HFS Plus

HFS Plus or **HFS+** (also known as **Mac OS Extended** or **HFS Extended**) is a journaling file system developed by Apple Inc. It replaced the Hierarchical File System (HFS) as the primary file system of Apple computers with the 1998 release of Mac OS 8.1. HFS+ continued as the primary Mac OS X file system until it was itself replaced with the Apple File System (APFS), released with macOS High Sierra in 2017. HFS+ is also one of the formats used by the iPod digital music player.

Compared to its predecessor HFS, also called *Mac OS Standard* or *HFS Standard,* HFS Plus supports much larger files (block addresses are 32-bit length instead of 16-bit) and using Unicode (instead of Mac OS Roman or any of several other character sets) for naming items. Like HFS, HFS Plus uses B-trees to store most volume metadata, but unlike most other file systems, HFS Plus supports hard links to directories. HFS Plus permits filenames up to 255 characters in length, and n-forked files similar to NTFS, though until 2005 almost no system software took advantage of forks other than the data fork and resource fork. HFS Plus also uses a full 32-bit allocation mapping table rather than HFS's 16 bits, improving the use of space on large disks.

Contents

History

HFS+	
Developer(s)	Apple Inc.
Full name	Hierarchical File System Plus
Introduced	January 19, 1998 with Mac OS 8.1
Partition identifier	Apple_HFS (Apple Partition Map)
	0xAF (MBR) HFS and HFS+
	Apple_HFSX (Apple Partition Map) when HFSX 48465300-0000-11AA-AA11-00306543ECAC (GPT)
Structures	
Directory contents	B-tree
File allocation	Bitmap
Bad blocks	B-tree
Limits	
Max. volume size	8 exabyte[1]
Max. file size	8 EB[2]
Max. number of files	4,294,967,295 ($2^{32} - 1$)
Max. filename length	255 characters (255 UTF-16 encoding units, normalized to Apple-modified variant of Unicode Normalization Format D)
Allowed characters in filenames	Unicode, any character, including NUL. OS APIs may limit some characters for legacy reasons
Features	
Dates recorded	access, attributes modified, backed up, contents modified, created
Date range	January 1, 1904 – February 6, 2040
Date	1 s

Codenamed *Sequoia* in development,[4] HFS+ was introduced with the January 19, 1998, release of Mac OS 8.1.[2]

With the release of the Mac OS X 10.2.2 update on November 11, 2002, Apple added optional journaling features to HFS Plus for improved data reliability. These features were accessible through the GUI, using the Disk Utility application in Mac OS X Server, but only accessible through the command line in the standard desktop client.[5]

With Mac OS X v10.3, all HFS Plus volumes on all Macs were set to be journaled by default. Within the system, an HFS Plus volume with a journal is identified as *HFSJ*.

Mac OS X 10.3 also introduced another version of HFS Plus called *HFSX*. HFSX volumes are almost identical to HFS Plus volumes, except that they are never surrounded by the HFS Wrapper that is typical of HFS Plus volumes and they optionally support case sensitivity for file and folder names. HFSX volumes can be recognized by two entries in the Volume Header, a value of HX in the signature field and 5 in the version field.[2]

resolution	
Forks	Yes
Attributes	Color (3 bits, all other flags 1 bit), locked, custom icon, bundle, invisible, alias, system, stationery, inited, no INIT resources, shared, desktop
File system permissions	Unix permissions, NFSv4 ACLs (Mac OS X v10.4 onward)
Transparent compression	Partial (decmpfs, on Mac OS X 10.6 and higher)[3]
Transparent encryption	Yes (on Mac OS X 10.7 and up). Per-home directory encryption is available with AES using HFS+-formatted .dmg volumes on OS X versions prior to 10.7 but later than Mac OS X 10.3
Other	
Supported operating systems	Mac OS 8.1, Mac OS 9, macOS/iOS/tvOS/watchOS/Darwin, Linux, Microsoft Windows (through Boot Camp IFS drivers)

Mac OS X 10.3 also marked Apple's adoption of Unicode 3.2 decomposition, superseding the Unicode 2.1 decomposition used previously. This change caused problems for developers writing software for Mac OS X.[6]

Mac OS X 10.3 introduced a number of techniques that are intended to avoid fragmentating files in HFS+.[7]

With Mac OS X 10.4, Apple added support for Inline Attribute Data records, something that had been a part of the Mac OS X implementation of HFS Plus since at least 10.0, but always marked as "reserved for future use".[8] Until the release of Mac OS X Server 10.4, HFS Plus supported only the standard UNIX file system permissions; however, 10.4 introduced support for access control list–based file security, which provides a richer mechanism to define file permissions and is also designed to be fully compatible with the file permission models on other platforms such as Microsoft Windows XP and Windows Server 2003.[9]

In Mac OS X Leopard 10.5, directory hard-linking was added as a fundamental part of Time Machine.

In Mac OS X Snow Leopard 10.6, HFS+ compression was added using Deflate (Zlib). In open source and some other areas this is referred to as AppleFSCompression or decmpfs. Compressed data may be stored in either an extended attribute or the resource fork. When using non-Apple APIs, AppleFSCompression is not always completely transparent.[10] OS X 10.9 introduced two new algorithms: LZVN (libFastCompression), and LZFSE.

In Mac OS X Lion 10.7, logical volume encryption (known as FileVault 2) was added to the operating system. This addition to the operating system in no way changed the logical structure of the file system. Apple's logical volume manager is known as Core Storage and its encryption at the volume level can apply to file systems other than HFS Plus. With appropriate hardware, both encryption and decryption should be transparent.

Design

HFS Plus volumes are divided into sectors (called logical blocks in HFS), that are usually 512 bytes in size. These sectors are then grouped together into allocation blocks which can contain one or more sectors; the number of allocation blocks depends on the total size of the volume. HFS Plus uses a larger value to address allocation blocks than HFS, 32 bits rather than 16 bits; this means it can access 4,294,967,296 ($= 2^{32}$) allocation blocks rather than the 65,536 ($= 2^{16}$) allocation blocks available to HFS.[2] When disks were small, this was of little consequence, but as larger-capacity drives became available, it meant that the smallest amount of space that any file could occupy (a single allocation block) became excessively large, wasting significant amounts of space. For example, on a 1 GB disk, the allocation block size under HFS is 16 KB, so even a 1-byte file would take up 16 KB of disk space. HFS Plus's system greatly improves space utilization on larger disks as a result.

File and folder names in HFS Plus are also encoded in UTF-16[11] and normalized to a form very nearly the same as Unicode Normalization Form D (NFD)[12] (which means that precomposed characters like "å" are decomposed in the HFS+ filename and therefore count as two code units[13] and UTF-16 implies that characters from outside the Basic Multilingual Plane also count as two code units in an HFS+ filename). HFS Plus permits filenames up to 255 UTF-16 code units in length.

Formerly, HFS Plus volumes were embedded inside an HFS standard file system. This was phased out by the Tiger transition to Intel Macs, where the HFS Plus file system was not embedded inside a wrapper. The wrapper had been designed for two purposes; it allowed Macintosh computers without HFS Plus support in their ROM to boot HFS Plus volumes and it also was designed to help users transition to HFS Plus by including a minimal HFS volume with a read-only file called *Where_have_all_my_files_gone?*, explaining to users with versions of Mac OS 8.0 and earlier without HFS Plus, that the volume requires a system with HFS Plus support. The original HFS volume contains a signature and an offset to the embedded HFS Plus volume within its volume header. All allocation blocks in the HFS volume which contain the embedded volume are mapped out of the HFS allocation file as *bad blocks*.[2]

Notable among file systems used for Unix systems, HFS Plus does not support sparse files.

There are nine structures that make up a typical HFS Plus volume:[2]

1. Sectors 0 and 1 of the volume are HFS **boot blocks**. These are identical to the boot blocks in an HFS volume. They are part of the HFS wrapper.[14]
2. Sector 2 contains the *Volume Header*, which is equivalent to the Master Directory Block in an HFS volume. The Volume Header stores a wide variety of data about the volume itself, for example the size of allocation blocks, a timestamp that indicates when the volume was created or the location of other volume structures such as the Catalog File or Extent Overflow File. The Volume Header is always located in the same place.
3. The *Allocation File* which keeps track of which allocation blocks are free and which are in use. It is similar to the Volume Bitmap in HFS, in which each allocation block is represented by one bit. A zero means the block is free and a one means the block is in use. The main difference with the HFS Volume Bitmap, is that the Allocation File is stored as a regular file – it does not occupy a special reserved space near the beginning of the volume. The Allocation File can also change size and does not have to be stored contiguously within a volume.
4. The *Catalog File* is a B-tree that contains records for all the files and directories stored in the volume. The HFS Plus Catalog File is very similar to the HFS Catalog File, the main differences being records are larger to allow more fields and to allow for those fields to be larger (for example to allow the longer 255-character unicode file names in HFS Plus). A record in the HFS Catalog File is 512 bytes in size; a record in the HFS Plus Catalog File is 4 KB in

the classic Mac OS and 8 KB in macOS. Fields in HFS are of fixed size, while in HFS Plus the size can vary depending on the actual size of the data they store.

5. The *Extents Overflow File* is another B-tree that records the allocation blocks that are allocated to each file as extents. Each file record in the Catalog File is capable of recording eight extents for each fork of a file; once those are used additional extents are recorded in the Extents Overflow File. Bad blocks are also recorded as extents in the Extents Overflow File. The default size of an extent record in the classic Mac OS is 1 KB and 4 KB in macOS.

6. The *Attributes File* is a new B-tree in HFS Plus that does not have a corresponding structure in HFS. The Attributes File can store three different types of 4 KB records: *Inline Data Attribute* records, *Fork Data Attribute* records and *Extension Attribute* records. Inline Data Attribute records store small attributes that can fit within the record itself. Fork Data Attribute records contain references to a maximum of eight extents that can hold larger attributes. Extension Attributes are used to extend a Fork Data Attribute record when its eight extent records are already used.

7. The *Startup File* is designed for non-Mac OS systems that lack HFS or HFS Plus support. It is similar to the Boot Blocks of an HFS volume.

8. The second-to-last sector contains the *Alternate Volume Header*, which is equivalent to the Alternate Master Directory Block of HFS. Note that this is the second-to-last-sector for the disk, not the volume; if the disk is larger than the volume, the AVH will be outside the range of the filesystem.

9. The last sector in the volume is reserved for use by Apple. It is used during the computer manufacturing process.[2]

Criticisms

HFS Plus lacks several features considered staples of modern file systems like ZFS and NTFS.[15] Data checksums are the most routinely cited missing feature.[16]

Besides checksumming, features of modern file systems that HFS+ lacks include:

- nanosecond timestamps[16]
- concurrent access (that is, more than one process can access the filesystem at the same time)[16]
- snapshotting[16]
- Support for dates beyond February 6, 2040[17]
- sparse file support[16]
- a better implementation of hard links (in other filesystems, these are typically multiple directory entries pointing to the same data blocks; hard links in macOS are implemented as small files that are stored in a special hidden directory)[16]

HFS Plus was not designed for Unix-like systems, so features such as file system permissions and hard links had to be retrofitted when Apple moved to Mac OS X.[16]

Other operating systems

Linux

The Linux kernel includes the hfsplus module[18] for mounting HFS+ filesystems read-write. HFS+ fsck and mkfs have been ported to Linux and are part of the hfsprogs package.[19]

In 2009, these drivers were diagnosed to be corrupting HFS+ drives with a capacity greater than 2 TB.[20] Consequently, Linux distributions such as Debian and Ubuntu stopped allowing mounting of HFS+ drives or partitions greater than 2 TB.[21] As of February 2011, work is in progress to lift this restriction.[22]

Under Linux's current HFS+ driver, journaling must be disabled in order to write data safely onto an HFS+ partition. Provided the partition isn't being used by Apple's Time Machine software, journaling can be disabled under macOS:[23] Using Disk Utility in OS X Yosemite, the user may hold Alt/Option and click "Disable Journaling" on the File menu, having first selected a mounted partition.

An HFS+ partition with journaling enabled may be forcibly mounted with write access under Linux, but this is unsupported and unwise.[23][24]

A Google Summer of Code project to implement write support to journaled HFS+[25] was accepted by the Linux Foundation in 2011 but was not completed at that time and is still a work in progress. Progress and improvements to the HFS+ driver, including some updates to journaling support, are posted on the linux-fsdevel mailing list[26] from time to time.

As of July 2011, Paragon Software Group provided kernel drivers that allow full read-write access to HFS+ journaled volumes.[27] The product is a proprietary implementation of HFS+ based on Paragon's proprietary UFSD library. There are both free and paid editions of the driver, and they include a utility for checking and repairing HFS+ volumes. According to the online documentation (free version[28] or the paid edition[29]), both the free edition and the paid edition currently support Linux kernels from 2.6.36 up to 4.12.x. Ubuntu, Debian, Fedora, OpenSUSE and CentOS are the only Linux distributions officially supported.[30]

Windows

As of May 2012, Apple has only released read-only HFS+ drivers for Windows XP, Windows Vista, and Windows 7 as part of the Boot Camp software in Mac OS X 10.6.[31] This means users on these systems can read data on the HFS+ drive, but not write to them. Microsoft has created an HFS+ driver for the Xbox 360 mainly for the purpose of reading HFS+-formatted iPods.

A free and opensource software – jHFSplus, based on HFSExplorer and jpfm – can be used to mount hfs/hfs+ partitions as read-only virtual folders.[32]

A freeware plugin for Total Commander exists, that can read, among others, HFS and HFS+ filesystems.[33]

DiskInternals Linux Reader can be used to extract/save folders/files out of HFS and HFS+ Hard Drives/Partitions.[34]

A commercial product, MacDrive, is also available for mounting HFS and HFS+ drives, optical discs, and other media in Windows Explorer, and allows both reading and writing to the volume, as well as repairing and formatting Mac disks.[35]

A commercial product, Paragon's HFS+ for Windows allows full read and write and disk management from all versions of Windows from Windows XP to Windows Server 2008.[36]

Cross-platform

A free (GPL) alternative is HFSExplorer written by Erik Larsson.[37] HFSExplorer is a Java application for viewing and extracting files from an HFS+ volume (Mac OS Extended) or an HFSX volume (Mac OS Extended, Case-sensitive). The volume can be located either on a physical disk, in various Apple disk image

and sparse disk image formats, or a raw file system dump. However, HFSExplorer is a read-only solution; it cannot write to HFS-formatted volumes.

See also

- Comparison of file systems
- Apple File System – replaced HFS+ in 2017
- HFS

References

1. "OS X: Mac OS Extended format (HFS Plus) volume and file limits" (http://support.apple.com/kb/HT2422). Support.apple.com. 2008-07-29. Archived (https://web.archive.org/web/201007300 00158/http://support.apple.com/kb/ht2422) from the original on 2010-07-30. Retrieved 2010-07-05.
2. "Technical Note TN1150: HFS Plus Volume Format" (https://developer.apple.com/legacy/library/technotes/tn/tn1150.html). *Apple Developer Connection*. March 5, 2004. Archived (https://web.archive.org/web/20130622040013/https://developer.apple.com/legacy/library/technotes/tn/tn1150.html) from the original on June 22, 2013. Retrieved 2007-03-28.
3. "10.6: Compress files with HFS+ compression – Mac OS X Hints" (http://hints.macworld.com/article.php?story=20090902223042255). Archived (https://web.archive.org/web/2012040103561 4/http://hints.macworld.com/article.php?story=20090902223042255) from the original on 2012-04-01.
4. Gregg Williams (May 19, 2000). "Strategy Mosaic: Sequoia--Mass Storage for Tomorrow and Beyond" (https://web.archive.org/web/20000519230946/http://devworld.apple.com/mkt/informe d/appledirections/sep97/stratmosaic.html). Archived from the original (http://devworld.apple.co m/mkt/informed/appledirections/sep97/stratmosaic.html) on May 19, 2000.
5. John Gruber (November 11, 2002). "Crow" (http://daringfireball.net/2002/11/crow). *DaringFireball.net*. Archived (https://web.archive.org/web/20070409153459/http://daringfireball. net/2002/11/crow) from the original on April 9, 2007. Retrieved 2007-03-28.
6. "Re: git on MacOSX and files with decomposed utf-8 file names" (https://web.archive.org/web/2 0110315014244/http://kerneltrap.org/mailarchive/git/2008/1/23/593749/thread). KernelTrap. 2010-05-07. Archived from the original (http://kerneltrap.org/mailarchive/git/2008/1/23/593749/t hread) on March 15, 2011. Retrieved 2010-07-05.
7. "Fragmentation in HFS Plus Volumes" (https://web.archive.org/web/20200103202947/https://os xbook.com/software/hfsdebug/fragmentation.html). *osxbook.com*. Archived from the original (htt ps://osxbook.com/software/hfsdebug/fragmentation.html) on January 3, 2020. Retrieved 10 January 2020.
8. John Siracusa (April 28, 2005). "OS X 10.4 Tiger: Metadata revisited" (https://arstechnica.com/r eviews/os/macosx-10.4.ars/6). *Ars Technica*. Archived (https://web.archive.org/web/200704050 40437/http://arstechnica.com/reviews/os/macosx-10.4.ars/6) from the original on April 5, 2007. Retrieved 2007-03-28.
9. "Apple – OS X Server – Windows Services" (https://web.archive.org/web/20071231203318/htt p://www.apple.com/br/server/macosx/windowsservices.html). *Apple.com*. Archived from the original (https://www.apple.com/br/server/macosx/windowsservices.html) on 2007-12-31. Retrieved 2007-11-12.
10. iohead LLC (2011). "fileXray user guide and reference" (https://filexray.com/fileXray.pdf) (PDF). iohead LLC. p. 31. Retrieved 14 July 2020.
11. "What is Mac OS HFS+ format?" (https://www.applexsoft.com/glossary/hfs-plus.html). *AppleXsoft*. Archived (https://web.archive.org/web/20180824144248/http://www.applexsoft.co m/glossary/hfs-plus.html) from the original on 2018-08-24. Retrieved 2018-11-06.

12. "Technical Q&A QA1235: Converting to Precomposed Unicode" (https://developer.apple.com/qa/qa2001/qa1235.html). *Apple Developer Connection*. February 7, 2003. Archived (https://web.archive.org/web/20080518105836/http://developer.apple.com/qa/qa2001/qa1235.html) from the original on May 18, 2008. Retrieved 2007-03-28.

13. There are some minor differences derived from the fact that the HFS Plus format was finalized before Unicode had standardized the NFD format (see "Unicode Subtleties" (https://developer.apple.com/legacy/library/technotes/tn/tn1150.html#UnicodeSubtleties) Archived (https://web.archive.org/web/20130622040013/https://developer.apple.com/legacy/library/technotes/tn/tn1150.html#UnicodeSubtleties) 2013-06-22 at the Wayback Machine for more information)

14. "Boot Blocks" (https://developer.apple.com/library/archive/documentation/mac/Files/Files-101.html#HEADING101-0). *Inside Macintosh*. Apple Inc. July 2, 1996.

15. Harris, Robin. "WWDC's biggest disappointment – ZDNet" (https://www.zdnet.com/article/wwdcs-biggest-disappointment/). Archived (https://web.archive.org/web/20151022030417/http://www.zdnet.com/article/wwdcs-biggest-disappointment/) from the original on 2015-10-22.

16. John Siracusa (July 20, 2011). "Mac OS X 10.7 Lion: the Ars Technica review" (https://arstechnica.com/apple/2011/07/mac-os-x-10-7/12/). Ars Technica. Archived (https://web.archive.org/web/20161222061714/http://arstechnica.com/apple/2011/07/mac-os-x-10-7/12/) from the original on 22 December 2016. Retrieved 18 January 2017.

17. "Mac OS X – ForensicsWiki" (http://forensicswiki.org/wiki/Mac_OS_X#HFS.2FHFS.2B_date_and_time_values). *forensicswiki.org*. Archived (https://web.archive.org/web/20170410052933/http://forensicswiki.org/wiki/Mac_OS_X#HFS.2FHFS.2B_date_and_time_values) from the original on 2017-04-10. Retrieved 2017-04-09.

18. "kernel/git/torvalds/linux.git – Linux kernel source tree" (https://archive.today/20120709100649/http://git.kernel.org/?p=linux/kernel/git/torvalds/linux-2.6.git;a=tree;f=fs/hfsplus;hb=HEAD). Archived from the original (https://git.kernel.org/?p=linux/kernel/git/torvalds/linux-2.6.git;a=tree;f=fs/hfsplus;hb=HEAD) on 2012-07-09.

19. "Debian – Details of package hfsprogs in sid" (http://packages.debian.org/sid/hfsprogs). Archived (https://web.archive.org/web/20090703165004/http://packages.debian.org/sid/hfsprogs) from the original on 2009-07-03.

20. "kernel/git/torvalds/linux.git – Linux kernel source tree" (https://archive.today/20120711142712/http://git.kernel.org/?p=linux/kernel/git/torvalds/linux-2.6.git;a=commitdiff;h=5c36fe3d87b3f0c85894a49193c66096a3d6b26f). *git.kernel.org*. Archived from the original (https://git.kernel.org/?p=linux/kernel/git/torvalds/linux-2.6.git;a=commitdiff;h=5c36fe3d87b3f0c85894a49193c66096a3d6b26f) on 2012-07-11. Retrieved 2016-10-14.

21. "#550010 – hfsplus corrupts filesystems >2TB – Debian Bug report logs" (http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=550010). Archived (https://web.archive.org/web/20160108163119/https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=550010) from the original on 2016-01-08. Retrieved 2011-02-17.

22. "[PATCH 2/3] hfsplus: lift the 2TB size limit — Linux Filesystem Development" (http://www.spinics.net/lists/linux-fsdevel/msg42242.html). Archived (https://web.archive.org/web/20110718213210/http://www.spinics.net/lists/linux-fsdevel/msg42242.html) from the original on 2011-07-18.

23. "How to mount a HFS partition in Ubuntu as Read/Write?" (http://superuser.com/questions/84446/how-to-mount-a-hfs-partition-in-ubuntu-as-read-write). Archived (https://web.archive.org/web/20111130084754/http://superuser.com/questions/84446/how-to-mount-a-hfs-partition-in-ubuntu-as-read-write) from the original on 2011-11-30.

24. Banks, Wyatt (2007-07-31), *Macintosh HFSPlus Filesystem for Linux* (https://archive.today/20120722061818/http://git.kernel.org/?p=linux/kernel/git/torvalds/linux.git;a=blob_plain;f=Documentation/filesystems/hfsplus.txt;hb=60fd4d6a1953accd3d57f8e4f3b0f4692598bf4e), archived from the original (https://git.kernel.org/?p=linux/kernel/git/torvalds/linux.git;a=blob_plain;f=Documentation/filesystems/hfsplus.txt;hb=60fd4d6a1953accd3d57f8e4f3b0f4692598bf4e) on 2012-07-22

25. "Implement HFSPlus Journal on Linux" (https://web.archive.org/web/20110405082701/http://w
    ww.google-melange.com/gsoc/proposal/review/google/gsoc2011/naota/1). Archived from the
    original (http://www.google-melange.com/gsoc/proposal/review/google/gsoc2011/naota/1) on
    April 5, 2011.
26. " 'hfsplus' in linux-fsdevel – MARC" (https://marc.info/?l=linux-fsdevel&w=2&r=1&s=hfsplus&q=
    b). Archived (https://web.archive.org/web/20160108163118/http://marc.info/?l=linux-fsdevel&w
    =2&r=1&s=hfsplus&q=b) from the original on 2016-01-08. Retrieved 2012-09-21.
27. "Overview – Paragon NTFS&HFS+ for Linux 9.5 Express" (http://www.paragon-software.com/h
    ome/ntfs-linux-per/). Archived (https://www.webcitation.org/61BaL6EAz?url=http://www.paragon
    -software.com/home/ntfs-linux-per/) from the original on 2011-08-24. Retrieved 2012-06-29.
28. "Full Features – Paragon NTFS/HFS+ for Linux 9.5 Express" (http://www.paragon-software.co
    m/home/ntfs-linux-per/features.html). Archived (https://web.archive.org/web/20120903064043/h
    ttp://www.paragon-software.com/home/ntfs-linux-per/features.html) from the original on 2012-
    09-03. Retrieved 2012-09-21.
29. "Full features – Paragon NTFS&HFS+ for Linux 9.5 Professional" (http://www.paragon-softwar
    e.com/business/ntfs-linux-professional/features.html). Archived (https://web.archive.org/web/20
    121202071435/http://www.paragon-software.com/business/ntfs-linux-professional/features.htm
    l) from the original on 2012-12-02. Retrieved 2012-09-21.
30. "FAQ & Glossary – Paragon NTFS&HFS+ for Linux 9.5 Professional" (http://www.paragon-soft
    ware.com/home/ntfs-linux-professional/faq.html). Archived (https://web.archive.org/web/201606
    07171924/http://www.paragon-software.com/home/ntfs-linux-professional/faq.html) from the
    original on 2016-06-07. Retrieved 30 June 2016.
31. "Apple – Support – Downloads – Bootcamp" (https://support.apple.com/downloads/bootcamp).
    apple.com. Archived (https://web.archive.org/web/20161019152644/https://support.apple.com/d
    ownloads/bootcamp) from the original on 2016-10-19. Retrieved 2016-10-18.
32. "jHFSplus" (https://sourceforge.net/projects/jhfsplus/). Shashank Tulsyan. 2011-06-29.
    Archived (https://web.archive.org/web/20110709042442/http://sourceforge.net/projects/jhfsplu
    s/) from the original on 2011-07-09.
33. "DiskInternals Reader for Total Commander" (http://www.diskinternals.com/reader-for-tc/).
    DiskInternals. 2011-03-16. Archived (https://web.archive.org/web/20111116030040/http://www.
    diskinternals.com/reader-for-tc) from the original on 2011-11-16. Retrieved 2011-11-23.
34. "DiskInternals Linux Reader" (http://www.diskinternals.com/linux-reader/). DiskInternals.
    Archived (https://web.archive.org/web/20120406183418/http://www.diskinternals.com/linux-rea
    der) from the original on 2012-04-06. Retrieved 2016-10-18.
35. "MacDrive 9 Standard" (https://web.archive.org/web/20130921020510/http://www.mediafour.co
    m/products/macdrive/standard). Mediafour. 2013-09-19. Archived from the original (http://www.
    mediafour.com/products/macdrive/standard) on 2013-09-21. Retrieved 2013-09-19.
36. "Write HFS+ volumes in Windows – Paragon HFS+ for Windows" (http://www.paragon-softwar
    e.com/home/hfs-windows/). Archived (https://web.archive.org/web/20161013093423/https://ww
    w.paragon-software.com/home/hfs-windows/) from the original on 2016-10-13. Retrieved
    2016-10-14.
37. "HFSExplorer" (http://www.catacombae.org/hfsexplorer/). *Catacombae*. Erik Larsson. 2015-10-
    13. Archived (https://web.archive.org/web/20180924203323/http://www.catacombae.org/hfsexpl
    orer/) from the original on 2018-09-24. Retrieved 2018-09-23.

External links

- hfsexplorer (http://www.catacombae.org/hfsx.html) – Open Source Explorer for MS-Windows®
  written in Java, with support for HFS, HFS+, and HFSX
- Apple "legacy" Technote 1150 (https://developer.apple.com/legacy/library/technotes/tn/tn1150.
  html) – HFS Plus Volume Format

- Apple "legacy" Technote 1189 (https://developer.apple.com/legacy/library/technotes/tn/tn1189.html) – The Monster Disk Driver Technote
- hfsdebug (http://www.osxbook.com/software/hfsdebug/) – A debugger for HFS Plus volumes by Amit Singh
- hfsprogs (http://packages.debian.org/sid/hfsprogs) – Userspace support tools for HFS+ filesystems under Linux systems (adapted from Apple's native tools).
- iBored (http://apps.tempel.org/iBored/) – A disk editor and viewer supporting HFS Plus
- MacDrive (http://www.mediafour.com/products/macdrive) – A commercial product for reading, writing and creating HFS and HFS+ volumes in Windows
- HFSleuth (http://newosxbook.com/index.php?page=downloads#HFSleuth) – A free command line tool allowing the inspection of HFS+/HFSX volumes and Disk Images (DMGs) for macOS and Linux
- Mac OS X 10.7 Lion: the Ars Technica review (https://arstechnica.com/apple/2011/07/mac-os-x-10-7/12/) – A criticism about this file system

---