

Using the FORFILES Command to Delete SQL Server Backups

Written By: Tim Ford -- 11/3/2008

Problem

Recently I constructed a new backup process that I want to institute globally across all my SQL Server 2005 instances. This backup process will not only backup all the databases I point it towards, but at the same time will script out the backup commands to a single file in the format of F_YYYYMMDD.sql if the backup process is a full database backup or D\T_YYMMDD_HHMMSS.sql if the backup process is a differential (D) or transaction log (T) backup. These script files are then stored in a subfolder under the backup directory on the SQL Server. The process works great, but I only don't want to keep every .sql file. I can't see the need to keep these scripts after 30 days. Is there a way I can automate a file deletion process from within SQL Server?

Solution

While there are many ways the file deletion process can be handled with T-SQL code. I use the xp_cmdshell command along with the FORFILES command for a very similar process to what you have outlined. Solid information of FORFILES is available from [Microsoft TechNet](#), but I will touch on much of the structure and use of FORFILES for your purposes in this tip.

The FORFILES command will select a subset of files and execute a command against the set. The command requires the following parameters and accepts the following variables:

Parameters

Parameter Name	Description
/p	Path
/m	Search Mask (default is *.*)
/s	Subdirectories will be searched recursively if this parameter is included
/c <command>	Command to be executed against each file in the result set, commands must be enclosed in double-quotes, default is "cmd c/ echo @file"
/d	Date range for file selection, using Last Modified Date as the criterion for the file. When the /d parameter is in the form of MM/DD/YYYY, file meeting the criteria of +/- the specified date are included. When in the format of a smallint (-32,768 - 32,768) the files +/- the files with a modified date +/- that number of days from the current date are included in the file result set.

Variables

Variable Name	Description
@FILE	File name
@FNAME	File name without extension
@EXT	File extension
@PATH	Full path of the file
@RELPATH	Relative path of the file
@ISDIR	Evaluates as TRUE if the file type is a directory

@FSIZE	File size in bytes
@FDATE	Last modified date stamp on the file
@FTIME	Last modified timestamp on the file

Using these parameters the following examples could be constructed to take care of your dilemma for deleting your backup script files. You can create scripts based upon modification date/time or backup type. You can even construct scripts that utilize both criteria. We will now take a closer look at these potential scripts. Remember that you will be executing these from within T-SQL code, so you will need to wrap the statements within an `xp_cmdshell` call in the format of `EXEC xp_cmdshell 'FORFILES COMMAND'`. Please note that in all examples I am using the `/Q` and `/F` flags for the `del` command. These signify that the command will use quiet mode (`/Q`) and will even delete read-only files (`/F`).

Examples

Delete all `.sql` files in the `C:\Backup` directory and its subfolders where the file modified date is older than 10/18/2008.

```
EXEC xp_cmdshell 'FORFILES /p c:\BACKUP /s /m *.sql /d 10/18/2008 /c "CMD /C del /Q /F @FILE"'
```

Delete all `.sql` files in the `C:\Backup` directory and its subfolders where the file modified date is more than 30 days old.

```
EXEC xp_cmdshell 'FORFILES /p c:\BACKUP /s /m *.sql /d -30 /c "CMD /C del /Q /F @FILE"'
```

Delete all `.sql` files in the `C:\Backup` directory and its subfolders where the file modified date is more than 30 days old and the file name starts with an "F_".

```
EXEC xp_cmdshell 'FORFILES /p c:\BACKUP /s /m F_*.sql /d -30 /c "CMD /C del /Q /F @FILE"'
```

Next Steps

- Modify the code above to fit your retention policy, file structure and needs.
- Add the modified code to a subsequent job step in the SQL Agent job that runs your backup process. Include [verbose SQL Agent job logging](#) to verify that the process is working correctly.
- Keep in mind that the `FORFILES` command does a lot more than just deleting files. As an example, it can be used to list files for other processes as well.
- Review [other file deletion automation tips](#) from MSSQLTips.com.
- Review tips on [xp_cmdshell](#) at MSSQLTips.com

Copyright (c) 2006-2008 [Edgewood Solutions, LLC](#) All rights reserved
[privacy statement](#) | [disclaimer](#) | [copyright](#)

Some names and products listed are the registered trademarks of their respective owners.