属性结构,用来记录和保存数据被修改的痕迹和

历史数据,以备需要小范围的数据恢复时能从其中依次按痕迹翻拣出被修改过的数据,并对源数据库进行覆盖操作。

看守进程用来监控数据变化事件,并发布备份通知,自动启动痕迹数据读取器进程,从数据源端的痕迹暂存库中将本次和因网络故障未被备份的数据变化情况记录到痕迹备份库。由于数据源端的扩展存储过程和备份机的看守进程分别部署在不同的软硬件系统上,在系统运行时各有专用的进程空间,二者间传递消息必须以某种通讯机制实现,在基于触发器的双机备份模型中,采用命名管道方式,将扩展存储过程看作看守进程的数据提供者,通过对命名管道数据端的状态侦测和数据读取,感知到 DBMS 的数据变化。

痕迹数据读取器实际上是一个代理程序,接到备份通知后,完成暂存库到备份库的数据转移和销毁等操作。

## 3 结束语

本文提出一种基于触发器技术的 SQL Server 数据库实时备份方案。该方案使用双机备份方式,根据触发器自动执行数据备份的代码和关系数据库存储数据变化前后的状态检索数据变化痕迹,可以方便实现数据恢复,具有很强的实时性和直观性。该方案已用于广西大学的教师信息网站的敏感信息备份,备份库的痕迹信息不仅很快恢复误操作或蓄意破坏损毁的数据,而且为数据库的安全审计和监控提供了依据,在实践中取得很好的效果。

### 参考文献:

- [1] Rick Dobson. SQL Server2000 与 Visual Basic Net 编程 [M]. 北京:清华大学出版社,2003.
- [2] 万锋,龚正良,杨辛宝. MS SQL Server 数据库数据恢复的研究[J]. 计算机工程,2005,31(6):84-86.
- [3] 张景辉,侯建海,秦洋. 一种实现 SQL Server 数据库的备份和恢复的方法[J]. 微计算机应用,2005,26(2):228-230.
- [4] 周舟,黄景文. SQL Server 触发器在轨迹保留中的应用[J]. 计算机技术与发展,2007,17(7):207-209.

(责任编辑:邓大玉)