IJARESM
INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN
ENGINEERING, SCIENCE & MANAGEMENT

COMPARATIVE STUDY OF TWO MODERN FILE SYSTEMS: NTFS AND HFS+

Viral H. Panchal[1], Brijal Panchal[2], Heta K. Desai[3]
Asst. professor, Computer Engg., S.N.P.I.T&RC, Umrakh, Gujarat, India[1]
Student, Science and Tech., University of Westminster, London, England, United Kingdom[2]
Asst. professor, Computer Engg., S.N.P.I.T&RC, Umrakh, Gujarat, India[3]

Abstract: This paper will include comparative analysis based on their individual features of their system as well as what they're equipped with and to know their limitations. Analyses of what are the common features in both the file systems, their circumstances and under which application they would be best over the other ones. I will also talk about several categories such as compatibility, portability, efficiency, and security features. Conclusion of this report will show which file system is more powerful and progressive to be use in future operating systems. There will be a brief introduction to the main network file system protocols. NFS is an essentially an application that is between a client and a server, therefore allowing user to remotely update and store files as if they were using there own PCs.

Keywords: **file system, operating system, NTFS, HFS+, Windows, mac OS**

1. INTRODUCTION

File systems are a fundamental part of modern operating systems. The term *filesystem* has somewhat two different meanings, both of which are commonly used. One meaning is the entire hierarchy of directories (referred to as a *directory tree), which* is used to organize and provides information about all the files in the system. Second meaning is the collection of files (*type* of filesystem) this is how the storage of data (i.e. Files, Folders, etc.) is organized.

There are few basic key functions that every file systems incorporates, i.e. Efficiency, organization of data for quick storage and retrieval, file operations like copy, move, create, rename, and delete, provision to boot up the file systems, apart from these basic functions there are additional functions which are provided by some file systems such as compression, encryption, file streams and others.

There are so many different types of file systems; they are designed differently according the need of particular operating systems. This report will be consisting of arguably two of the most common file systems, which are NTFS and HFS+. New Technology File System (NTFS) used by Microsoft Windows Based OS and HFS+ (Hierarchical File System Plus) is used by Apple OS X systems [1].

This report will include comparative analysis based on their individual features of their system as well as what they're equipped with and to know their limitations. Analyses of what are the common features in both the file systems, their circumstances and under which application they would be best over the other ones. I will also talk about several categories such as compatibility, portability, efficiency, and security features. Conclusion of this report will show which file system is more powerful and progressive to be use in future operating systems.

There will be a brief introduction to the main network file system protocols. NFS is an essentially an application that is between a client and a server, therefore allowing user to remotely update and store files as if they were using there own PCs. Furthermore I will be explaining how the network file system protocols relate to and work with NTFS & HFS+.

2. WHAT IS NTFS (NEW TECHNOLOGY FILE SYSTEMS)?

NTFS is a standard file system of Microsoft operating systems starting from Windows NT/2000; it is also the most common file system on end-user computers. NTFS has several improvements over last File system of Microsoft operating systems [2]. Such as improved support for metadata and advanced data structure, which provides a combination of performance, reliability and compatibility that are not found in any other file systems. It is designed to quickly perform standard file operations such as read, write and search; on a very large hard disk it can even perform an advanced operations such as file-system recovery. To protect anything that comes in contact with Windows NT Security model is used by NTFS. This security could be implemented for a file by establishing a security descriptor as a file record attributes in MFT (Master File Table). MFT is very vital part of the NTFS structures, all data stored on a volume is contained in the MFT.

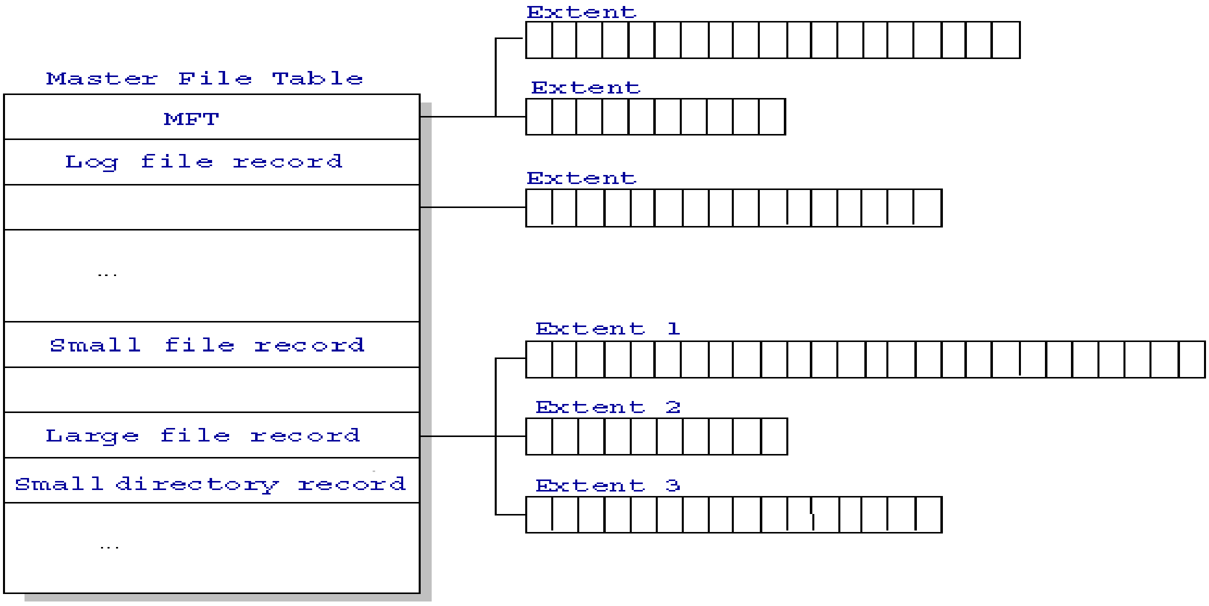


Figure.1 MFT (Master File Table) [6]

As you can see the Figure 1, NTFS uses MFT as a database to store information about every file and directory.

Some of the Main Feature of NT File Systems is:

- Journaling – Record all the changes in the files and directories
- Compression – Compression of partition, folders and files
- Sparse Files – Permits programs to create very large files but to consumer disk space only as needed.
- Volume Shadow Copy – By copying old, newly overwritten data, it allows to keep historical versions of file and folders on NTFS volume as a shadow copy.
- Hard Links – Represent of a target file on the same storage drive but it only acts like one.

- Disk Quotas – Provide more precise control of network-based storage.
- Encryption File Systems. – File and directory level encryption

3. WHAT IS HFS+ (HIERARCHICAL FILE SYSTEMS PLUS)?

HFS+ is the file system found on Mac OS 8.1 and later (Including Mac OS X) developed by Apple Inc. to replace their Hierarchical File System (HFS) as the primary file system used on Apple computers [3]. HFS+ uses 32 bit block addresses, which helps in supporting the larger files. HFS couldn't support more than 65536 allocation blocks due to 16 bit allocation mapping table, for reliable data integrity and availability.

There are Five Special Files, which are implemented in this HFS+ system compare to their old ones.

- Catalog File - Organized as a B-Tree for Fast and Efficient Searches
- Extent Overflow File - Additional extents are store in a B-Tree in this file
- Allocation File - Specifying whether an allocation block is free or not
- Attribute File - Contains attribute information regarding particular files or folders
- Startup File - Allows computers to boot that have built in support for HFS+

Next Chapter will consist of full explanation of their features and comparison.

4. COMPARISON OF TWO FILE SYSTEMS

4.1 Features of File Systems

One of the Main functions in NTFS is called Journaling, which helps NTFS files systems to record all the changes that happen in files and their directories. If there was a sudden system crash as after this function will increase its reliability and recoverability which will helps in to restore the data in the fast and very efficient way.

The benefits of Journaling in NTFS are outweighing the small negative performance penalty. In addition to that when performing small operations the speed difference is minimal and if the hard drive is being used as a server only then become noticeable, over the years hard drives have become significantly larger and the extra protection provided by journaling which helps in recover from most data corruption scenarios.

However Apple enhanced HFS+ by adding Metadata Journaling, at the beginning stages when it first released the journaling can be manually enabled or disabled by the user, but since October 2013 when Mac OS X v10.3 released, it enables the journal by default. The whole purpose of adding Journaling to HFS+ was to address the problem when there is premature removal of an external disk or flash drive or a power outage is when the file system may be interrupted while critical updates are in progress.

HFS+ uses Catalog file structure, which stores data records of files and directories in the volume, B-tree is used for managing the catalog records and thus browsing that catalog B-tree can retrieve directory contents. Each files and directories are marked by CNID (Catalog Node ID), which contains four types of entries as *File Thread Record, File Record, Directory Thread Record and Directory Record.*

However the major disadvantage of HFS+ is that all the file and directory records are stored in the Catalog file data structure, which means during multitasking all program will get into the queues if single program writes into this structure and it will lead to issues which will lower the performance grade of the file system. HFS+ doesn't provide reliability and it involves high risk by this means if the Catalog file structure gets destroyed the whole file system comes down and performance issues taking into account as in this present day files involves high volume of data storage capabilities.

Both the systems support hard links. Hard link is a file, which represents another file on the same volume without actually duplicating the data of that file [5]. In NTFS by creating hard links you can use the same file name as the original file but appear in different folders, use different file names from the original file but appear in the same folder and use different file name from the original file and appear in different folders. So in short it allows there to be single file in several locations without copying the files. However in HFS+ it has the ability of a directory and file structure to use pointers, which are associated with a file name and points to the address of the physical file. So for example when the file is been copied and renamed, then these two files point to the same address.

One of the features, which are used in NTFS, it’s a Security method and it is supported by the use of an Access Control List. Simply ACL’s are used within NTFS to see and control which user are allowed to access different files or folders.

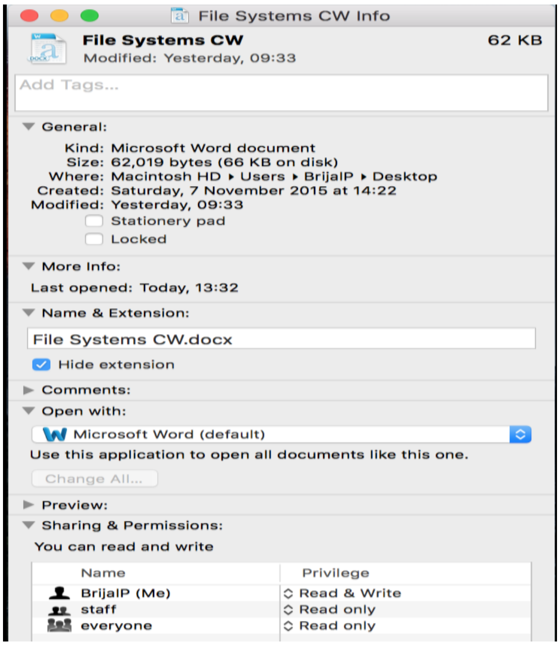
4.2 Comparison based on Storage Limits

Storage limits in Both NTFS and HFS+ are almost similar. They both have Maximum filename length of 255 bytes. However the maximum volume size are different for both the file systems, In NTFS Maximum Volume Size is 16EB whereas in HFS+ it’s slightly less than 8 EB, not only this if we look at the Maximum file size NTFS is superior then HFS+ because Maximum file size in NTFS is 16EB compare to 8EB of maximum file size in HFS+.[4]

What it means is that NTFS has a slight advantage here as it has more capacity to handle very large number of files and folders compare to HFS+.

4.3 Comparison based on File System Storage Metadata and Characteristics

Both the file systems are very similar when it comes to storing different types of metadata. To name few types such as author, last edited, created, file owner etc.



File Systems CW Info

File Systems CW 62 KB
Modified: Yesterday, 09:33

Add Tags...

General:
Kind: Microsoft Word document
Size: 62,019 bytes (66 KB on disk)
Where: Macintosh HD ▸ Users ▸ BrijalP ▸ Desktop
Created: Saturday, 7 November 2015 at 14:22
Modified: Yesterday, 09:33
□ Stationery pad
□ Locked

More Info:
Last opened: Today, 13:32

Name & Extension:
File Systems CW.docx
☑ Hide extension

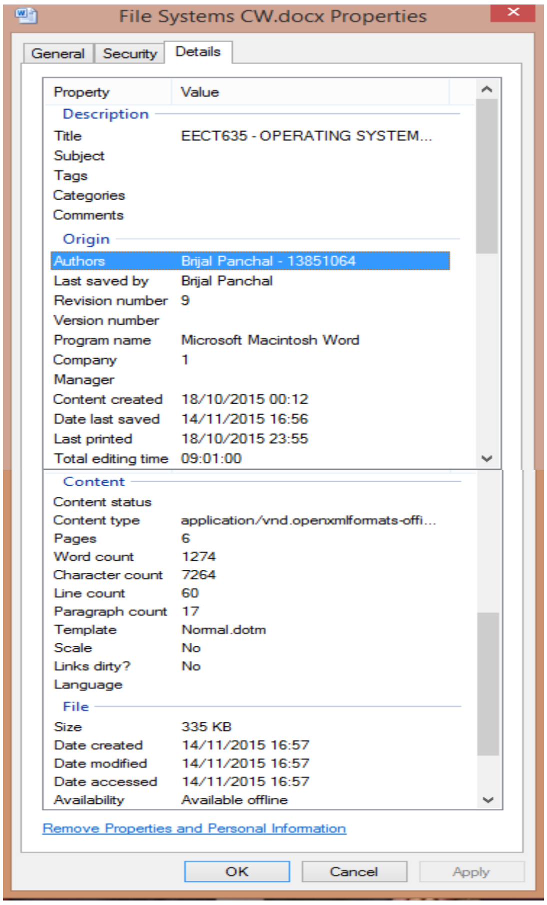
Comments:

Open with:
Microsoft Word (default)
Use this application to open all documents like this one.
Change All...

Preview:

Sharing & Permissions:
You can read and write

Name	Privilege
BrijalP (Me)	Read & Write
staff	Read only
everyone	Read only



File Systems CW.docx Properties

Property	Value
Description	
Title	EECT635 - OPERATING SYSTEM...
Subject	
Tags	
Categories	
Comments	
Origin	
Authors	Brijal Panchal - 13851064
Last saved by	Brijal Panchal
Revision number	9
Version number	
Program name	Microsoft Macintosh Word
Company	1
Manager	
Content created	18/10/2015 00:12
Date last saved	14/11/2015 16:56
Last printed	18/10/2015 23:55
Total editing time	09:01:00
Content	
Content status	
Content type	application/vnd.openxmlformats-offi...
Pages	6
Word count	1274
Character count	7264
Line count	60
Paragraph count	17
Template	Normal.dotm
Scale	No
Links dirty?	No
Language	
File	
Size	335 KB
Date created	14/11/2015 16:57
Date modified	14/11/2015 16:57
Date accessed	14/11/2015 16:57
Availability	Available offline

As you can see in the Figure 1 and Figure 2

below which shows Metadata information in Mac OS and Windows OS.
Figure. 2 Mac OS
Figure. 3 Windows OS
If we look at the characteristics of these two file systems, NTFS is much better in more aspect. Lets look at the similarities first, File change log are both available in the system which means they both keep the record of the changes that's been made in data or metadata changes such as namespaces of files or directories. NTFS has case sensitive whereas HFS+ only has partially case sensitive, which means mix of capital and small letters when it comes to naming a file or directories.

NTFS provides a very strong Encryption to its disk partition; EFS (Encryption File System) uses symmetric encrypt data. It stores your data in a coded format that only you can read. The beauty of encryption is that it is very effortless and invisible to you as the authorized owner. Widows will automatically encrypt your files even before you store them and decrypt it again whenever you want to read or modify any of your files. As you can see in the figure.1 and figure.2 which shows two different pic of file information, it shows clearly that the files are well protected and only the author (Main User) is permissioned to 'Read and Write' function while others are 'Read Only' which means no modification can be made. This is what gives NTFS an extra edge over HFS+.

4.4. Compatibility, Portability and Efficiency

In terms of file systems, Compatibility is a measure to see how many different operating systems a file system can work on. There are some file systems such as zettabyte file systems (ZFS), which can operate on systems such as Linux or Solaris, however they cannot operate on Microsoft NT making it compatible on only certain operating systems. Clearly a file system that is more compatible in more operating systems would be a better choice for developers to choose.

Staring with HFS+, this system is the only system that works on time machine, this time machine is a backup software application distributed by Apple OS X but while HFS+ is the best way to format drives for use on Macs, Windows does not support it. If an external drive will be use such as a USB with certain PCs, in this case it is highly recommended to use an application called MacDrive. When MacDrive is installed on a Windows PC, it will make it possible to read & write to HFS+ drives.

Comparing this directly to Apple OS X system, NTFS is only partly compatible with Mac OS X. Macs can read files on NTFS drives only, though it cannot write to them. Getting such files from a PC to a Mac would work fine as normal, however to work the other way from Mac to PCs there problems moving files in that way.

In terms of other file systems, NTFS is a file system that can be used on many different operating systems for example these systems include, Windows 9x, Windows NT, DOS, Linux, Mac OS, Mac OSX and more. These can use NTFS by normally or they can have a full separate kernel to use it. On the other side, HFS+ is less adaptable in comparison to NTFS. OS X can create and use the following partition structure, which are Apple partition map, GUID partition scheme and also the master boot record (Non-boot volume). HFS+ can create, read and write to systems FAT32 and ExFAT. However this can only read from FAT32, ExFAT and also NTFS. This shows that HFS+ is fairly exclusive system which is mainly designed to be compatible and suitable by an 'Apple' based operating system. Although it can read from NTFS, that is all it can do but it cannot be possible to work from one another. Again comparing these two on compatibility, NTFS is a lot more compatible to use. This is especially suitable if for example a running an organization whereby there are multiple operating systems that in use working between each other. Using HFS+ as file system would ultimately limit users by only working on few operating systems.

Portability in terms computer science is the measure of how easy an application can be transferred from one computer environment to another. A computer software application is considered portable to a new environment if the effort required to adapt it to the new system is within practical limits.

This section below would cover how efficient the following file systems are in terms of managing disk space. This is a key factor to consider when choosing what sort file systems to use for an operating system. Mainly this is how well the system would manage its available space to reduce waste or unused space. For example, let’s consider a normal hard disk with a memory space of 4GB, and we want to sort various types of files ranging from 2KB to about 10KB, we want to see how the files are stored and which techniques are used. There are instances where by a hard disk again with 4GB of memory would be full yet the user has only saved about 3.5GB. This example showing a huge waste of 0.5GB of space, this result is due to how file systems operate and manage files.

Beginning here with NTFS, this particular file system works in a way where by when a hard disk is formatted (initialized), it is divided into partitions or major divisions of the total physical hard disk space. Within each partition, the operating system keeps track of all the files that are stored by that operating system. Each file is actually stored on the hard disk in one or more clusters or disk spaces of a predefined uniform size. Using NTFS, the sizes of clusters range from 512 bytes to 64 kilobytes. Windows NT provides a recommended default cluster size for any given drive size. [1]. This means that beforehand NTFS would set a fixed cluster, for example a cluster size could be like 4KB. In this case if there is a file that is in size 3.8KB it would fit into one cluster with a usage efficiency of 95% which is fairly good, however if there is another file that is 4.2KB in size, this would take up two clusters that in total would take up 8KB of space. Here the usage efficiency would be 52.2%, which is almost shows almost one whole cluster of space is wasted which is very negative. In comparison HFS+ works slightly differently, this system works with 512 byte sectors that are grouped into allocation blocks. Allocation blocks are disk blocks that are in use currently or active.

5. MAIN NETWORK FILE SYSTEM PROTOCOLS

The Network File System (NFS) is a client-server application which allows a computer user to view, store and update files on a remote computer as they were doing on its own local computer.

It is a one kind of distributed file system protocol originally designed by Sun Microsystems in 1984. It is an open standard defined in RFCs, which allows anyone to implement it.

The user's system needs to have an NFS client and the other computer needs the NFS server. Both the systems require that you also have TCP/IP installed since the NFS server and client use TCP/IP as the program that sends the files and updates back and forth. However, the User Datagram Protocol (UDP), which comes with TCP/IP, is used instead of TCP with older versions of NFS.

The NFS protocol uses the Remote Procedure Call (RPC) method of communication between computers.

Using NFS, a system administrator or a user can mount all or a portion of a file system. The portion of file system that is mounted can be accessed with any privileges (read-only or read-write) go with your access to each file.

The NFS protocol defines a set of operations that a server must support. These operations are reading and writing files, accessing file attributes, searching for a file within a directory, reading a set of directory links, and manipulating links and directories. All these operations are implemented as RPCs.

NFS uses the virtual file system (VFS) layer to handle local and remote files. VFS provides standard file system interface, which hides the difference between accessing local and remote file systems.

Design Goals of NFS

- Operating System (OS) independence and interoperability
- Simple crash recovery for clients and servers
- Transparent access (client programs don't know files are remote)
- Maintain local file system semantics
- Reasonable performance

NFS Versions

NFSv1 (NFS Version 1) was used by Sun only for in-house experimental purposes. When the development team added substantial changes to NFS version 1, they decided to release it in the newer version, i.e. NFS version 2 (NFSv2).

NFSv2 (NFS version 2) was defined in RFC 1094 in 1989 and it was originally operated over UDP.

The main focus of this version was to keep the server side ***staleless*** (no open and close operations, server must check permission on each read and write call) with locking implemented outside of the core protocol.

NFSv3 (NFS version 3) was defined in RFC 1813 in 1995. The major changes from NFSv2 to NFSv3 are:

- Sizes and offsets are widened from 32 bits to 64 bits
- Support for asynchronous writes on the server to improve the write performance
- Additional file attributes in many replies, to avoid the need to re-fetch them
- The 8KB data size limitation on the READ and WRITE procedures is relaxed

NFSv4 (NFS version 4) was defined in RFC 3010 in 2000 and revised in RFC 3530 in2003. This version includes performance improvements, mandates strong security, and introduces a ***stateful*** (open and close operations are provided server checks permission at file open time) protocol. Version 4 became the first version developed with the Internet Engineering Task Force (IETF) after Sun Microsystems handed over the development of the NFS protocols.

NFSv4.1 (NFS version 4.1) was defined in RFC 5661 in 2010. In this version the clients may now access storage devices directly and in parallel. It also eliminates the classic NFS bottleneck of having only one server.

WebNFS is an extension to NFSv2 and NFSv3, which allows NFS to integrate more easily into web browsers and to enable operation through firewalls. WebNFS offers what Sun believes is a faster way to access web pages and other Internet files.

7. CONCLUSION

In this report I have done a research on NTFS and HFS+ widely popular file systems, while doing so I have understood both the file systems and their feature on a more deeper level and by comparing NTFS and HFS+, I have come to a conclusion that even NTFS is not as efficient or as faster as HFS+ but NTFS is widely and very common file system, even though NTFS has very minimum requirements to run a file system, its being used very frequently as a journaling file system. When it comes to security feature of file systems here NTFS has an edge over HFS+, also Compatibility wise NTFS is much better then HFS+. One of the biggest advantage if NTFS is that it can support large volumes of data which helps to manage the on growing data efficiency. So overall it shows that NTFS would be the best choice of

REFERENCES

[01] [1] http://forensicswiki.org/wiki/HFS%2B

[02] [2] https://support.microsoft.com/en-us/kb/100108

[03] [3] http://www.linux.org/threads/hierarchical-file-system-plus-hfs.4493/

[04] [4]http://www.researchgate.net/publication/267298416_Feature_Based_Comparison_of_Modern_File_Systems

[05] [5]http://www.2brightsparks.com/resources/articles/NTFS-Hard-Links-Junctions-and-Symbolic-Links.pdf

[06] [6] http://ntfs.com/images/NTFS-MFT-structure.gif - Figure 1

[07] [7] http://iit.qau.edu.pk/books/OS_8th_Edition.pdf