



Understanding NTFS Hard Links, Junctions and Symbolic Links

Author: Conrad Chung, [2BrightSparks](#)

Introduction

Microsoft has implemented linking tools in its NTFS operating file systems since Windows NT 4.0 and Windows 2000. These links provide a convenient method for users to access their data. Over the years Microsoft continued to improve these tools which can now be used to link files and/or directories together. The latest form of file link was introduced in Windows Vista and continues to form part of their latest operating system. In this article we will look at what these file links are and how they differ from one another.

What are NTFS Hard Links, Junctions and Symbolic Links?

Hard links, junction points and symbolic links are linking processes used by Windows to associate other files, directories or volumes. One of the methods to create file links is to use the command line program **mklink**:

To create a hard link: *mklink /H linkName target*

To create a junction: *mklink /J linkName target*

To create a symbolic link: *mklink /D linkName target*

The following sections describes each type of file link in detail.

Hard Links

A hard link is a file that represents another file on the same volume without actually duplicating the data of that file. More than one hard link can be created to point at the same file. Hard links cannot link to a file that is on a different partition, volume or drive. Hard links on directories are not supported as it would lead to inconsistencies in parent directory entries.

Although a hard link is essentially a mirrored copy of the target file that it is pointing to, no additional hard drive space is required to store the hard link file. If a 1GB file is mirrored by 3 hard links, the total space used on the partition will only be 1GB instead of 4GB.

In addition, if any of the hard links or the original file(s) is/are deleted, the data will not be deleted and the rest of the other links will still be able to access it. Changes made to the data contents via any of the hard links or the original will be propagated to the rest of the other items automatically.

Hard links only work on Microsoft Windows operating systems that support NTFS partitions (Windows NT 4.0 or later) while FAT and ReFS file systems do not work with hard links.

An example of using hard links is when a user needs to have a file stored in two different folders. He could copy the file to the other folder and have two copies of the same file. However, twice the amount of storage space would be used. Also, if file contents of one file is changed, the other file will be outdated unless the newer file is copied over to replace it. Both issues could be solved with the use of hard links.

Junctions

Sometimes referred to as soft links, the function of a junction is to reference a target directory, unlike a hard link which points to a file. Junctions can be created to link directories located on different partitions or volume, but only locally on the same computer. It does this through the implementation of the NTFS feature called reparse points. Redirected targets in junctions are defined by an absolute path. An absolute path refers to a path which will contain the root element and the complete directory list that is required to locate the target. For example, `\Main\Folder\report` is an absolute path. All of the information required to locate the target is contained in the path string.

Like hard links, directory junctions do not take up additional space even though they are stored on the drive partition; their function is to point to the original files in the original directory. Thus, it should be noted that if the target is deleted, moved or renamed, all junctions which point to the target will break and continue to point to a non-existing directory. Content changes from any of the junction links or the target will automatically propagate to the rest.

Junctions are only compatible with Windows 2000 or later. An example in which junctions are often used is on Windows Vista, where the name “`C:\Documents and Settings`” is a junction that points to `C:\Users`. Thus, older programs that reference hard-coded legacy file paths can continue to work in Vista.

Symbolic Links

Symbolic links were recently introduced in Windows Vista/Windows Server 2008 or later. An NTFS symbolic link is a file system object that points to another file system object. In simpler terms, it is a more advanced type of shortcut. Symbolic links can point to any file or folder either on the local computer or using a SMB path to point at targets over a network (the target machine on the remote end needs to run Windows Vista or later). They do not use any disk space.

A symbolic link could use either a relative path or an absolute path to point to its target. A relative path has to be combined with another path in order to properly access the target file. For a detailed explanation between the difference of absolute and relative paths, please refer to this link:

[http://msdn.microsoft.com/en-us/library/aa363878\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/aa363878(v=vs.85).aspx)

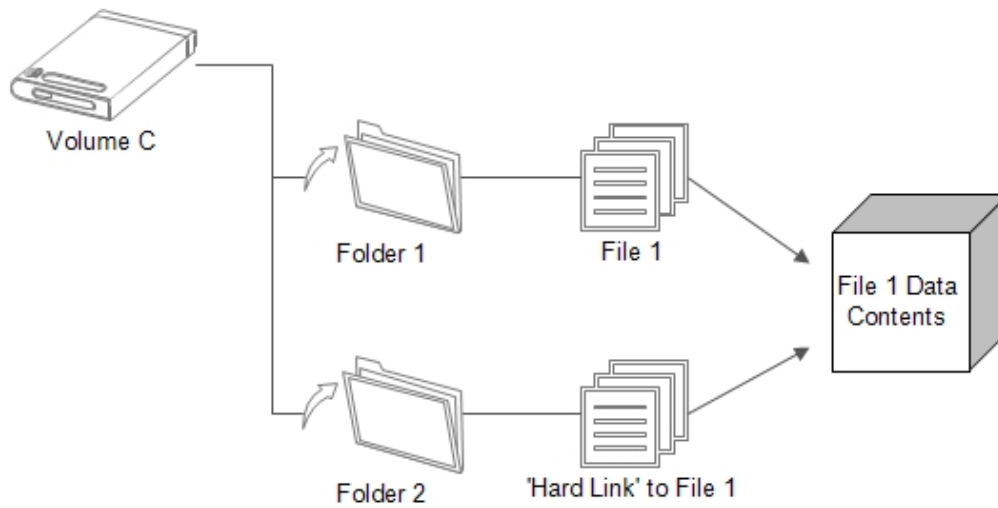
Symbolic links are transparent to users – they appear as normal files or directories. All applications will be able to recognize both the link and the target. Like junctions, symbolic links will become a stale link if the target is moved, renamed or deleted. The operating system does not check to see if the target exists.

Comparison table of hard link, junction and symbolic link

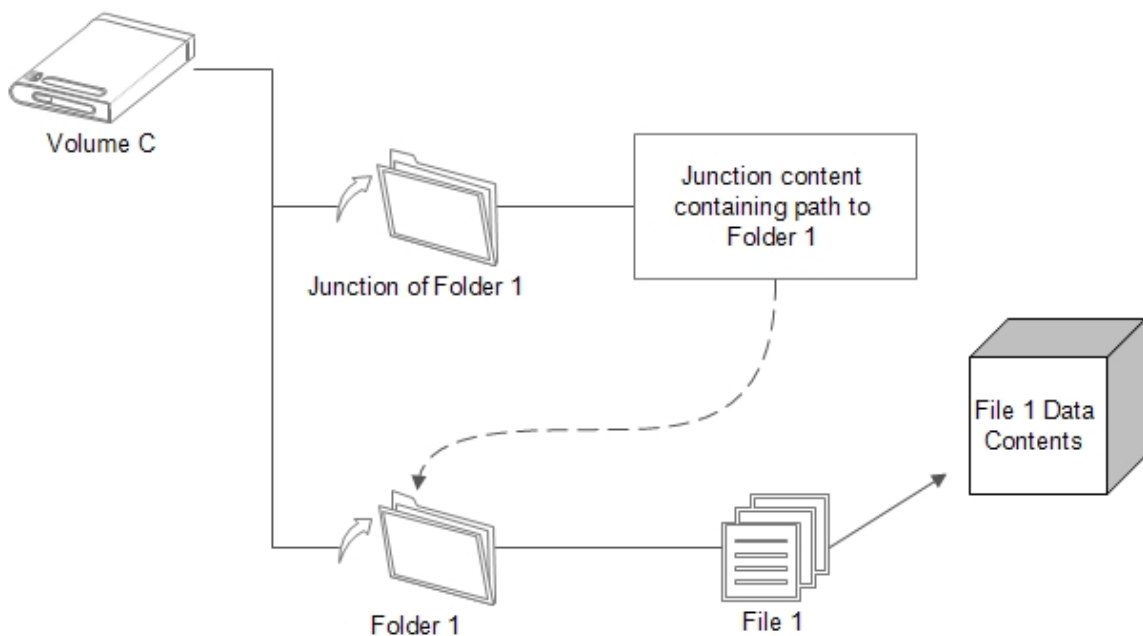
	Hard Link	Junction	Symbolic Link
Supported OS	Windows NT4 or later	Windows 2000 & Windows XP or later	Windows Vista or later
Storage requirement for target	Files must be on the same volume	Directories (paths) must be on local computer	Any path – local or remote, relative or absolute SMB file or path
When a hard link/junction/symbolic link is deleted using Windows Explorer...	The original file and other hard links pointing to it remains. If all associated links are removed, the data is deleted	Windows Vista or later: target is unchanged. Windows 2000, XP & 2003: target & sub-folders are deleted	Target is unchanged
The target is moved...	Hard link stays valid	Junction turns invalid	Symbolic link turns invalid
Relative path allowed?	Not applicable	Not allowed; path becomes absolute when saved	Allowed
Works on files?	Yes	No	Windows Vista or later
Works on folders?	No	Yes	Windows Vista or later

Examples of a hard link and junction

Hard Link



Junction



Configuration Settings in SyncBackSE & SyncBackPro that use file links

In the backup and synchronization programs – SyncBackPro/SE/Free, users are provided with some settings to choose how they want the program to handle these links if encountered during a profile run.

Expert > Copy/Delete > Folders > Ignore NTFS junction points (reparse points) – Configurable in all 3 models of SyncBack, this setting is recommended and turned on by default. Junctions and symbolic links are ignored during profile runs with it enabled, but hard links are not ignored. This setting supports both NTFS and ReFS file systems. It should be noted that if this setting is disabled, SyncBackPro/SE/Free could copy the files stored within the two different paths (the original path and the link path) twice and create two real folders of those names on the Destination.

Expert > Copy/Delete > Advanced > Copy symbolic links as-is instead of copying the file the link points to – When this setting is enabled, the program will copy only the symbolic link and not the file that the link points to. This means the link on the destination will be pointing to the same file the source link is pointing to. Thus users who enable this setting need to ensure the symbolic links are using relative paths instead of absolute paths. This setting is only available in SyncBackSE and SyncBackPro. Pre-requisites to use this setting are: This option is only available on Windows Vista and newer, when the Standard Windows File Copy method is being used, and is only used when copying between NTFS or ReFS file systems. Please also note that this setting will not work (and is disabled) over FTP, Cloud, To Email, etc. and the profile cannot have compression enabled as symbolic links must be stored ‘natively’ on the drive.

Summary

Now you have the basic knowledge on how hard links, junctions and symbolic links work in a NTFS Windows operating system, you are hopefully in a better position to recognize issues relating to them.