

[Activate your FREE membership today](#) | [Log-in](#)TechTarget Windows Media
> WEB SITES

ADVERTISEMENT

Microsoft SQL Server 2005 delivers 60,359 QphH for the 3-terabyte TPC-H benchmark.

SEE FOR YOURSELF >>

Microsoft SQL Server 2005

HOME | NEWS | TOPICS | ITKNOWLEDGE EXCHANGE | TIPS | ASK THE EXPERTS | WEBCASTS | WHITE PAPERS | WINDOWS IT DOWNLOADS

SEARCH this site and the web SEARCH ADVANCED SEARCH | SITE INDEX

Search Powered by Google



[Microsoft Dynamics: Implement ERP Software Everyone Gets and Reduce Your Support Time.](#)

[Home](#) > [SQL Server Tips](#) > [Stored Procedures](#) > Examples of SQL Server stored procedures and parameters

[EMAIL THIS](#)

SQL Server Tips:

TIPS & NEWSLETTERS TOPICS

STORED PROCEDURES

Examples of SQL Server stored procedures and parameters

Denny Cherry

09.06.2007

Rating: -4.67- (out of 5)

RSS FEEDS: [Expert advice on database development](#)



Microsoft included several hundred stored procedures in the various versions of Microsoft SQL Server and it has documented a good percentage of them. But many stored procedures remain undocumented. Some are used within the Enterprise Manager GUI in SQL 2000 and were not intended to be used by other processes. Microsoft has slated some of these stored procedures to be removed (or they have been removed) from future versions of SQL Server. While these stored procedures can be very useful and save you lots of time, they can be changed at any time in their function or they can simply be removed.

The chart below shows that while many of the procedures have been carried through from one version of Microsoft SQL Server to another, new stored procedures have been introduced, and some have been removed from the install package. Most, if not all, of the procedures require the user to be a member of the sysadmin fixed server role in order to execute the procedures. The stored procedures that interact with the file system also require that the user executing the procedure (as well as SQL Server's service account) have access to the file/folder.

Procedure Name	SQL 2000	SQL 2005	SQL 2008
sp_executesql	X		
sp_MSforeachdb	X	X	X
sp_MSforeachtable	X	X	X
sp_readerrorlog	X	X	X
xp_create_subdir		X	X
Xp_delete_file		X	X
xp_dirtree	X	X	X
xp_fileexist	X	X	X
xp_fixdrives	X	X	X
xp_getfiledetails	X		
xp_getnetname	X	X	X
xp_loginconfig	X	X	X
xp_makecab	X		
xp_msver	X	X	X
xp_get_mapi_profiles	X	X	X

xp_subdirs	X	X	X
xp_test_mapi_profile	X	X	X
xp_unpackcab	X		

sp_executesql

Microsoft removed this handy little procedure called `sp_executesql` from SQL Server in SQL Server 2005. It allows you to generate dynamic SQL code on the fly by using a `SELECT` query. Then, the resulting SQL commands will be executed against the database. It permits you to create a single piece of code that can, in a single step, find the number of records in every table in your database (as the example shows). This is an undocumented stored procedure and there is no way of knowing why it was removed. But, alas, this handy utility is gone.

```
exec sp_executesql 'SELECT "SELECT "" + name + """,
count(*) FROM " + name
from sysobjects
where xtype = "U"'
```

sp_MSforeachdb / sp_MSforeachtable

Two procedures, `sp_MSforeachdb` and `sp_MSforeachtable`, are wrappers around a cursor. They allow you to execute T-SQL code against each database on your SQL Server and each table within the current database, respectively. You cannot, however, use an `sp_MSforeachtable` command within an `sp_MSforeachdb` command in SQL 2000 and prior. The cursor name that was used within those procedures was the same (`hCForEach`) and would therefore return an error saying that the cursor name was already in use for each execution of the `sp_MSforeachtable`. In SQL Server 2005, Microsoft resolved this issue. In order to "next" the command, you must tell one of the procedures it will be using a different replacement character other than the default question mark. I change the replacement character in the database command because it's easier.

Print each table name in the current database.

```
exec sp_MSforeachtable 'print "?"'
```

Print each database on the current server.

```
exec sp_MSforeachdb 'print "?"'
```

Print each table on the current server.

```
exec sp_MSforeachdb 'use [@] exec sp_MSforeachtable "print
""@.?"", '@'
```

sp_readerrorlog / xp_readerrorlog

The stored procedure `sp_readerrorlog` actually comes in two forms. Each works the same; one is simply a wrapper for the second. The wrapper stored procedure is `sp_readerrorlog` and it calls `xp_readerrorlog`. Both have four input parameters, but only the first two are useful to us. The first parameter establishes the file number that you wish to view. The second is the log to view (1 or null for `ERRORLOG`, 2 for `SQL Agent Log`). This allows you to view your error logs quickly and easily instead of having to look at the bloated log viewer that now comes with SQL Server 2005 and SQL 2008.

View the current SQL `ERRORLOG` file.

```
exec sp_readerrorlog
```

```
exec sp_readerrorlog 0, 1
```

View the Prior SQL Agent Log file.

```
exec sp_readerrorlog 1, 2
```

sp_create_subdir

Introduced in SQL Server 2005, the `sp_create_subdir` stored procedure is very

REFERENCE DESK

Stored procedures

NEWS, TIPS & MORE

- [Problem attaching stored procedure and ...](#) (EXPERT ANSWER)
 - [Schedule stored procedure and save output ...](#) (EXPERT ANSWER)
 - [Copy stored procedure between SQL Server ...](#) (EXPERT ANSWER)
 - [BULK insert to import data from file to ...](#) (EXPERT ANSWER)
- [VIEW MORE](#)

VENDOR CONTENT

- [Innovate Faster with Oracle Database 11g](#) (WHITE PAPER)

Microsoft
POWER YOU CAN'T MISS.
Microsoft SQL Server 2005
delivers 60,359 QphH
for the 3-terabyte TPC-H benchmark.
SEE FOR YOURSELF >>
Microsoft
SQL Server 2005

- [Oracle Database 11g for Data Warehousing and Business Intelligence](#) (WHITE PAPER)
- [Ten hacker tricks to exploit SQL Server systems](#) (PODCAST)
- [DuPont case highlights insider threat](#) (PODCAST)

SEE ALSO

- **Related Topics:**
[Stored procedures](#), [.NET development](#), [C#](#), [Distributed Management Objects \(DMO\)](#), [SQL/Transact SQL \(T-SQL\)](#)
- **Site Highlights:**
[Availability/Scalability](#)
[Backup and Recovery](#)

GET E-MAIL UPDATES

Submit your e-mail below to receive SQL Server-related news, tech tips and more, delivered to your inbox.

Development

E-mail:

Not a member? We'll activate your FREE membership with your subscription.

handy because you can use it to create folders on SQL Server's hard drive or on a network share from within T-SQL.

```
exec xp_create_subdir 'c:\MSSQL\Data'
```

xp_delete_file

Use the `xp_delete_file` stored procedure introduced in SQL Server 2005 to delete files from SQL Server's hard drive or a network share from within T-SQL.

xp_dirtree

The `xp_dirtree` procedure allows you to view the folder tree and/or file list beneath a folder. This procedure has several parameters that control how deep the procedure searches and whether it returns files and folders or folders only. The first parameter establishes the folder to look in. (Recommendation: Do not run this procedure against the root of the drive that Windows is installed on because it will take some time to generate the tree and return the data.) The second parameter limits the number of recursive levels that the procedure will dig through. The default is zero or all levels. The third parameter tells the procedure to include files. The default is zero or folders only, a value of 1 includes files in the result set. Specifying a third value not equal to zero will add an additional column to the output called `file` which is a bit field showing the entry in a folder or file.

Get the full directory tree.

```
exec xp_dirtree 'd:\mssql\'
```

Get the first two levels of the directory tree.

```
exec xp_dirtree 'd:\mssql\', 2
```

Get the first three levels of the directory tree, including files.

```
exec xp_dirtree 'd:\mssql\', 3, 1
```

xp_fileexist

This SQL Server stored procedure, `xp_fileexist`, is used to determine if a file exists on SQL Server's hard drive or on a network share. It is extremely useful in stored procedures that load data from flat files. It allows you to check and see if the file exists before attempting to blindly load the file. The procedure has two parameters. Use the first parameter to determine if the file or folder you want exists. The second is an output parameter, which when specified, returns a 1 or 0 if the file exists or does not.

Without the parameter.

```
exec xp_fileexist 'c:\importfile.csv'
```

With the parameter.

```
DECLARE @file_exists int
exec xp_fileexist 'c:\importfile.csv', @file_exists OUTPUT
SELECT @file_exists
```

xp_fixeddrives

The procedure `xp_fixeddrives` is one of the most useful procedures. It presents a list of all drive letters and the amount of free space each drive has. The parameter has a single optional input parameter that can filter the results by drive type. A value of 3 will return all mass storage devices (CD-ROM, DVD, etc.); a value of 4 will return the hard drives; while a value of 2 will return removable media (USB thumb drives, flash drives, etc.).

Return all drives.

```
exec xp_fixeddrives
```

Return hard drives only.

```
exec xp_fixdrives 2
```

xp_getfiledetails

The procedure xp_getfiledetails is another extremely useful procedure, which was last available in SQL Server 2000. This procedure returns size, date and attribute information about the file specified, including date and times created, accessed and modified.

```
exec xp_getfiledetails 'c:\fileload.csv'
```

xp_getnetname

The procedure xp_getnetname returns the name of the physical machine where Microsoft SQL Server is installed. You can have the machine name returned as a record set or as a variable.

Without the parameter.

```
exec xp_getnetname
```

Using the parameter.

```
DECLARE @machinename sysname
exec xp_getnetname @machinename OUTPUT
select @machinename
```

xp_loginconfig

This SQL Server stored procedure will tell you some basic authentication information about the user executing it. It tells you the authentication method (Windows versus SQL Login), the default domain of the server, the audit level, as well as some internal separator information.

```
exec xp_loginconfig
```

xp_makecab

Back in SQL Server 2000, Microsoft gave us the ability to compress OS files directly from T-SQL without having to shell out to DOS via xp_cmdshell and run third-party software, like pkzip or winzip. That command was xp_makecab. It allows you to specify a list of files you want to compress as well as the cab file you want to put

Visit our SQL Server IT Knowledge Exchange:

- [Denny Cherry's blog](#)
- [Questions from your peers](#)

them in. It even lets you select default compression, MSZIP compression (akin to the .zip file format) or no compression. The first parameter gives the path to the cab file in which you want to create or add files to. The second parameter is the compression level. The third parameter applies if you want to use verbose logging. Starting with the fourth parameter and on down are the names of the files you want to compress. In my testing, I was able to pass 45 file names to be compressed to the extended stored procedure, which means that it is a very flexible solution to your data compression requirements.

```
exec xp_makecab 'c:\test.cab', 'mszip', 1, 'c:\test.txt', 'c:\test1.txt'
```

xp_msver

The procedure xp_msver is very useful when looking for system information. It returns a wealth of information about the host operating system -- the SQL version number, language, CPU type, copyright and trademark information, Microsoft Windows version, CPU count and affinity settings, physical memory settings and your product key. This procedure has many input parameters that allow you to filter down the records that are returned. Each parameter is a sysname data type, which accepts the name of one of the records. If any parameters are specified, only the rows specified as a parameter are returned.

No filter specified.

```
exec xp_msver
```

Return only Platform and Comments records.

```
exec xp_msver 'Platform', 'Comments'
```

xp_get_mapi_profiles

The `xp_get_mapi_profiles` procedure assists you in configuring SQL Mail. When executed, it will call to Windows via the SQL Mail component of SQL Server and display a list of available MAPI profiles that are configured in Outlook and it specifies which profile is the default profile. If it doesn't display any records, then either Outlook is not configured correctly or SQL Server is not running under a domain account with Outlook profiles configured. In order to use this procedure in SQL Server 2005 or SQL Server 2008, you must enable the "SQL Mail XPs" option in the Surface Area Configuration tool or within the `sp_configure` procedure.

```
exec xp_get_mapi_profiles
```

xp_subdirs

The `xp_subdirs` procedure displays a subset of the information available through `xp_dirtree`. `xp_subdirs` will display all the subfolders in a given folder. It can be very handy when you are building a directory tree within a table dynamically and you do not want to worry about the extra parameters of the `xp_dirtree` procedure.

```
exec xp_subdirs 'd:\mssql'
```

xp_test_mapi_profiles

The procedure `xp_test_mapi_profiles` is another undocumented stored procedure that is very useful when you are setting up SQL Mail. It will start, then stop, a MAPI session to ensure that MAPI is configured correctly and working within the confines of Microsoft SQL Server. I should note that it does not verify the mail server configuration within the MAPI client (Outlook) nor does it send a test message.

The procedure accepts a single input parameter. That parameter is the name of the MAPI profile you wish to test. Like the `xp_get_mapi_profiles` procedure, for this

More on SQL Server stored procedures:

- [Stored procedure: A simple way to perform crosstab operations](#)
- [Q&A: Writing stored procedures in SQL Server 2005](#)

stored procedure to function in SQL Server 2005 and SQL Server 2008, you must enable the "SQL Mail XPs" option in the Surface Area Configuration tool or within the `sp_configure` procedure.

When working with the SQL Mail stored procedures, be aware that SQL Mail is still slated for removal from the Microsoft SQL Server platform. That means the procedures

`sp_get_mapi_profiles` and `xp_test_mapi_profiles` are slated for removal, as they are part of the SQL Mail subsystem. You should do all mail work on SQL Server 2005 and later using Database Mail instead of SQL Mail to ensure code portability with future versions of SQL Server. Microsoft initially slated SQL Mail for removal in SQL Server 2008, however, based on its inclusion in the current beta release, its future in SQL Server 2008 is unknown.

xp_unpackcab

Along with the `xp_makecab` procedure comes the `xp_unpackcab` extended stored procedure, and it does just what it says: It extracts files from cab files. The first parameter is the cab file, the second is the path you want to extract to and the third is verbose logging. A fourth parameter lets you specify the "extract to" file name.

```
exec xp_unpackcab 'c:\test.cab', 'c:\temp', 1
```

While this is not intended to be a complete list of the undocumented stored procedures in SQL Server, it does provide a reference point for many of these procedures with the hope of making the lives of the SQL Server administrators easier. Remember, you should never count on these procedures surviving from one SQL Server version to the next, nor should you expect their code base to remain the same between versions. That said, go code and enjoy.

All information provided about Microsoft SQL Server 2008 (Katmai) is based on beta edition 10.0.1019 of the software and is subject to change without notice.

ABOUT THE AUTHOR:

Denny Cherry is a DBA and database architect managing one of the largest SQL Server installations in the world, supporting more than 175 million users. Denny's primary areas of expertise are system architecture, performance tuning, replication and troubleshooting.
Copyright 2007 [TechTarget](#)

Rate this Tip

To rate tips, you must be a member of searchSQLServer.com.

[Register now](#) to start rating these tips. [Log in](#) if you are already a member.

Share - [Digg This!](#) [Bookmark with Del.icio.us](#)

 **SQL SERVER RELATED LINKS****Ads by Google****[Complex Queries?](#)**

Stop throwing hardware at your Data Warehouse!
www.infobright.com

[Dell™ SQL Servers](#)

Enterprise Data Mgmt Solutions From Dell™. Official Site.
www.Dell.com

[Generate Stored Procedure](#)

Integrated with SSMS 2005 You decide what to generate
www.etekglobalinc.com

[SQL Performance Tuning](#)

5 Key Areas You Must Focus on to Improve Performance & Availability.
www.heroix.com

[Sql Server Tuning](#)

Manage your IT infrastructure from one console—Download trial today.
www.Microsoft.com/Essentials

 **RELATED CONTENT****■ Stored procedures**

- [BULK insert to import data from file to SQL Server database](#)
- [Tricks to increase SQL Server query performance](#)
- [Stored procedure to find fragmented indexes in SQL Server](#)
- [Use table-valued parameters for SPs in SQL Server 2008](#)
- [How to insert into SQL Server system tables](#)
- [Creating SQL Server tables and columns: Quick tips to know](#)
- [Creating SQL Server tables: A best practices guide](#)
- [Just installed SQL Server 2005 Express can't find database](#)
- [Calling an extended stored procedure in SQL Server 2005](#)
- [Executing a trigger before SQL Server table is updated](#)

■ SQL Server overview

- [Optimize T-SQL data types in SQL Server](#)
- [Update SQL Server table statistics for performance kick](#)
- [SQL Server blob data types for accessing and storing data](#)
- [Synchronize databases on two separate active servers](#)
- [SQL Server Blog Watch](#)
- [SQL Server backup and restore school](#)
- [Temporary tables in SQL Server vs. table variables](#)
- [FAQ: Creating and altering SQL Server tables](#)
- [SQL Server consolidation: Why, when and how to consolidate SQL Servers](#)
- [Viewing database tables in Query Analyzer](#)
- [SQL Server overview Research](#)

■ Stored Procedures

- [Stored procedure to find fragmented indexes in SQL Server](#)
- [Use table-valued parameters for SPs in SQL Server 2008](#)
- [Top 10 SQL Server development questions](#)

[FAQ: SQL Server stored procedure how-tos](#)
[Q&A: Writing stored procedures in SQL Server 2005](#)
[Stored procedures in SQL Server: A dozen must-have tips](#)
[Calling a stored procedure remotely](#)
[Delete .bak files automatically with CLR](#)
[Basic Transact-SQL programming constructs: 15 tips, 15 minutes](#)
[T-SQL identifiers](#)

i RELATED GLOSSARY TERMS

Terms from [Whatis.com](#) – the [technology online dictionary](#)

- [library](#) (SearchSQLServer.com)
- [trigger](#) (SearchSQLServer.com)

i RELATED RESOURCES

- [2020software.com](#), trial software downloads for [accounting software](#), [ERP software](#), [CRM software and business software](#) systems
- Search [Bitpipe.com](#) for the latest [white papers](#) and [business webcasts](#)
- [Whatis.com](#), the online [computer dictionary](#)

DISCLAIMER: Our Tips Exchange is a forum for you to share technical advice and expertise with your peers and to learn from other enterprise IT professionals. TechTarget provides the infrastructure to facilitate this sharing of information. However, we cannot guarantee the accuracy or validity of the material submitted. You agree that your use of the Ask The Expert services and your reliance on any questions, answers, information or other materials received through this Web site is at your own risk.

[HOME](#) | [NEWS](#) | [TOPICS](#) | [ITKNOWLEDGE EXCHANGE](#) | [TIPS](#) | [ASK THE EXPERTS](#) | [WEBCASTS](#) | [WHITE PAPERS](#) | [WINDOWS IT DOWNLOADS](#)

[About Us](#) | [Contact Us](#) | [For Advertisers](#) | [For Business Partners](#) | [Site Index](#) | [RSS](#)

SEARCH

TechTarget provides enterprise IT professionals with the information they need to perform their jobs - from developing strategy, to making cost-effective IT purchase decisions and managing their organizations' IT projects - with its network of [technology-specific Web sites, events and magazines](#).

[TechTarget Corporate Web Site](#) | [Media Kits](#) | [Reprints](#) | [Site Map](#)

All Rights Reserved, [Copyright 2005 - 2007](#), TechTarget | [Read our Privacy Policy](#)

