

CSS SQL Server Engineers

This is the official team Web Log for Microsoft Customer Service and Support (CSS) SQL Support. Posts are provided by the CSS SQL Escalation Services

Sparse File Errors: 1450 or 665 due to file fragmentation: Fixes and Workarounds

psssql

3 Mar 2009 11:30 PM

1

You might be familiar with the sparse file problem that Bob Dorr has blogged about in the past. <http://blogs.msdn.com/psssql/archive/2008/07/10/sql-server-reports-operating-system-error-1450-or-1452-or-665-retries.aspx>. We wanted to update you with the work we have been involved in the past few months. There are several fixes that we plan to release or already released to address the different aspects of this problem. The list provided below summarizes all of the documentation and fix available for you regarding this problem. We will keep this list updated as we come across more information or changes.

Product & Version	Release Vehicle	KB Article #	Title	Description of problem and fix
Windows Server 2003, 2008 Windows Vista		967351	A heavily fragmented file in an NTFS volume may not grow beyond a certain size	The article explains the root cause of this problem and the technical details of the limitation.
Windows Server 2008 RTM Windows Vista SP1	Regular QFE	957065	Error status message from the ntfs.sys driver when some applications update very large files in Windows Server 2008 or in Windows Vista systems: "0xc0000427 STATUS_FILE_SYSTEM_LIMITATION"	Windows Vista RTM and Win 20008 RTM had a problem that reduces the attribute limits to a much smaller number than what Windows 2003 had. This hotfix increases the attribute limitation in Windows 2008 and Vista OS and brings it on par with the Windows 2003 limit. If you are running Windows Vista or Windows 2008 and you encounter this problem, the first step is to apply this fix.
SQL Server 2005	SP2 CU12	961123	FIX: The SQL Server service stops when you run one of the DBCC CHECK commands on a SQL Server 2005 database or when you create a database snapshot for a SQL Server 2005 database	When running DBCC CHECKDB command and if you run into this limitation, you could encounter Out Of Memory errors in SQL Server. This is because the children threads associated with the DBCC CHECKDB command starts accumulating error messages relating to the internal snapshot failures and quickly consumes all the available memory configured for the SQL Server instance. Under such conditions, you will notice that MEMORYCLERK_SQLGENERAL consuming a big portion of the SQL Server memory. Applying this fix will make sure that such memory consumption will not happen.
SQL Server 2005	SP3 CU1			
SQL Server 2008	RTM			
SQL Server 2005	SP3 CU2 SP2 CU14	967164	FIX: The SQL Server service stops responding, and you receive error 1450 when you create database snapshot or run a DBCC CHECK statement on a database in SQL Server 2005	In Windows 2003, when we hit the STATUS_INSUFFICIENT_RESOURCES condition, we will retry the write operation indefinitely. This can result in the DML operations which need to perform Copy-On-Write end up in this loop for a long time. You notice the

				<p>following error message repeatedly in the SQL Server Error log:</p> <p>The operating system returned error 1450(Insufficient system resources exist to complete the requested service.) to SQL Server during a write at offset 0x0000159d8ae000 in file with handle 0x00000000000000AAC.</p> <p>After you apply this fix, we will avoid the retry for the write to a sparse file for this specific error condition. As a result, the DBCC command stops immediately when the first error 1450 is encountered. Also snapshot database is marked as suspect when the first error 1450 is encountered for a write to the snapshot file.</p>
SQL Server 2008	RTM CU4 SP1 CU1			
SQL Server 2005	SP3 CU3 SP2 CU14	<u>967470</u>	Fix to avoid aggressively pushing pages to the snapshot	When we perform an update or delete to a heap that modifies a small number of rows, SQL Server performs a scan of the heap. This scan could trigger a copy-on-write to the snapshot of several additional pages than what the update or delete needs to modify. This can happen when the update or delete is performed using a predicate condition. Because of this problem we will end up with copy-on-write of both dirty and non-dirty pages. The fix is to perform copy-on-write for pages which are actually modified by the update or delete.
SQL Server 2008	RTM CU4 SP1 RTM			

Note:

Refer to the following articles on how to get the latest Cumulative Updates:

The [SQL Server 2008 builds](#) that were released after SQL Server 2008 was released

The [SQL Server 2008 builds](#) that were released after SQL Server 2008 Service Pack 1 was released

The [SQL Server 2005 builds](#) that were released after SQL Server 2005 Service Pack 3 was released

The [SQL Server 2005 builds](#) that were released after SQL Server 2005 Service Pack 2 was released

The [list of bugs](#) that are fixed in SQL Server 2008 Service Pack 1

Even with all these fixes, it is possible that you still hit the limitation due to the nature of the workload or physical layout and design of your databases. Below you will find a compilation of workarounds we have been discussing with customers and provides relief.

If this problem happens with the internal snapshot created by DBCC CHECKDB family of commands

You can schedule DBCC CHECKDB during a time window when there will be comparatively less data modifications and hence less Copy-On-Writes to the snapshot sparse files

Use WITH TABLOCK option of DBCC CHECKDB to avoid using the internal snapshot creation.

You can evaluate alternatives to DBCC CHECKDB command. You can execute DBCC CHECKTABLE in a loop for all tables followed by DBCC CHECKALLOC.

Evaluate if you can create an explicit snapshot on a different drive which has very less activity and execute DBCC CHECKDB against that snapshot database.

Avoid running multiple DBCC CHECKDB commands concurrently on different databases. This will reduce the probability of fragmentation.

Avoid performing large scale maintenance work [like re-indexing or shrink operations] when DBCC CHECKDB is still running.

Create a explicit snapshot of the source database and execute DBCC CHECKDB against the snapshot database

Evaluate using various options of DBCC CHECKDB like PHYSICAL_ONLY to reduce the lifetime of the internal snapshot

created by the DBCC command.

Evaluate using Trace Flag 2566 to avoid expensive data purity checks. Refer to a known issue: KBA 945770
<http://support.microsoft.com/default.aspx?scid=kb;EN-US;945770>

If this problem happens with the explicit snapshot created on a database for reporting purposes

Reduce the lifetime of the snapshot sparse files. The snapshots have to be dropped and re-created more frequently before they start to encounter these error messages.

Create the snapshot on a different drives other than the source data files and with less file activity to reduce file level fragmentation.

Drop the snapshot database before doing extensive maintenance work or large data load operations. Re-create the snapshots after the maintenance work is completed.

Applicable to both situations

Find out which tables/indexes result in the most write activity during the lifetime of the snapshot and separate them out into a different file group with many files of comparatively smaller sizes.

Make sure the data files are not fragmented physically to a great extent. Use the Operating System Disk management tools to understand the physical file system fragmentation of the data files and reduce the fragmentation.

Specify a lower FILLFACTOR to reduce the frequency of page splits and resulting Copy-On-Writes

Reduce the size of the data files. Create more data files which are less than 64 GB in size approximately and spread the data across these files in a separate file group.

Note: Refer to the SQL Server Books Online for the applicable locks held by DBCC CHECKDB when the different options are specified for the command. Also depending on the options chosen different levels of checks are performed by the DBCC CHECKDB command. Consider all of this information before you decide to use these options on your production server.

Thanks

Suresh B. Kandoth

SQL Server Senior Escalation Engineer

Tweet

7

Like

2

Share

1

Save this on Delicious

Comments

25 Nov 2013 4:56 PM



Wes

I have Windows 2008 R2 x64 in a DW environment with 200+ gb of ram, and after a integrity check maintenance plan (DBCC CHECKDB) runs, I get : DESCRIPTION: The operating system returned error 665(The requested operation could not be completed due to a file system limitation) to SQL Server during a write at offset 0x00002a6a490000 in file 'L:\MSSQL11.MSSQLSERVER\MSSQL\Data\mydb.mdf:MSSQL_DBCC10'. Additional messages in the SQL Server error log and system event log may provide more detail. This is a severe system-level error condition that threatens database integrity and must be corrected immediately. Complete a full database consistency check (DBCC CHECKDB). This error can be caused by many factors; for more information, see SQL Server Books Online.



Recent Posts

[How It Works: Behavior of a 1 Trillion Row Index Build \(Gather Streams from SORT\)](#)

Posted 7 days ago by [psssql](#)

[How to grab multiple parent/child elements from XML Data Source](#)

Posted 7 days ago by [Adam W. Saxton](#)

[SharePoint Adventures : Using Claims with Reporting Services](#)

Posted 8 days ago by [Adam W. Saxton](#)

[Version 9.04.0013 of the RML Utilities for x86 and x64 has been released to the download center](#)

Posted 12 days ago by [psssql](#)

[RS, SharePoint and Forefront UAG Series – Intro](#)

Posted 13 days ago by [Adam W. Saxton](#)