



## Diskeeper Can Boost Your SQL Server's Performance

*Software Spotlight by Brad M. McGehee*

One of the biggest hardware bottlenecks of any SQL Server is disk I/O. And anything that we, as DBAs, can do to reduce SQL Server's use of disk I/O will help boost its performance. Some of the most common things DBAs do to reduce disk I/O bottlenecks include:

- Tuning queries to minimize the amount of data returned.
- Using fast disks and arrays.
- Using lots of RAM, so more data is cached.
- Frequent DBCC REINDEXing of data to remove logical database fragmentation.

Another less frequently used method to reduce overall disk I/O, but nonetheless important, is to perform physical defragmentation of SQL Server program files, database files, transaction logs, and backup files. Physical file fragmentation occurs in two different ways. First, individual files are broken into multiple pieces and scattered about a disk or an array (they are not contiguous on the disk). Second, free space on the disk or array consists of little pieces that are scattered about, instead of existing as fewer, larger free spaces. The first condition requires a disk's head to make more physical moves to locate the physical pieces of the file than contiguous physical files. The more physically fragmented a file, the more work the disk drive has to do, and disk I/O performance is hurt. The second condition causes problems when data is being written to disk. It is faster to write contiguous data than noncontiguous data scattered over a drive or array. In addition, lots of empty spaces contribute to more physical file fragmentation.

If your SQL Server is highly transactional, with mostly INSERTS, UPDATES, and DELETES, physical disk fragmentation is less of an issue because few data pages are read, and writes are small. But if you are performing lots of SELECTS on your data, especially any form of a scan, then physical file fragmentation can become a performance issue as many data pages need to be read, causing the disk head to perform a lot of extra work.

Built into Windows 2000 and Windows 2003 is a built-in defragmentation utility, but it is slow, doesn't always do a good job of defragmentation, and cannot be easily scheduled. Because of this, I decided to give Executive Software's Diskeeper 8.0 software a try. This defragmentation utility comes in five different versions, including:

- Home
- Professional
- Administrator
- Standard Server
- Server Enterprise

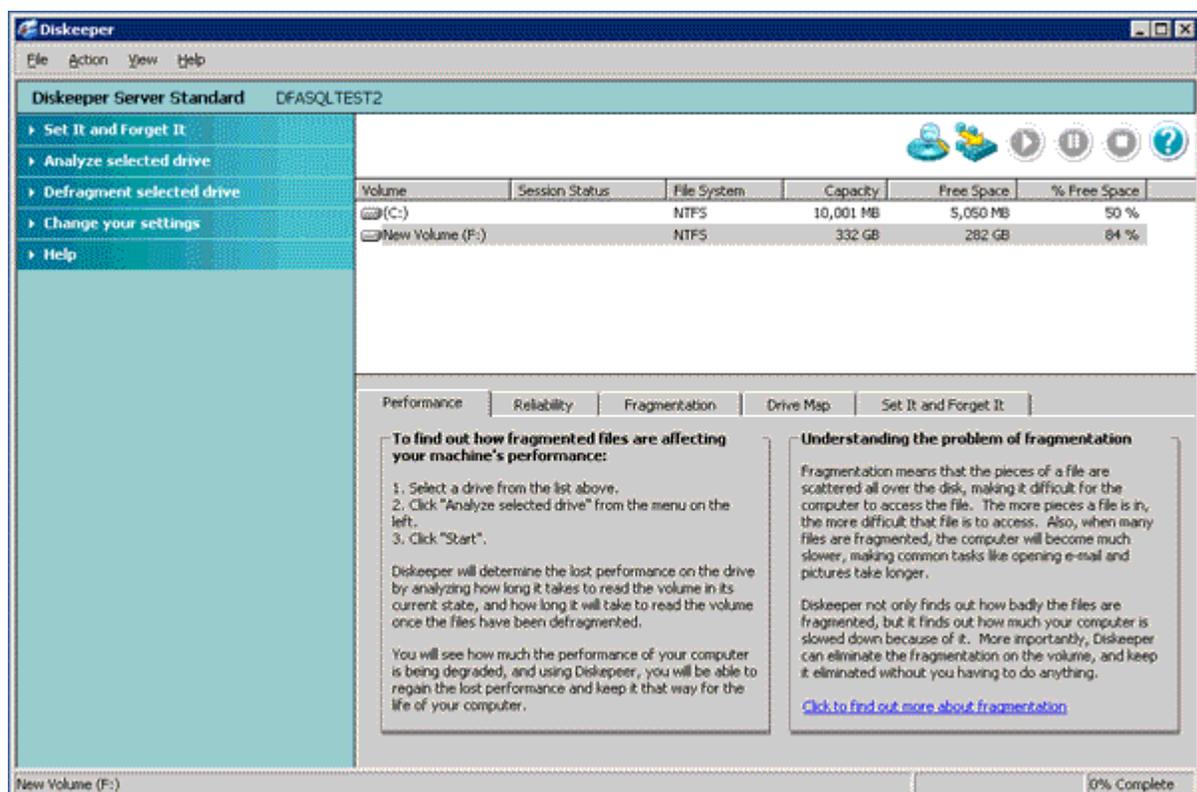
For this article, I used both Professional and Standard Server. Both are very similar, but have a slightly different feature set, as you might expect. The Professional Edition is designed for your desktop, while the Standard Server is designed for servers. The goal of this article is to tell you how Diskeeper works; it is not an article testing the pros and cons of the software.

## Diskeeper in Action

One of the best ways to see what Diskeeper can do for you is to give it a try, and that's what I am going to do here. To see what Diskeeper can do for one of my SQL Server's, I installed it on a SQL Server that had been in production for about 5 months, and had never been defragged before.

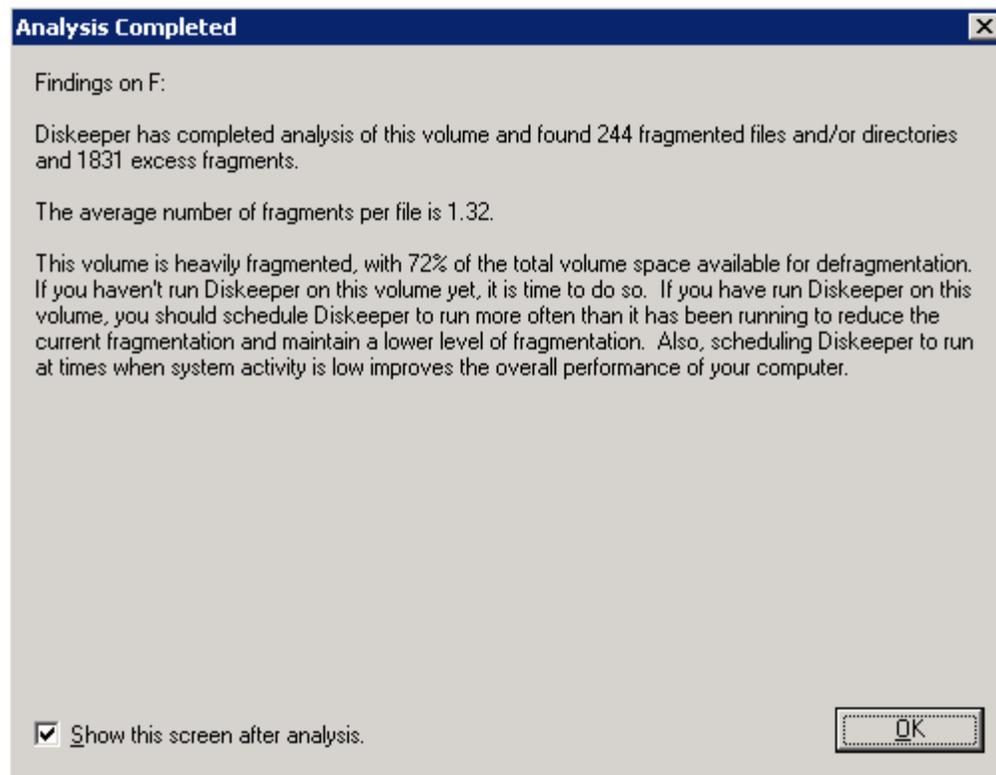
Diskeeper includes a tool that allows you to analyze the amount of fragmentation before and after the defragmentation occurs, which can be enlightening, as you will soon see.

So after installing Diskeeper, I started it up, and here's what the screen looks like (modified to fit the web page).

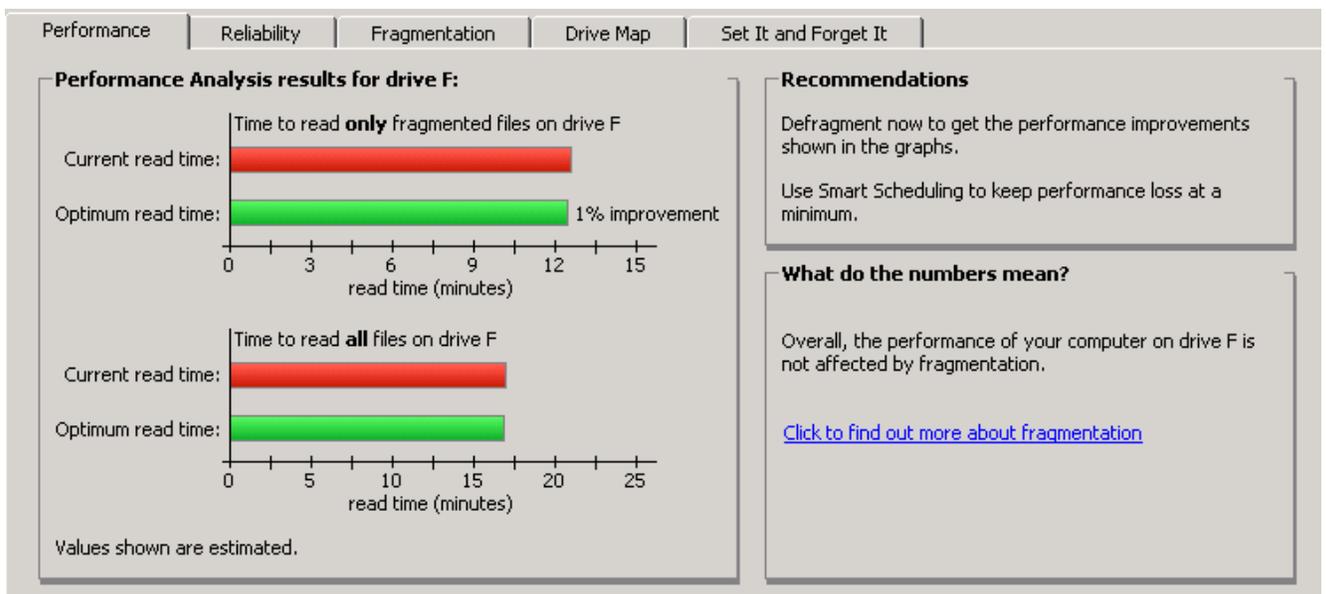


The first thing I wanted to do was to see how fragmented the files were on the server, especially on drive F, where the database MDFs, LDFs, and backup files were located. In addition, the SQL Server executable files were also located on drive F. Drive C only had the OS.

So my first step was to analyze drive F. Diskeeper produces several reports after an analysis is done, as you can see here



As you can see in this report, it has found 244 fragmented files and 1,831 fragments, which Diskeeper considers a heavily fragmented disk, and it recommends that fragmentation take place. Besides the summary above, it also produces some addition reports, which we see below.



The Performance Report above is a little more ambiguous than the first report. The graph above indicate that disk I/O performance will only be improved about 1% if defragmentation is performed. The recommendation is to defrag now, but under the section that says, "What do the numbers mean?" the report says that overall, performance is not affected by the current level of fragmentation.

The screenshot shows the Windows Reliability and Performance tool interface. The 'Reliability' tab is selected, displaying a 'Critical' reliability analysis for drive F: with a red 'X' icon. The analysis states that the computer's reliability is severely affected because the volume is heavily fragmented (73% fragmentation). Recommendations include defragmenting drive F: now using Diskkeeper. A separate section titled 'Understanding the effects of fragmentation on reliability' explains that fragmentation of frequently used files can lead to crashes and hangs. A link is provided to find out more about fragmentation.

On the other hand, the Reliability Report is not ambiguous at all. It claims the computer's reliability is severely affected by the current level of fragmentation and that the drive needs to be defragged. The claim here is that the fragmentation can lead the OS to having trouble doing its normal, everyday work, and can cause crashes and program hangs if not corrected.

The screenshot shows the 'Fragmentation' tab of the Windows Reliability and Performance tool. It displays a detailed 'Results of fragmentation analysis for the drive F:'. The analysis is divided into three main sections: Volume information, Fragmentation percentage, and File fragmentation. It also includes details on Paging file fragmentation, Directory fragmentation, and Master File Table (MFT) fragmentation. At the bottom, there are links to 'Print report', 'Save report', and 'Click to view the most fragmented files'.

Volume New Volume (F:):	
Volume size	= 332 GB
Cluster size	= 4 KB
Used space	= 51,050 MB
Free space	= 282 GB
Percent free space	= 84 %

Fragmentation percentage	
Volume fragmentation	= 11 %
Data fragmentation	= 73 %

File fragmentation	
Total files	= 4,734
Average file size	= 12,542 KB
Total fragmented files	= 221
Total excess fragments	= 1,533
Average fragments per file	= 1.32

Paging file fragmentation	
Paging/Swap file size	= 0 bytes
Total fragments	= 0

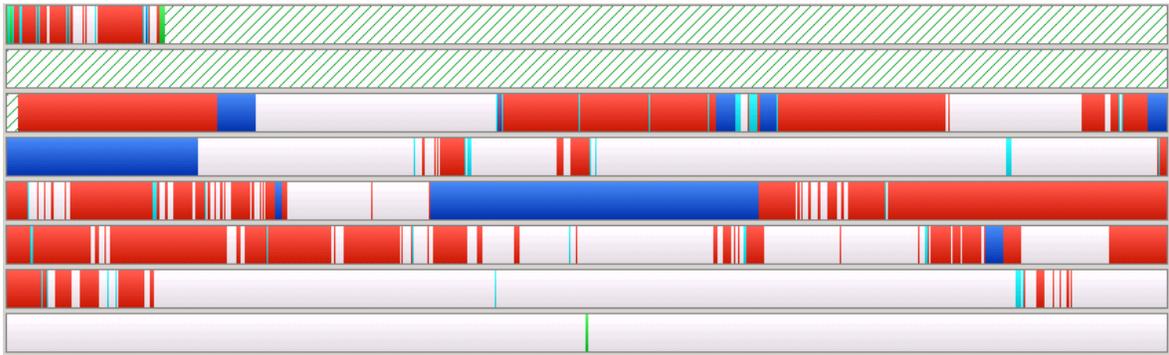
  

Directory fragmentation	
Total directories	= 127
Fragmented directories	= 23
Excess directory fragments	= 298

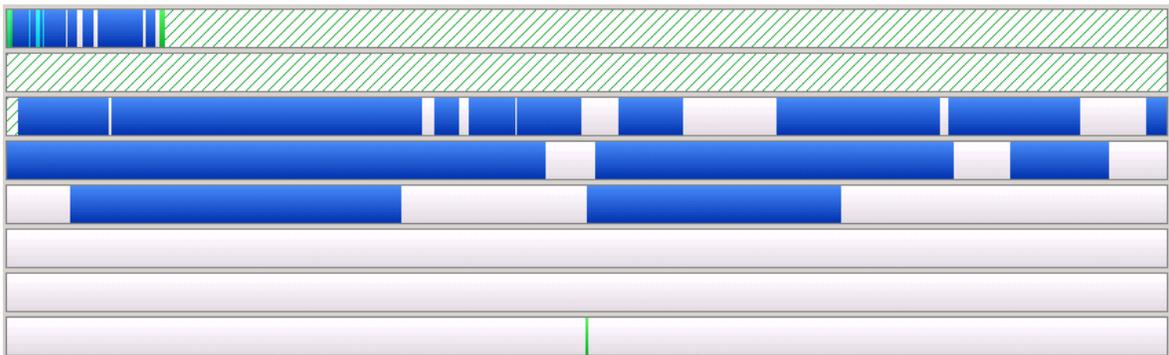
Master File Table (MFT) fragmentation	
Total MFT size	= 5,652 KB
MFT records In Use	= 4,880
Percent MFT in use	= 86 %
Total MFT fragments	= 2

The Fragmentation Report provides a lot of interesting data, which is used to support the recommendations made earlier.



The Drive Map Report is the most visually interesting one to me, and show how the file fragments are spread about the disk. The blue indicates non-fragmented files, and the red indicates fragmented files. Green refers to files that cannot be moved, such as some system files.

Based on Diskeeper's recommendation to defrag the drive, I manually selected this option and waited for the defragmentation to complete. When it did, here's what the drive map looked like.



As you can see, Diskeeper virtually eliminated all disk fragmentation on this drive. I won't bore you by showing you all of the other reports again, but they indicated that the disk was now 100% defragged. At this point, file fragmentation is not an issue and cannot negatively affect SQL Server's performance.

## Now What?

Now that the disk is defragged, are we done? No. Fragmentation never stops. Although NTFS will try to minimize file fragmentation, it doesn't do a very good job at it. Because of this, defragmentation need to be done continually, if you want optimal disk I/O performance. To accomplish this goal, Diskeeper allows you to schedule defragmentation on an as needed basis. While you have several scheduling options, the easiest to use is the "Set it and Forget It" option. By selecting this option, Diskeeper automatically works to defrag your system while using the least

amount of resources possible, on an ongoing basis. This setting should work well for most SQL Server's, assuming they are not already overextended. If your SQL Server is very busy, you will want to manually schedule defrags at time when the server's resources are less taxed.

### **Can Diskeeper Defrag Open SQL Server MDF and LDF Files?**

One of the biggest concerns about disk defragmentation is defragmenting files that are currently in use. In other words, if the file is actively being used, can Diskeeper, or any defrag utility for that matter, defrag it?

When I directed this question to Executive Software, the response was: "Diskeeper defragments the drive on the OS level, using the Windows MoveFile API's". Having this in mind, Diskeeper is able to safely defragment a drive on an OS level whether or not it is running a SQL or Exchange Server.

While my tests are not finitely conclusive, I ran my experiment above, the SQL Server MDF and LDF files were indeed defragged, even though they were open at the time.

### **Give Diskeeper a Try**

Fortunately, you can try Diskeeper (any version) for free. That is the best way to find out if it will meet your needs.

*About the author:* Brad M. McGehee is a full-time DBA with a large manufacturing company, and publisher of [www.SQL-Server-Performance.Com](http://www.SQL-Server-Performance.Com), a website specializing in SQL Server performance tuning and clustering. He is an MVP, MCSE+I, MCSD, and former MCT.