

# Engineering Windows 7

**Welcome to our blog dedicated to the engineering of Microsoft Windows 7**

## **Disk Defragmentation – Background and Engineering the Windows 7 Improvements**



**Steven Sinofsky** 25 Jan 2009 3:00 AM

**89**

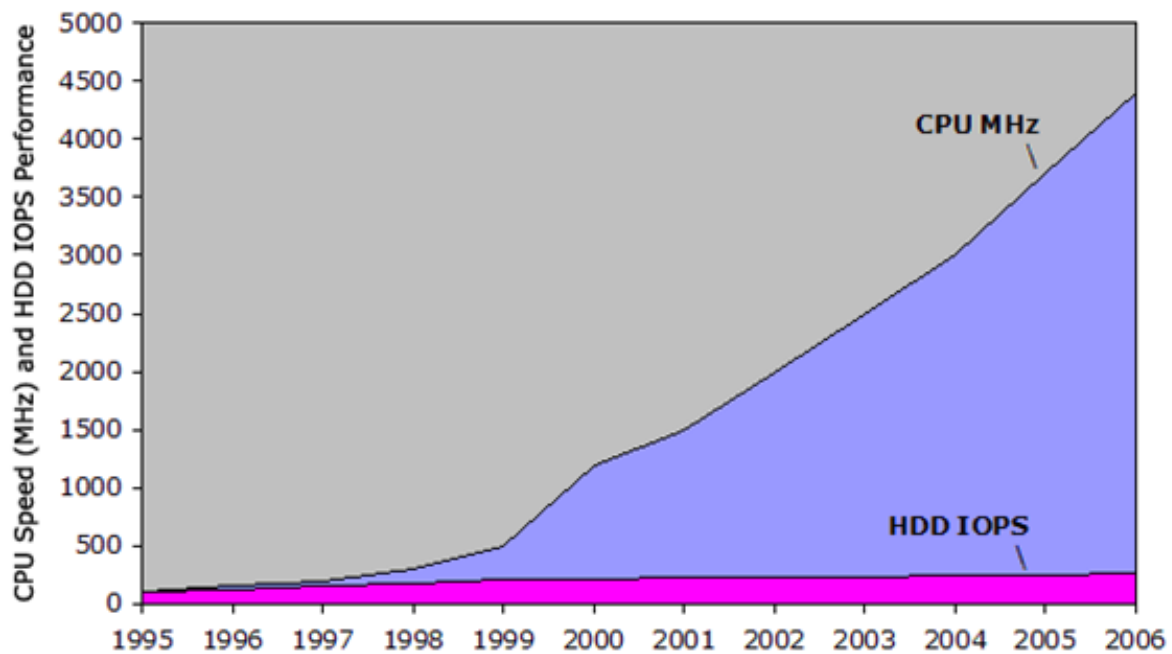
*One of the features that you've been pretty clear about (I've received over 100 emails on this topic!) is the desire to improve the disk defrag utility in Windows 7. We did. And from blogs we saw a few of you noticed, which is great. This is not as straight forward as it may appear. We know there's a lot of history in defrag and how "back in the day" it was a very significant performance issue and also a big mystery to most people. So many folks came to know that if your machine is slow you had to go through the top-secret defrag process. In Windows Vista we decided to just put the process on autopilot with the intent that you'd never have to worry about it. In practice this turns out to be true, at least to the limits of automatically running a process (that is if you turn your machine off every night then it will never run). We received a lot of feedback from knowledgeable folks wanting more information on defrag status, especially during execution, as well as more flexibility in terms of the overall management of the process. This post will detail the changes we made based on that feedback. In reading the mail and comments we received, we also thought it would be valuable to go into a little bit more detail about the process, the perceptions and reality of performance gains, as well as the specific improvements. This post is by Rajeev Nagar and Matt Garson, both are Program Managers on our File System feature team. --Steven*

In this blog, we focus on disk defragmentation in Windows 7. Before we discuss the changes introduced in Windows 7, let's chat a bit about what fragmentation is, and its applicability.

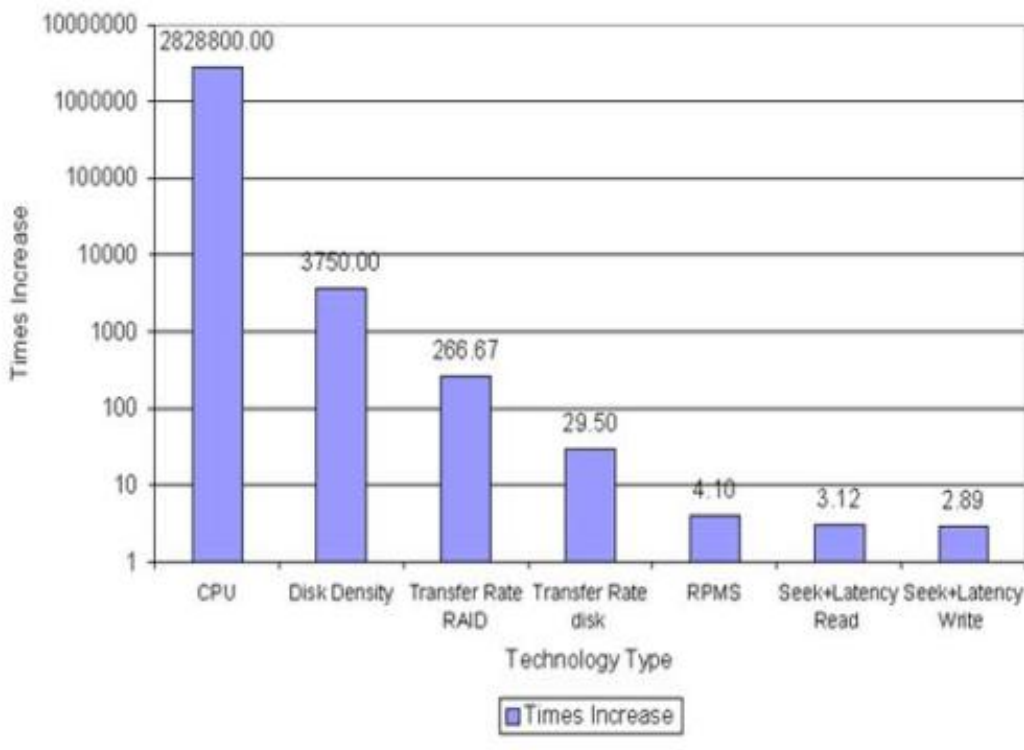
Within the storage and memory hierarchy comprising the hardware pipeline between the hard disk and CPU, hard disks are relatively slower and have relatively higher latency. Read/write times from and to a hard disk are measured in milliseconds (typically, 2-5 ms) – which sounds quite fast until compared to a 2GHz CPU that can compute data in less than 10 nanoseconds (on average), once the data is in the L1 memory cache of the processor.

This performance gap has only been increasing over the past 2 decades – the figures below are noteworthy.

Historical Trends of CPU and IOPS performance



Performance Improvement Various Technologies 1977-2005



In short, the figures illustrate that while disk capacities are increasing, their ability to transfer data or write new data is not increasing at an equivalent rate – so disks contain **more** data that takes **longer** to read or write. Consequently, fast CPUs are relatively idle, waiting for data to do work on.

Significant research in Computer Science has focused on improving overall system I/O performance, which has lead to two principles that the operating system tries to follow:

1. Perform less I/O, i.e. try and minimize the number of times a disk read or write request is

issued.

2. When I/O is issued, transfer data in relatively large chunks, i.e. read or write in bulk.

Both rules have reasonably simply understood rationale:

1. Each time an I/O is issued by the CPU, multiple software and hardware components have to do work to satisfy the request. This contributes toward increased latency, i.e., the amount of time until the request is satisfied. This latency is often directly experienced by users when reading data and leads to increased user frustration if expectations are not met.
2. Movement of mechanical parts contributes substantially to incurred latency. For hard disks, the “*rotational time*” (time taken for the disk platter to rotate in order to get the right portion of the disk positioned under the disk head) and the “*seek time*” (time taken by the head to move so that it is positioned to be able to read/write the targeted track) are the two major culprits. By reading or writing in large chunks, the incurred costs are amortized over the larger amount of data that is transferred – in other words, the “per unit” data transfer costs decrease.

File systems such as NTFS work quite hard to try and satisfy the above rules. As an example, consider the case when I listen to the song “Hotel California” by the Eagles (one of my all time favorite bands). When I first save the 5MB file to my NTFS volume, the file system will try and find enough contiguous free space to be able to place the 5MB of data “together” on the disk. Since logically related data (e.g. contents of the same file or directory) is more likely to be read or written around the same time. For example, I would typically play the entire song “Hotel California” and not just a portion of it. During the 3 minutes that the song is playing, the computer would be fetching portions of this “related content” (i.e. sub-portions of the file) from the disk until the entire file is consumed. By making sure the data is placed together, the system can issue read requests in larger chunks (often pre-reading data in anticipation that it will soon be used) which, in turn, will minimize mechanical movement of hard disk drive components and also ensure fewer issued I/Os.

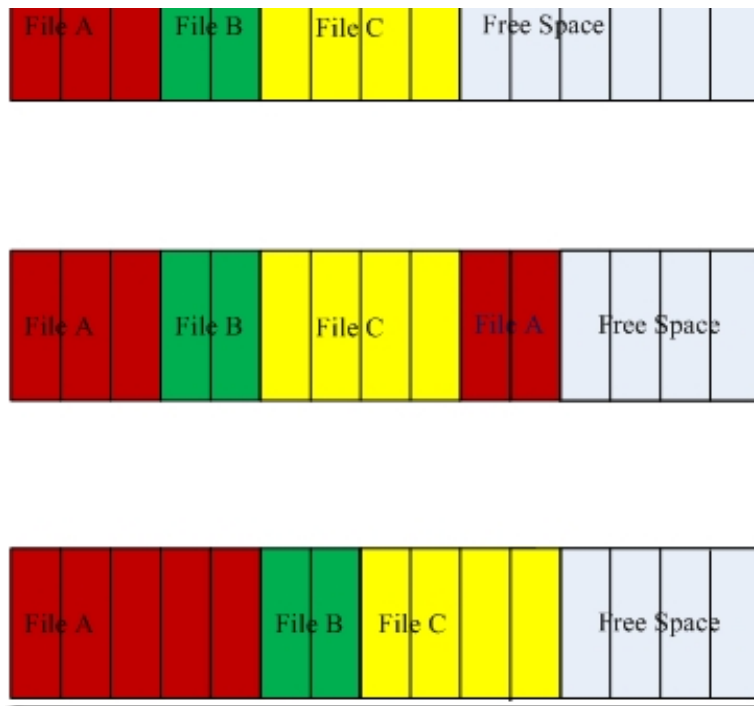
Given that the file system tries to place data contiguously, when does fragmentation occur? Modifications to stored data (e.g. adding, changing, or deleting content) cause changes in the on-disk data layout and can result in fragmentation. For example, file deletion naturally causes space de-allocation and resultant “holes” in the allocated space map – a condition we will refer to as “fragmentation of available free space”. Over time, contiguous free space becomes harder to find leading to fragmentation of newly stored content. Obviously, deletion is not the only cause of fragmentation – as mentioned above, other file operations such as modifying content in place or appending data to an existing file can eventually lead to the same condition.

So how does defragmentation help? In essence, defragmentation helps by moving data around so that it is once again placed more optimally on the hard disk, providing the following benefits:

1. Any logically related content that was fragmented can be placed adjacently
2. Free space can be coalesced so that new content written to the disk can be done so efficiently

The following diagram will help illustrate what we’re discussing. The first illustration represents an ideal state of a disk – there are 3 files, A, B, and C, and all are stored in contiguous locations; there is no fragmentation. The second illustration represents a fragmented disk – a portion of data associated with File A is now located in a non-contiguous location (due to growth of the file). The third illustration shows how data on the disk would look like once the disk was defragmented.





Nearly all modern file systems support defragmentation – the differences generally are in the defragmentation mechanism, whether, as in Windows, it's a separate, schedulable task or, whether the mechanism is more implicitly managed and internal to the file system. The design decisions simply reflect the particular design goals of the system and the necessary tradeoffs. Furthermore, it's unlikely that a general-purpose file system could be designed such that fragmentation never occurred.

Over the years, defragmentation has been given a lot of emphasis because, historically, fragmentation was a problem that could have more significant impact. In the early days of personal computing, when disk capacities were measured in megabytes, disks got full faster and fragmentation occurred more often. Further, memory caches were significantly limited and system responsiveness was increasingly predicated on disk I/O performance. This got to a point that some users ran their defrag tool weekly or even more often! Today, very large disk drives are available cheaply and % disk utilization for the average consumer is likely to be lower causing relatively less fragmentation. Further, computers can utilize more RAM cheaply (often, enough to be able to cache the data set actively in use). That together, with improvements in file system allocation strategies as well as caching and pre-fetching algorithms, further helps improve overall responsiveness. Therefore, while the performance gap between the CPU and disks continues to grow and fragmentation does occur, combined hardware and software advances in other areas allow Windows to mitigate fragmentation impact and deliver better responsiveness.

So, how would we evaluate fragmentation given today's software and hardware? A first question might be: *how often does fragmentation actually occur and to what extent?* After all, 500GB of data with 1% fragmentation is significantly different than 500GB with 50% fragmentation. Secondly, *what is the actual performance penalty of fragmentation, given today's hardware and software?* Quite a few of you likely remember various products introduced over the past two decades offering various performance enhancements (e.g. RAM defragmentation, disk compression, etc.), many of which have since become obsolete due to hardware and software advances.

The incidence and extent of fragmentation in average home computers varies quite a bit depending on available disk capacity, disk consumption, and usage patterns. In other words, there is no general answer. The actual performance impact of fragmentation is the more interesting question but even more complex to accurately quantify. A meaningful evaluation of the performance penalty of fragmentation would require the following:

- Availability of a system that has been "aged" to create fragmentation in a typical or representative manner. But, as noted above, there is no single, representative behavior. For example, the frequency and extent of fragmentation on a computer used primarily for web

- Selection of meaningful disk-bound metrics e.g. boot and first-time application launch post boot.
- Repeated measurements that can be statistically relevant

Let's walk through an example that helps illustrate the complexity in directly correlating extent of fragmentation with user-visible performance.

In Windows XP, any file that is split into more than one piece is considered fragmented. Not so in Windows Vista if the fragments are large enough – the defragmentation algorithm was changed (from Windows XP) to ignore pieces of a file that are larger than 64MB. As a result, defrag in XP and defrag in Vista will report different amounts of fragmentation on a volume. So, which one is correct? Well, before the question can be answered we must understand why defrag in Vista was changed. In Vista, we analyzed the impact of defragmentation and determined that the most significant performance gains from defrag are when pieces of files are combined into *sufficiently large chunks* such that the impact of disk-seek latency is not significant relative to the latency associated with sequentially reading the file. This means that there is a point after which combining fragmented pieces of files has no discernible benefit. In fact, there are actually negative consequences of doing so. For example, for defrag to combine fragments that are 64MB or larger requires significant amounts of disk I/O, which is against the principle of minimizing I/O that we discussed earlier (since it decreases total available disk bandwidth for user initiated I/O), and puts more pressure on the system to find large, contiguous blocks of free space. Here is a scenario where a certainly amount of fragmentation of data is just fine – doing nothing to decrease this fragmentation turns out to be the right answer!

Note that a concept that is relatively simple to understand, such as the amount of fragmentation and its impact, is in reality much more complex, and its real impact requires comprehensive evaluation of the entire system to accurately address. The different design decisions across Windows XP and Vista reflect this evaluation of the typical hardware & software environment used by customers. Ultimately, when thinking about defragmentation, it is important to realize that there are many additional factors contributing towards system responsiveness that must be considered beyond a simple count of existing fragments.

The defragmentation engine and experience in Windows 7 has been revamped based on continuous and holistic analysis of impact on system responsiveness:

In Windows Vista, we had removed all of the UI that would provide detailed defragmentation status. We received feedback that you didn't like this decision, so we listened, evaluated the various tradeoffs, and have built a new GUI for defrag! As a result, in Windows 7, you can monitor status more easily and intuitively. Further, defragmentation can be safely terminated any time during the process and on all volumes very simply (if required). The two screenshots below illustrate the ease-of-monitoring:





Disk Defragmenter consolidates fragmented files on your computer's hard disk to improve system performance. [How does Disk Defragmenter help?](#)

Schedule:

**Scheduled defragmentation is turned on**

Run at 1:00 AM every Wednesday

Next scheduled run: 1/28/2009 1:00 AM

[Configure schedule...](#)

Current status:

Disk	Last Run	Progress
win7betavhd (C:)	Never run	
vhdboot (D:)	1/14/2009 11:51 AM (0% fragmented)	
secondos (E:)	1/14/2009 11:51 AM (0% fragmented)	
data1 (F:)	1/21/2009 8:25 AM (0% fragmented)	
data2 (G:)	1/14/2009 11:25 AM (0% fragmented)	
(X:)	1/21/2009 8:25 AM (0% fragmented)	

To best determine if your disks need defragmenting right now, you need to first analyze your disks.

[Analyze disk](#)

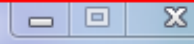
[Defragment disk](#)

[Close](#)



Disk Defragmenter

[Send Feedback](#)




Disk Defragmenter consolidates fragmented files on your computer's hard disk to improve system







 performance. [How does Disk Defragmenter help?](#)

Schedule: \_\_\_\_\_


**Scheduled defragmentation is turned off**


 **Turn on schedule...**

Current status: \_\_\_\_\_

Disk	Last Run	Progress
 win7betavhd (C:)	Never run	
 vhdboot (D:)	1/14/2009 11:51 AM (0% fragmented)	
 segundos (E:)	1/14/2009 11:51 AM (0% fragmented)	
 data1 (F:)	1/21/2009 8:25 AM (0% fragmented)	
 data2 (G:)	1/14/2009 11:25 AM (0% fragmented)	
 (X:)	1/21/2009 8:25 AM (0% fragmented)	

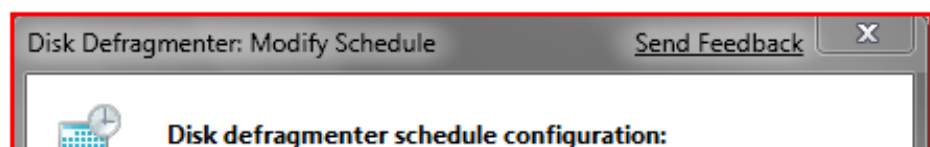
To best determine if your disks need defragmenting right now, you need to first analyze your disks.

 **Analyze disk**

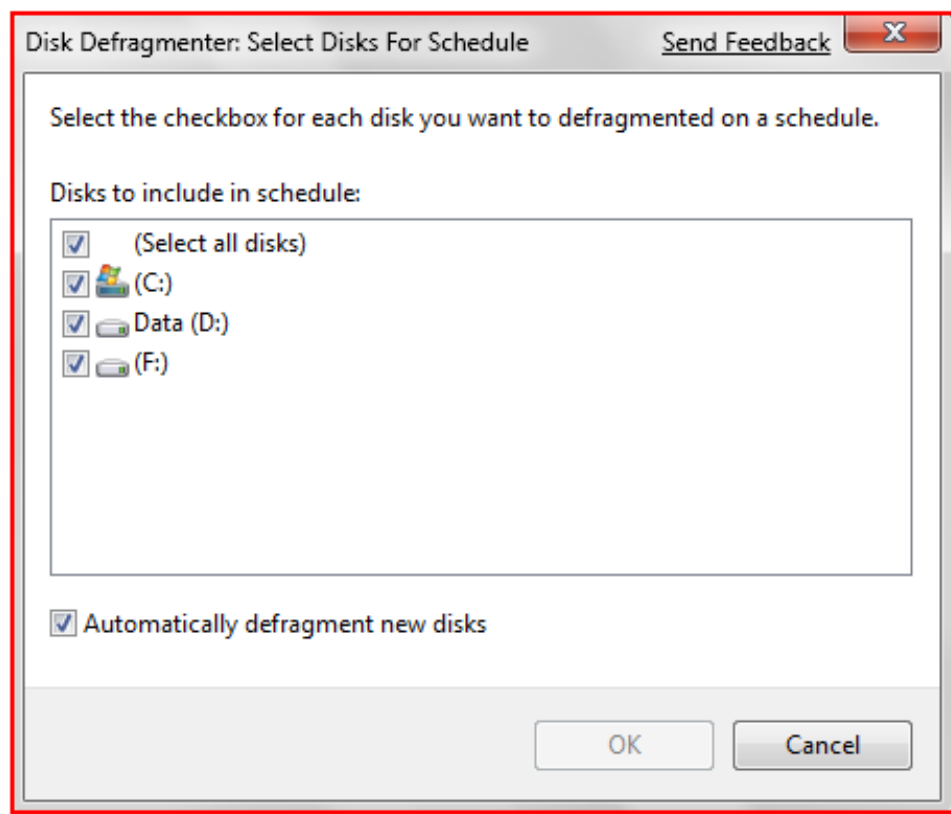
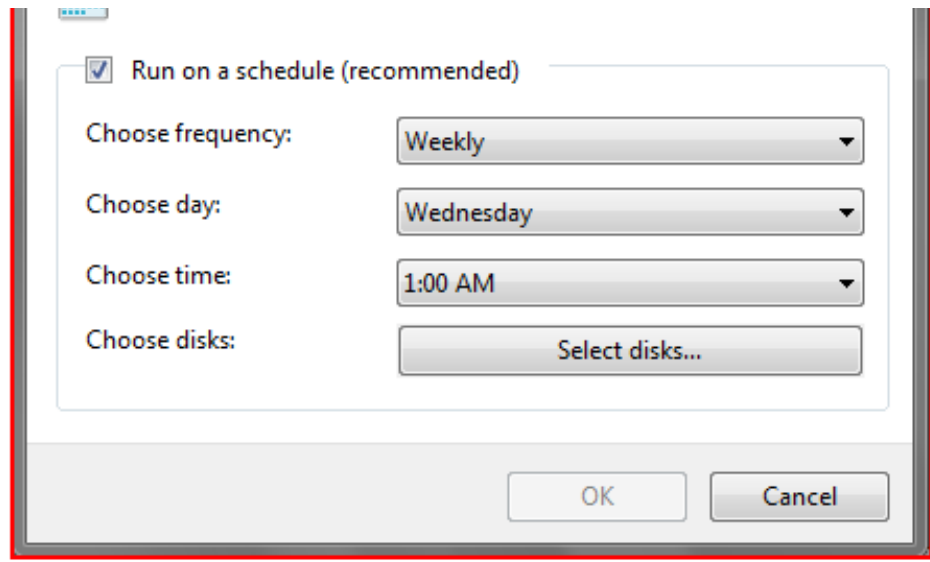
 **Defragment disk**

**Close**

In Windows XP, defragmentation had to be a user-initiated (manual) activity i.e. it could not be scheduled. Windows Vista added the capability to schedule defragmentation – however, only one volume could be defragmented at any given time. Windows 7 removes this restriction – multiple volumes can now be defragmented in parallel with no more waiting for one volume to be defragmented before initiating the same operation on some other volume! The screen shot below shows how defragmentation can be concurrently scheduled on multiple volumes:







Among the other changes under the hood in Windows 7 are the following:

- Defragmentation in Windows 7 is more comprehensive – many files that could not be re-located in Windows Vista or earlier versions can now be optimally re-placed. In particular, a lot of work was done to make various NTFS metadata files movable. This ability to relocate NTFS metadata files also benefits volume shrink, since it enables the system to pack all files and file system metadata more closely and free up space “at the end” which can be reclaimed if required.
- If solid-state media is detected, Windows disables defragmentation on that disk. The physical nature of solid-state media is such that defragmentation is not needed and in fact, could decrease overall media lifetime in certain cases.
- By default, defragmentation is disabled on Windows Server 2008 R2 (the Windows 7 server release). Given the variability of server workloads, defragmentation should be enabled and scheduled only by an administrator who understands those workloads.



Best practices for using defragmentation in Windows 7 are simple – you do not need to do anything! Defragmentation is scheduled to automatically run periodically and in the background with minimal impact to foreground activity. This ensures that data on your hard disk drives is efficiently placed so the system can provide optimal responsiveness and I can continue to enjoy glitch free listening to the Eagles :-).

Rajeev and Matt

## **Comments**



**ari9910**

25 Jan 2009 9:12 PM

Has it ever occurred to anyone on the team that the best time to run things like the defragmenter, updater, and backup programs are when we AREN'T using the system. as said in the blog, it won't run if we turn the system off every night. So if those things ran when we weren't using the computer yet it was still on, those operations would run and we wouldn't have to worry about it. I find it increasingly annoying that the updater wants to update (and annoy me about it) while i'm using it because i shut the computer off at the 3:00AM time it is set to run. Similar goes for the defragmenter. I install 3rd party programs because of that, i would appreciate if I didn't have to do that because, i personally love Microsoft and hope that one day it will actually think more about customer satisfaction and less about profits (to a reasonable degree of course).



**Asesh**

25 Jan 2009 10:48 PM

What about a new file system that doesn't need defragmenting? When will see such a file system in Windows? Linux already has one. We had high hopes from WinFS and hopefully when it's done it will remove the need to defragment our hard disks from time-to-time.

Please don't release the next version of Windows (after Windows 7) until WinFS is done and a UNIX based kernel or a better one is implemented because Windows is still prone to viruses unlike Mac OS X and Linux which are more secure than Windows.

I hope all the Windows' coders out there will see this because it's high time so we need such a file system and kernel. Thanks



**mark\_ms**

25 Jan 2009 10:51 PM

I've had a keen interest in defrag products since the days of the 286 and 40Mb hard drives. I find it particularly refreshing to learn about where the line crosses between useful defragging and wasted effort and how much newer technology has been minimizing the effect of fragmentation. Now that I have a better understanding of how Windows 7 defrags, I am now even less concerned about managing it myself.

I really pity the 3rd party defragmenters out there. They now have a higher bar to clear to convince me that the default defragger isn't good enough.

I don't really mind the defragger kicking in when I'm doing low I/O work e.g. browsing the web but from the screenshots, it seems that this has to be scheduled. I can't always know what kind of work I would be doing and am reluctant to specify a time when I know the computer would be on and I am working on it. So, I would prefer that there is an option for the defragger to kick in during low I/O if it hasn't been getting a chance to run at the normal scheduled time.

One thing I do is recompress tv recordings. Since my recompressor is currently single-threaded, I am very much tempted to run two at the same time on my dual-core machine. Would this create much more fragmented files than if I run them one at a time?

**obsidience**

25 Jan 2009 11:24 PM

First off, I'm delighted that tool disk defragmentation tool was deemed important enough to be blogged on. Kudos!

But if you want to improve the speed and efficiency of this new OS then you need to think about:

- 1) A defrag tool that can be scheduled to run outside of windows when no files are locked by the OS. This would allow for a fully defragged disk.
- 2) A defrag tool that can profile a standard computer session and defrag files accordingly. Moving heavily accessed files to the outer areas of the disk platter where data density to rotational speed is highest
- 3) A defrag tool that analyzes the boot up pattern and can place files sequentially along the zone allowing for less access and more reads.

If you investigated these ideas I'm sure you can further reduce your boot up times by a substantial amount.

**SouthPaw42**

25 Jan 2009 11:45 PM

Wouldn't a file system that included a process to reduce fragmentation during file creation and update. I have an OS HDD that is 60% empty space but after 30 days is 50% fragmented. A drive that empty should not be fragmented.

**Xentrax [Vyacheslav Lanovets]**

26 Jan 2009 12:49 AM

I believe that scheduling logic must be changed too. People who do not understand what defrag is tend to turn off their PCs early. Windows MUST detect that defrag did not have a chance to run for a month or two and advise changing scheduled time accordingly or even do it silently according to user behaviour pattern. But this will be too user-friendly for Microsoft :)

For advanced users Vista defrag can be scheduled to run at boot time, and in this mode it seems to defrag more than during user session. And reboot can be scheduled too. And sometimes even hardware wakeup can be scheduled :)

As Vista itself, Vista defrag is already good but it seems like scheduling logic did not change in Windows 7. Kind of: it won't work for my mom who at the same time really needs defrag and does not leave your PC turned on at 1:00 AM!

Windows 8 or 9?

**II ARROWS**

26 Jan 2009 3:44 AM

Asesh, "file system that doesn't need defragmenting" simply doesn't exist.

You wrote "Unix", those file systems work defragmenting while writing new file or expanding them. The results are slower writing time, in most case if a file is written you are working and this slows your work. Windows perform this while you are not working.

Think about converting a video, this operation requires a large amount of IO operation in RAM and in disk, generating a lot of file that have to be deleted at the end of the process. If the file system has to defrag while writing, conversion time would be increased.

Also, you wrote "WinFS". WinFS was not a file system, but an added layer that works on NTFS, an heavy one.

An idea on which WinFS was built is splitting up the meta data from data, and in this article I see another step closer to WinFS.

**sirus**

26 Jan 2009 4:45 AM

@ Asesh

A file system which has the characteristic you mention doesn't exist. If you take Linux and its most used file system, which is EXT3, the fragmentation percentage is indeed low, however it's calculated only on big chunks, just like on Windows Vista. If you try another file system such as XFS which is certainly faster than EXT3 you'll notice that it suffers a greater fragmentation and due to that XFS is provided with a de-fragmentation tool.

Ending the off-topic, I'd like an "intelligent" de-fragmentation process that runs when I'm not heavily using my HD and possibly an "intelligent" scheduler which automatically modifies its configuration based on users' habits.

**Eiki**

26 Jan 2009 4:59 AM



**Robert Wray**

26 Jan 2009 5:17 AM

@Eiki

One of the bullet points at the end of the entry:

"If solid-state media is detected, Windows disables defragmentation on that disk. The physical nature of solid-state media is such that defragmentation is not needed and in fact, could decrease overall media lifetime in certain cases."



**Domenico**

26 Jan 2009 5:41 AM

My computer is always on With Vista or Windows 7.

automatically defrag form me is very interesting and work fine,

in the Beta of Windows 7, Defrag is even better.



**d\_e**

26 Jan 2009 5:47 AM

@Asesh: It seems you have no clue about kernels and filesystems. Especially the windows kernel.

1. WinFS is not a filesystem. It's a database. NTFS runs underneath.
2. There is not filesystem which doesn't fragment. Such a thing is impossible.
3. "UNIX or better". What do you mean by this? Unix-compatible systems aren't inherently more secure than Windows. The design of Unix is very old (and IMHO outdated). I'm confident that Microsoft is wise enough to keep their current kernel.
4. A secure kernel doesn't give you a secure computer. Because most users aren't computer literate and will gladly install anything if an email tells them to do so.

To the article: I believe defragmentation is misunderstood by most users. Many think this is some sort of magic powder that will make their machine much faster. It won't. Users shouldn't have to worry about defragmentation. The decision to hide all this in Vista was the right one. You guys get the 100+ emails from self-proclaimed experts which (mostly) have no clue what they're writing about.

The only thing I'm wondering is why my disk (according to the HDD LED) works only 50% when resuming from hibernate...

**II ARROWS**

26 Jan 2009 6:11 AM

I've a question about my test configuration:

I've an HDD with Vista x64 and 7 x64, another disk used for data.

I always move Document(and music, picture and video) folder on the second disk to preserve them if I need to format or simply change machine.

The structure is like XP:

My Documents

|----My Music

|----My Pictures

|----My Videos

After set in the library the folders and set as default save directories, if I boot Vista and try to access the Documents sub folder(Only music, pic and video) it alerts me that I need to be the owner. Only for those folder, files in My documents can be opened and modified.

May it depends because of the new "intelligent defrag" splitting meta data, and 7 doesn't want Vista to "undo" 7's work?

Or is just a bug that 7 wants those right?

**Asesh**

26 Jan 2009 9:45 AM

@d\_e

Read it carefully, I didn't say WinFS is a file system. And unix based kernel is more secure than Windows kernel :P Yes I do use both Linux and Mac OS X so I can say that.

**tgrand**

26 Jan 2009 10:41 AM

"The physical nature of solid-state media is such that defragmentation is not needed..."

Did you guys check this or just take some SSD manufacturers word for it? It's really disappointing to hear this from you. I was skeptical about this claim so I tested it myself. It wasn't hard at all to get a 30% performance hit reading a fragmented file from an SSD. Sure, that's not nearly as bad as it would've been on a spinning disk, but 30% is still significant - especially when you're paying a huge price premium to use one of these drives for performance reasons!

I argued this point with the manufacturer and they eventually conceded, but said the performance benefit of defragging was outweighed by the shortening of the drive's lifespan. I said "OK great, you should claim this, instead of claiming that it's 'not needed', which is a half-truth at best." As you might imagine, they weren't very receptive to that idea. And I only got to do some limited testing. I'm sure things get much worse when you have a nearly full system drive used over a long time.

By the way, here's why fragmentation hurts on an SSD despite the minimal "seek time" penalty: most SSDs only get their high throughput when the individual I/O requests are for sufficiently large amounts of data. Fragmentation can easily turn a file read operation from a handful of large fast reads into hundreds of tiny slow ones.

