
SQL Server数据库压缩备份小记

Published on Apr 27, 2021 UTC by Wang Ye <<https://wangye.org>>

维护的一台服务器近期做了异地备份，具体的做法实际上也很简单，通过任务计划每日午夜将数据库备份文件压缩，再通过群晖的Cloud Station Backup回传至异地群晖NAS中，当然群晖NAS做了多盘RAID，这样数据也多了一层保护，本来数据库备份这里也考虑看能否使用第三方软件，结果在搜索时找到了微软官方的解决方案《[Schedule and automate backups of SQL Server databases in SQL Server Express](#)》，简单易用，比较nice~

这里再简单记录一下，作为一个备忘，以下操作仅供参考，建议大家先测试确保无误后再运用在生产服务器上，本人不对下面操作带来的可能数据损坏等问题负责，敬请理解。

1 备份数据库

1.1 安装必须的软件

这个是先决条件，一般服务器上可能缺少这些软件，如果不安装将无法使用下面的命令，当然你也可以使用替代命令，这里不再介绍，所要安装的软件列表如下，请依次下载，检查完整性，并按顺序安装：

[Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019](#)

[ODBC Driver for SQL Server](#)

[sqlcmd Utility](#)

1.2 创建备份存储过程

参考微软官方的脚本[SQL_Express_Backups.sql](#)，这里摘录如下：

```
-- Copyright ? Microsoft Corporation. All Rights Reserved.  
-- This code released under the terms of the  
-- Microsoft Public License (MS-PL, http://opensource.org/licenses/ms-  
© 2023 Wang Ye
```

```

pl.html.)
USE [master]
GO
/***** Object:  StoredProcedure [dbo].[sp_BackupDatabases] *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author: Microsoft
-- Create date: 2010-02-06
-- Description: Backup Databases for SQLExpress
-- Parameter1: databaseName
-- Parameter2: backupType F=full, D=differential, L=log
-- Parameter3: backup file location
-- =====
CREATE PROCEDURE [dbo].[sp_BackupDatabases]
    @databaseName sysname = null,
    @backupType CHAR(1),
    @backupLocation nvarchar(200)
AS
    SET NOCOUNT ON;
    DECLARE @DBs TABLE
    (
        ID int IDENTITY PRIMARY KEY,
        DBNAME nvarchar(500)
    )
    -- Pick out only databases which are online in case ALL
databases are chosen to be backed up
    -- If specific database is chosen to be backed up only
pick that out from @DBs
    INSERT INTO @DBs (DBNAME)
    SELECT Name FROM master.sys.databases
    where state=0
    AND name= ISNULL(@databaseName ,name)
    ORDER BY Name
    -- Filter out databases which do not need to backed up
    IF @backupType='F'

```

```

        BEGIN
            DELETE @DBs where DBNAME IN
('tempdb','Northwind','pubs','AdventureWorks')
            END
        ELSE IF @backupType='D'
            BEGIN
                DELETE @DBs where DBNAME IN
('tempdb','Northwind','pubs','master','AdventureWorks')
                END
            ELSE IF @backupType='L'
                BEGIN
                    DELETE @DBs where DBNAME IN
('tempdb','Northwind','pubs','master','AdventureWorks')
                    END
            ELSE
                BEGIN
                    RETURN
                END
            -- Declare variables
            DECLARE @BackupName nvarchar(100)
            DECLARE @BackupFile nvarchar(300)
            DECLARE @DBNAME nvarchar(300)
            DECLARE @sqlCommand NVARCHAR(1000)
            DECLARE @dateTime NVARCHAR(20)
            DECLARE @Loop int
            -- Loop through the databases one by one
            SELECT @Loop = min(ID) FROM @DBs
            WHILE @Loop IS NOT NULL
            BEGIN
-- Database Names have to be in [dbname] format since some have - or _
in their name
                SET @DBNAME = '['+(SELECT DBNAME FROM @DBs WHERE ID = @Loop)+']'
-- Set the current date and time n yyyyhhmmss format
                SET @dateTime = REPLACE(CONVERT(VARCHAR, GETDATE(),101), '/', '') +
                '_' + REPLACE(CONVERT(VARCHAR, GETDATE(),108), ':', '')
-- Create backup filename in path\filename.extension format for
full,diff and log backups
                IF @backupType = 'F'

```

```

        SET @BackupFile = @backupLocation+REPLACE(REPLACE(@DBNAME,
'[, '], '], '))+ '_FULL_' + @dateTime+ '.BAK'
    ELSE IF @backupType = 'D'
        SET @BackupFile = @backupLocation+REPLACE(REPLACE(@DBNAME,
'[, '], '], '))+ '_DIFF_' + @dateTime+ '.BAK'
    ELSE IF @backupType = 'L'
        SET @BackupFile = @backupLocation+REPLACE(REPLACE(@DBNAME,
'[, '], '], '))+ '_LOG_' + @dateTime+ '.TRN'
-- Provide the backup a name for storing in the media
    IF @backupType = 'F'
        SET @BackupName = REPLACE(REPLACE(@DBNAME, '[, '], '], '))
+' full backup for '+ @dateTime
    IF @backupType = 'D'
        SET @BackupName = REPLACE(REPLACE(@DBNAME, '[, '], '], '))
+' differential backup for '+ @dateTime
    IF @backupType = 'L'
        SET @BackupName = REPLACE(REPLACE(@DBNAME, '[, '], '], '))
+' log backup for '+ @dateTime
-- Generate the dynamic SQL command to be executed
    IF @backupType = 'F'
        BEGIN
            SET @sqlCommand = 'BACKUP DATABASE ' +@DBNAME+ ' TO
DISK = '''+@BackupFile+ ''' WITH INIT, NAME= ''' +@BackupName+''',
NOSKIP, NOFORMAT'
        END
    IF @backupType = 'D'
        BEGIN
            SET @sqlCommand = 'BACKUP DATABASE ' +@DBNAME+ ' TO
DISK = '''+@BackupFile+ ''' WITH DIFFERENTIAL, INIT, NAME= '''
+@BackupName+''', NOSKIP, NOFORMAT'
        END
    IF @backupType = 'L'
        BEGIN
            SET @sqlCommand = 'BACKUP LOG ' +@DBNAME+ ' TO DISK =
'''+@BackupFile+ ''' WITH INIT, NAME= ''' +@BackupName+''', NOSKIP,
NOFORMAT'
        END
-- Execute the generated SQL command

```

```
EXEC (@sqlCommand)
-- Goto the next database
SELECT @Loop = min(ID) FROM @DBs where ID>@Loop
END
```

选择master数据库，运行脚本，忘记选择也没事，脚本已经USE [master]，如果一切正常，那么在master数据库内就建立了备份存储过程 [dbo].[sp_BackupDatabases]。

1.3 执行备份命令

(1) 例子：使用Windows认证模式全局完整备份实例.\SQLEXPRESS所有数据库到D:\SQLBackups\路径下，这两内容请根据实际情况进行修改。

```
sqlcmd -S .\SQLEXPRESS -E -Q "EXEC sp_BackupDatabases
@backupLocation='D:\SQLBackups\' , @backupType='F'"
```

(2) 例子：使用用户名<YourSQLLogin>和密码<StrongPassword>登录并差异备份实例.\SQLEXPRESS所有数据库到D:\SQLBackups\路径下，这用户名、密码、实例名、备份路径内容请根据实际情况进行修改。

```
sqlcmd -U -P -S .\SQLEXPRESS -Q "EXEC
sp_BackupDatabases @backupLocation = 'D:\SQLBackups', @BackupType='D'"
```

还有其他用法，具体请参考微软[《Schedule and automate backups of SQL Server databases in SQL Server Express》](#)文档，这里不再赘述。

上面命令执行完毕后路径D:\SQLBackups\下将存放有备份的数据库文件，以BAK扩展名命名。我们可以在Windows 任务计划里周期性执行命令，以实现无人值守的数据库备份操作。

2 删除、压缩备份存档

2.1 删除旧的备份存档

随着备份命令的多次执行目的路径下备份文件将会越来越多，为了节约服务器空间我们可能需要手动删除旧的备份文件，是否可以通过命令自动删除旧的备份存档呢？

求助于万能的StackOverflow，我找到了这个方案[《Batch file to delete files older than N days》](#)，比如使用下面的命令可以删除D:\SQLBackups\路径下7天前的备份文件：

```
forfiles /p "D:\SQLBackups" /s /m *.BAK /D -7 /C "cmd /c del @path"
```

这里的天数设置在/D -命令后，比如7天就是/D -7，注意数字前面的短横线，据原作者介绍这个命令适用于Windows 7及以后版本（包括Windows 10）。

2.2 压缩备份存档

实际操作下来，这种备份模式对于远程网络传输并不是很友好，部门业务数据库的备份文件比较大，有的能有十几GB或者更多，需要传输很长时间，严重占用服务器带宽，所以对于远程备份文档传输来说，需要在传输前先压缩，实际测试下来压缩比还是相当可观的，使用[7zip](#)这个小巧开源的压缩解压，7zip有命令行版本，位于安装路径下7z.exe，比如通过下面的命令压缩D:\SQLBackups\路径下所有的BAK文件到backups.7z：

```
7z.exe a -t7z backups.7z D:\SQLBackups\*.BAK
```

这样所有的数据库备份将会被合并压缩成一个backups.7z，如果数据库较多，那么这个文件的体积也会很大，不利于后期管理，后来我的解决办法的每个BAK单独压缩，参考2.1的命令写法，我们可以尝试下面的命令：

```
forfiles /p "D:\SQLBackups" /s /m *.BAK /C "cmd /c C:\PROGRA~1\7-Zip\7z.exe a -t7z D:\SQLBackups\@file.7z @path"
```

这样在D:\SQLBackups\路径下将会生存7z扩展名的压缩文件，通过网络传输此压缩文件到备份服务器上大大提高效率，这里C:\PROGRA~1\7-Zip\7z.exe根据你的7zip安装路径可能需要修改。

3 备份、压缩、删除旧备份合并操作

参考1、2节所叙述内容，整理一下自己的服务器SQL Server数据库备份方案，分为6步：1. 设立临时目录和同步目录；2. 将同步目录链接到同步备份软件上（群晖 Cloud Station Backup）；3. 删除同步目录下旧存档；4. 备份数据库到临时目录下；5. 压缩数据库并将压缩文件移入同步目录下；6. 同步软件同步；7. 删除临时目录下备份文件。

比如我们服务器上临时目录为D:\SQLBackups\Temp，同步目录为D:\SQLBackups\Sync，其中同步软件只管D:\SQLBackups\Sync路径下文件的备份，通过定期运行下面的批处理命令即可：

```
@ECHO OFF
forfiles /p "D:\SQLBackups\Sync" /s /m *.* /D -7 /C "cmd /c del @path"
sqlcmd -S .\SQLEXPRESS -E -Q "EXEC sp_BackupDatabases
@backupLocation='D:\SQLBackups\Temp\' , @backupType='F'"
forfiles /p "D:\SQLBackups\Temp" /s /m *.BAK /C "cmd /c C:\PROGRA~1\7-
Zip\7z.exe a -t7z D:\SQLBackups\Sync\@file.7z @path"
DEL /F /S /Q D:\BackupTemp\sqlserver\*.*
```

上述命令还请读者根据需要修改，这里就先介绍这么多。

External References (Links)

《Schedule and automate backups of SQL Server databases in SQL Server Express》

<https://docs.microsoft.com/en-us/troubleshoot/sql/admin/schedule-automate-backup-database>

Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019

<https://support.microsoft.com/en-us/topic/the-latest-supported-visual-c-downloads-2647da03-1eea-4433-9aff-95f26a218cc0>

ODBC Driver for SQL Server

<https://docs.microsoft.com/en-us/sql/connect/odbc/download-odbc-driver->

[for-sql-server?view=sql-server-ver15](#)

sqlcmd Utility

<https://docs.microsoft.com/en-us/sql/tools/sqlcmd-utility?view=sql-server-ver15>

SQL_Express_Backups.sql

https://raw.githubusercontent.com/microsoft/mssql-support/master/sample-scripts/backup_restore/SQL_Express_Backups.sql

《Batch file to delete files older than N days》

<https://stackoverflow.com/questions/51054/batch-file-to-delete-files-older-than-n-days>

7zip

<https://www.7-zip.org/>