NTFS Misreports Free Space?

ntdebug 3 Jul 2008 1:08 PM | 11

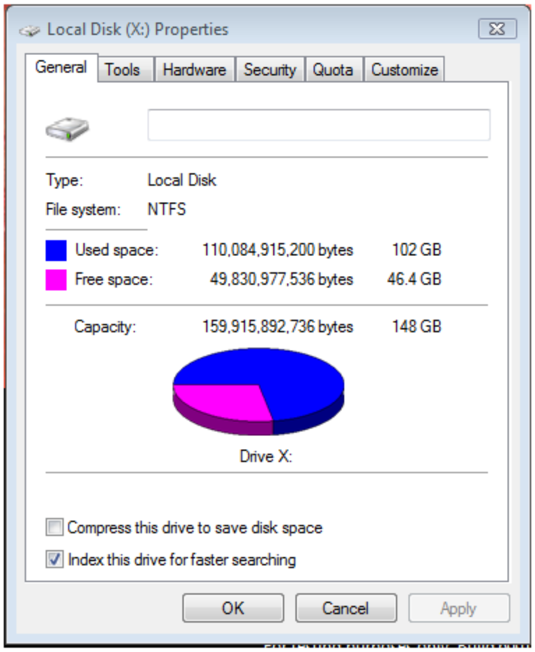
I have recently seen a number of issues where customers called in to report a significant difference between the "Size on disk" for the root of a volume, and the reported amount of "Used space" in the volume properties. While considering this, I noticed that my own C: drive had a mysterious 20GB difference.

Below is the story of how I found the answer.

Before we begin, there are two methods used for calculating disk usage…

Method 1 – Volume Bitmap Analysis
The % used and %free indication shown below with the pie chart is based on volume bitmap analysis. The hidden $Bitmap:$Data:"" stream is read from the volume via the FSCTL_GET_VOLUME_BITMAP IO control code, and the ones and zeroes are counted. A binary "1" represents an allocated cluster, while a "0" indicates a free cluster. The total number of set bits indicates the total number of allocated clusters on the volume. This is the fastest and most reliable method of calculating allocated and free space.

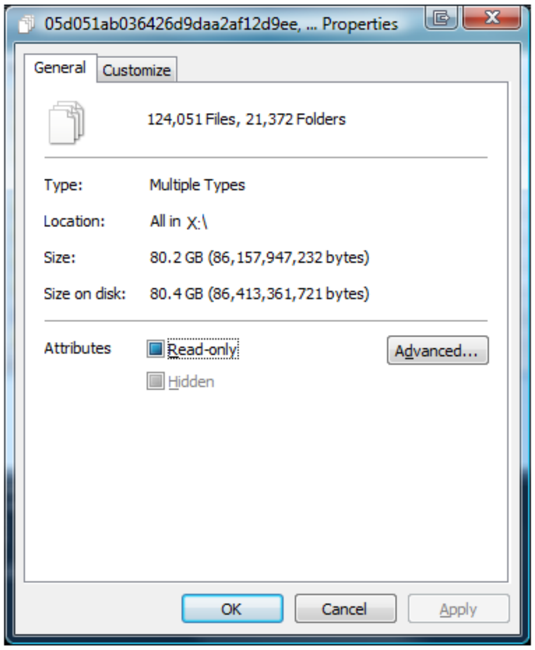
Method 2 – File Scanning with FindFirstFileEx / FindNextFile
Navigating to the "C:\" folder, we select all files/folders, then right click, and then select properties. A dialog will appear as the shell enumerates all files that it can find in the root and subfolders. The "Size" and "Size on disk" fields will eventually display the total size of all primary data streams for all files that are found by Explorer. In my case, this method only finds 80.25 GB of 102 GB (leaving a 21.75 GB difference).



Type: Local Disk
File system: NTFS

🟦	Used space:	110,084,915,200 bytes	102 GB
🟪	Free space:	49,830,977,536 bytes	46.4 GB
	Capacity:	159,915,892,736 bytes	148 GB

☐ Compress this drive to save disk space
☑ Index this drive for faster searching



124,051 Files, 21,372 Folders

Type: Multiple Types
Location: All in X:\
Size: 80.2 GB (86,157,947,232 bytes)
Size on disk: 80.4 GB (86,413,361,721 bytes)

Attributes ■ Read-only
☐ Hidden

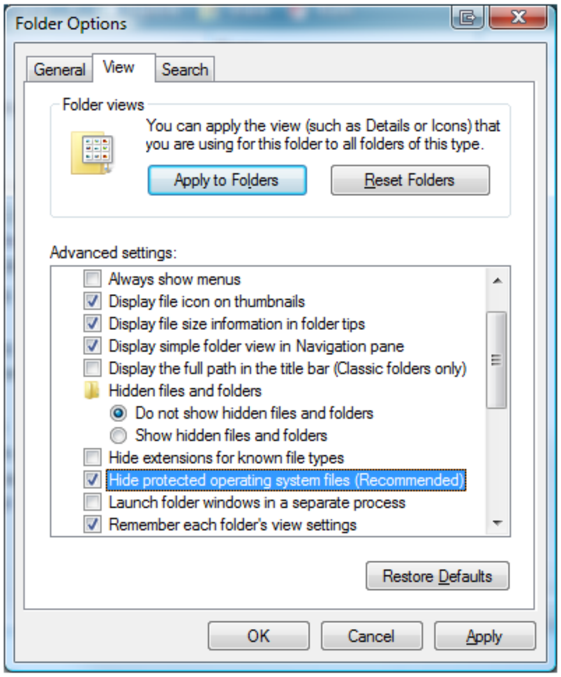
Why such a big difference?

The short answer is - **"If you can't see it with Explorer, it doesn't get added to the total size."**

How to find the hidden disk usage…

Step 1:

First, check to see if your files are all actually visible with the current user account and Explorer settings. Uncheck the "Hide protected operating system files" and select "Show hidden files and folders". When this is completed, look at the size on disk again.



Folder views

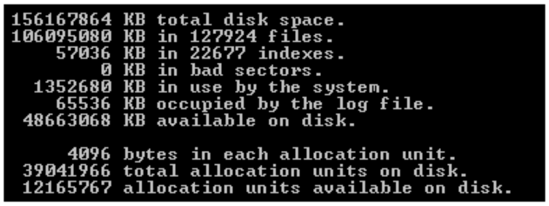
You can apply the view (such as Details or Icons) that you are using for this folder to all folders of this type.

Advanced settings:

- □ Always show menus
- ☑ Display file icon on thumbnails
- ☑ Display file size information in folder tips
- ☑ Display simple folder view in Navigation pane
- □ Display the full path in the title bar (Classic folders only)
- Hidden files and folders
  - ● Do not show hidden files and folders
  - ○ Show hidden files and folders
- □ Hide extensions for known file types
- ☑ Hide protected operating system files (Recommended)
- □ Launch folder windows in a separate process
- ☑ Remember each folder's view settings

Does it match the pie chart within a few percent? If not, continue to Step 2.

Step 2:
Run CHKDSK and save the output. The summary information at the end is what we want to see.



```
156167864 KB total disk space.
106095080 KB in 127924 files.
    57036 KB in 22677 indexes.
        0 KB in bad sectors.
  1352680 KB in use by the system.
    65536 KB occupied by the log file.
 48663068 KB available on disk.

     4096 bytes in each allocation unit.
 39041966 total allocation units on disk.
 12165767 allocation units available on disk.
```

Based on CHKDSK output, we can calculate the total metadata usage by adding up the following…

KB	GB	Description
57036	.05 GB	Space used by 22677 indexes.
0	0 GB	Space used by $Badclus file.
1352680	1.29 GB	Space used by $MFT.
65536	.06 GB	Space used by $Loffile.
1475252	**1.4 GB**	**Metadata Total**

In this example, metadata usage accounts for only 1.4 GB. If the metadata number were a high percentage of the usage, then we need to take closer a look at how the volume is being used instead of simply looking for hidden files. **High metadata usage will be the subject for part 2 of this blog.**

We can see from the CHKDSK output that the total space occupied by our user files is 106,095,080 KB (101.18 GB). This is a large percentage of the total usage, so we should be looking at the user files to see why we can't see these files with Explorer.

Step 3:
Do you have permissions to see all files/folders the volume?

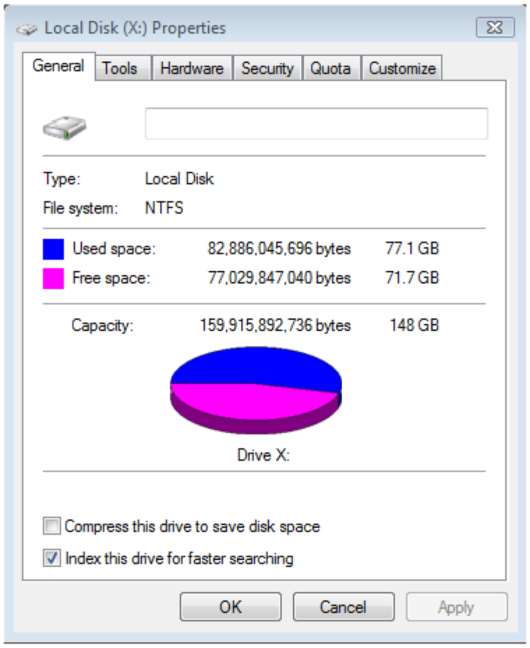
a. If this is a system drive, look through the "\Documents and Settings", or "\users" folder and see if you can browse all of the profile folders. If not, you may need contact the owner(s) to check their folder size and see what they have saved in their user folder. If they are like me, they will have 5 years worth of memory dumps, MP3's, VHD's, PDF's, etc.

b. Try "CACLS.EXE x:\ /T", or just browse through the folders on the drive looking for an "Access Denied" error. Once this happens, give yourself permissions to that part of the subtree (if permitted by your administrative privileges) or have someone who has access to that folder enumerate the contents for you. Check the number of files and the size on disk for all files in that folder and add it to the total.

c. Continue this process until you have a total of all files in all folders.

d. Does the total make sense? If not, then continue to the next step.

In the case of my mysterious 20 GB difference, I found an old user profile from a previous installation. Sure enough, I received an access denied error when browsing to it. To access the folder, I acquired ownership. This user was in the habit of collecting memory dumps and virtual machines in a subfolder on his desktop. I recovered quite a bit of free space by using the delete key. I rescanned the volume, but to my disappointment, there was still a significant difference.

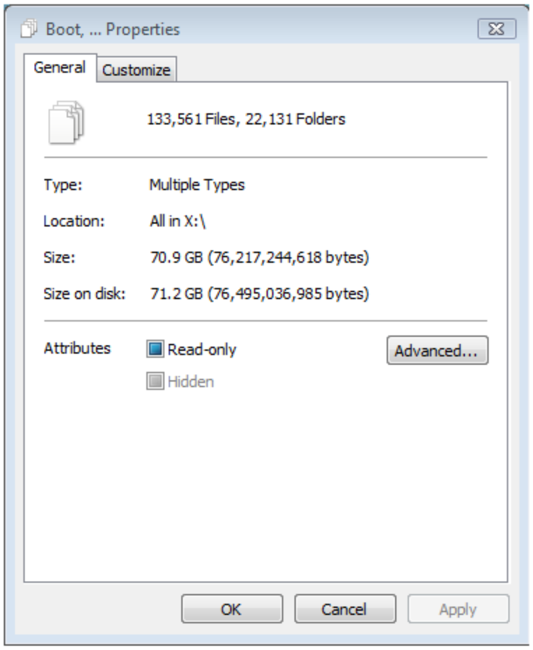


Type: Local Disk
File system: NTFS

Used space: 82,886,045,696 bytes 77.1 GB
Free space: 77,029,847,040 bytes 71.7 GB

Capacity: 159,915,892,736 bytes 148 GB

☐ Compress this drive to save disk space
☑ Index this drive for faster searching



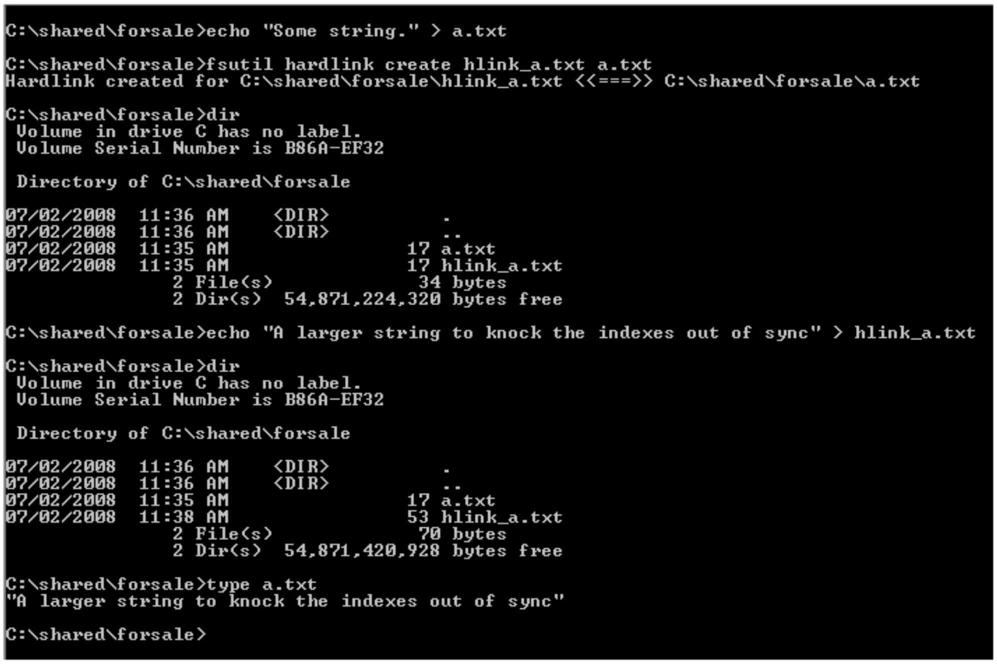
133,561 Files, 22,131 Folders

Type: Multiple Types
Location: All in X:\
Size: 70.9 GB (76,217,244,618 bytes)
Size on disk: 71.2 GB (76,495,036,985 bytes)

Attributes ■ Read-only
☐ Hidden

Step 4.
Are there alternate streams? Try using STREAMS.EXE from (**http://technet.microsoft.com/en-us/sysinternals/default.aspx**). Streams will recurse subdirectories and report space usage by alternate named streams. Pipe this to a text file and analyze it with a text editor or spreadsheet program.

Step 5.
Hard links can be problematic when calculating the total usage via the file scan method. Hard links are not extensively used, but this is worth mentioning. A hard link is an extra index entry that points to an existing file. These are created via the **CreateHardLink** function. Although the file actually exists in one location, each hard link has its own size field. Because of this, a single file can be added twice to the running total. Another thing to know is that hard links are not synchronized, so it is possible that only one link will show the actual file size (see the example below).

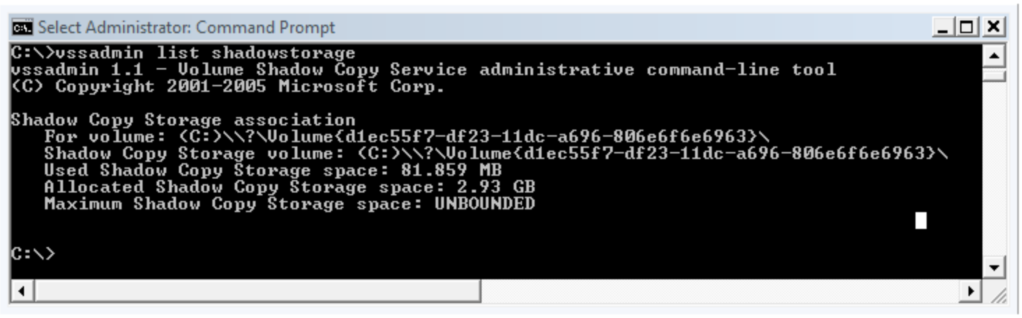


```
C:\shared\forsale>echo "Some string." > a.txt

C:\shared\forsale>fsutil hardlink create hlink_a.txt a.txt
Hardlink created for C:\shared\forsale\hlink_a.txt <<===>> C:\shared\forsale\a.txt

C:\shared\forsale>dir
 Volume in drive C has no label.
 Volume Serial Number is B86A-EF32

 Directory of C:\shared\forsale

07/02/2008  11:36 AM    <DIR>          .
07/02/2008  11:36 AM    <DIR>          ..
07/02/2008  11:35 AM                17 a.txt
07/02/2008  11:35 AM                17 hlink_a.txt
               2 File(s)             34 bytes
               2 Dir(s)  54,871,224,320 bytes free

C:\shared\forsale>echo "A larger string to knock the indexes out of sync" > hlink_a.txt

C:\shared\forsale>dir
 Volume in drive C has no label.
 Volume Serial Number is B86A-EF32

 Directory of C:\shared\forsale

07/02/2008  11:36 AM    <DIR>          .
07/02/2008  11:36 AM    <DIR>          ..
07/02/2008  11:35 AM                17 a.txt
07/02/2008  11:38 AM                53 hlink_a.txt
               2 File(s)             70 bytes
               2 Dir(s)  54,871,420,928 bytes free

C:\shared\forsale>type a.txt
"A larger string to knock the indexes out of sync"

C:\shared\forsale>
```

Unfortunately, there are few options available for detecting hard link paradoxes, but it is something to consider when the file scan shows **more usage** than the bitmap. Since we have the opposite situation here, hard links are not a significant factor.

Step 6.
Is Volume Shadow Copy Service maintaining diff area files for snapshots? Use VSSADMIN LIST SHADOWSTORAGE to find out. Add shadow storage to the running total.

VSSVC pre-allocates space for making volume snapshots. In order to support this feature, diff area files are kept in the "\System Volume Information" folder. This pre-allocated space is used to maintain point-in-time information for the "Previous Versions" feature and for the "System Restore" application. If you are the type of user who prefers to minimize the impact of such features, then you can resize your shadow storage with VSSADMIN so it has less impact on disk usage. I prefer to leave these features at their default settings (and just make a note of how much disk space it is using).



```
C:\>vssadmin list shadowstorage
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2005 Microsoft Corp.

Shadow Copy Storage association
   For volume: (C:)\\?\Volume{d1ec55f7-df23-11dc-a696-806e6f6e6963}\
   Shadow Copy Storage volume: (C:)\\?\Volume{d1ec55f7-df23-11dc-a696-806e6f6e6963}\
   Used Shadow Copy Storage space: 81.859 MB
   Allocated Shadow Copy Storage space: 2.93 GB
   Maximum Shadow Copy Storage space: UNBOUNDED


C:\>
```

Step 7.

If your numbers still don't make sense, then it's time to check for actively hidden files / folders. There are many different rootkit scanners that can help you identify the presence of actively hidden files & folders. Try using a scanner like **Rootkit Revealer**. If you suspect that your machine has been compromised by a rootkit, refer to **http://www.microsoft.com/uk/business/security/security-malware.mspx**.

Seeing that I did not have a significant amount of mysterious usage, I was satisfied that there was no malicious logic hiding my files. After taking into account all of the factors, there was only a 1.57GB difference remaining (which was accounted for by the contents of a user's recycle bin).

Whenever you see a difference between the "Used space", and "Size on disk", always ask yourself "What am I not seeing?"

Best regards,

Dennis Middleton *"The NTFS Doctor"*

Comments

NTFS Misreports Free Space?
4 Jul 2008 5:53 AM

PingBack from **http://www.ditii.com/2008/07/04/ntfs-misreports-free-space/**

Vince
5 Jul 2008 11:56 AM

Are you kidding me?!??? These are the steps you have to take?? I'm sure the average user will have no problem "piping" the output of STREAMS.EXE to a text file.

Microsoft should be embarassed by this, and shouldn't be surprised why more consumers are moving to the MAC OS.

[Alternate named streams are common to many OS platforms and each platform handles these a little differently (see URL).

http://en.wikipedia.org/wiki/Fork_%28filesystem%29

A common reason we have to examine named streams is because Mac clients can create large forked files on Windows shares and you need to keep track of their usage.

Windows does a great job of making forks & alternate named streams available to application developers on the various OS platforms, but IT professionals should understand how to see the effects of using this feature.

Regards, Dennis]

edgar
7 Jul 2008 4:26 PM

Thank you for this good article.

But this is only the beginning, for a nice transparent UI application.

Right ? :)

[That is left as an exercise for the reader. **This MSDN article** should help you get started. Or if you prefer .NET coding, check out **this article**.]

Matt Johnson's Technical Adventures
9 Jul 2008 8:38 AM

The list is a little longer today because of not posting last week. Enjoy! Microsoft Advanced Windows

David Walker
30 Sep 2008 11:56 AM

"High metadata usage will be the subject for part 2 of this blog."

Is that part coming any time soon?

Phylyp
1 Oct 2008 5:14 AM

I came across this article via a link from Raymond Chen's blog - a very nice coverage.

It is interesting to note that hard links are not synchronized - I need to go back and review my usage of hard links on my PCs to see if this might cause a problem...

Darrain Brown
1 Oct 2008 11:18 AM

> But this is only the beginning, for a nice transparent UI application

It's not free, but we use a software product called FolderSizes (**http://www.foldersizes.com**) to help us with this sort of problem. It seems to have much of this logic built into it, plus a nifty treemap view to boot. :-)

The Old New Thing
4 Mar 2009 10:27 AM

Because there is no need for it to keep track of that information.

virtual
1 Apr 2013 2:01 AM

sir ,

i only want to know where does NTFS stores the sum of free cluster on the drive / volume . is it somewhere in $Bitmap , boot Sector ,... etc.

[You can count the bits in the $Bitmap:$Data:”” attribute, but this only reliable on the volume if it is not mounted.]

Kottees
23 Sep 2013 4:55 PM

Dennis,

I just want to say a BIG thank you, Step 6 made my day. You just prevented an another outage. Thanks again.

Scott Magee
11 Apr 2014 3:04 AM

Excellent article, thanks!

My issue turned out to be 80Gb consumed by VSS although VSS was Disabled!!!

I had to enable it, reduce size allocated and then disable.

Thanks Again

Scott