International Journal of Advance Research in Science and Engineering
Volume No. 10, Issue No. 07, July 2021
www.ijarse.com
IJARSE
ISSN 2319 - 8354

FLASH SEARCH

Manvi Pandey, Parvinder Singh Saini, Naseem Siddiqui, Vivek Mishra

Computer Science & Engineering, Buddha Institute of Technology, Gorakhpur, India

ABSTRACT

Desktop search tools search within a user's own computer files as opposed to searching the Internet. Desktop search products are software alternatives to the search software included in the operating system, helping users sift through desktop files. Traditionally, Windows Search recursively scans folders in order to locate a file which is very slow and time consuming. In order to overcome this issue, we have implemented a different way of searching which uses the MFT file present in NTFS file systems. The NTFS file system contains a file called the master file table, or MFT. There is at least one entry in the MFT for every file on an NTFS file system volume. All information about a file, including its size, time and date stamps, permissions, and data content, is stored in MFT. The benefit of searching using the MFT file is that instead of scanning the entire drive for locating a file, we only search for the record of that file in the MFT file. We are performing only one operation on the hard disk (bringing the MFT into RAM). The actual search takes place in the memory which makes the searching instantaneous. We used Python for making this application.

Keywords *– MFT, NTFS, Python, Windows*

1. INTRODUCTION

For a long time, searching has been one of the things windows users have complained about relentlessly. And for good reason. In our own testing, on an SSD, coupled with the 8300H processor, and 8 GiB of RAM, searching for a simple resume file stored in the default library folder ‘Documents’ took more than 10 seconds to bring up a meaningful result, and even longer to actually find the file that we wanted. When we are doing mission critical work and every second is important, such long wait times are unacceptable and unbelievably frustrating.

Flash Search solves this issue of slow searches and introduces some cool tricks to make searching files as easy and convenient as possible. The application has a very simple, clean, and easy to use interface. The moment you open the application, you are presented with a list of all the files in your system already and a search box for you to type in your queries. The search box is activated by default and brings instant results on every single keypress. These results can then easily be sorted through one of their displayed attributes in order for you to find your files faster.

Flash Search also introduces some handy syntax such as wildcards and boolean operators. This makes it possible to do things such as combining the results of two different searches into one with the help of the boolean or

operator, or to find all files that do not contain a specific substring. Wildcards can be used in order to make up for any or a specific number of characters missing in a query. Results can also be filtered on the basis of the substring they start or end with. All of these individual syntax tools can be easily combined with each other and make Flash Search even more useful.

2. BACKGROUND

Windows Search (also known as Instant Search) is a content index desktop search platform by Microsoft introduced in Windows Vista as a replacement for both the previous Indexing Service of Windows 2000 and the optional MSN Desktop Search for Windows XP and Windows Server 2003, designed to facilitate local and remote queries for files and non-file items in compatible applications including Windows Explorer. It was developed after the postponement of WinFS and introduced to Windows constituents originally touted as benefits of that platform.

Windows Search creates a locally managed Index of files — documents, emails, folders, programs, photos, tracks, and videos — and file contents, as well as of non-file items including those of Microsoft Outlook for which users can perform incremental searches based on details such as authors, contents, dates, file names, file types, people, and sizes; the Index stores actual prose from inside documents and metadata properties from other content. Control Panel and Settings can also be searched.

Windows Search was introduced in Windows Vista as a replacement for the previous Indexing Service to facilitate data discovery and management, promote greater rapidity of search results, and to unify desktop search platforms across Microsoft Windows; it was also available as an optional download for Windows XP. There have been several updates to its functionality since its introduction, and it is also included in Windows 7, Windows 8, Windows 8.1, and Windows 10.

The problem with an indexing mechanism that indexes content is that while it makes the search more powerful in theory, it is still a special use case. Most people are looking to find files by name. A common example would be to find a file called ‘My Resume.pdf’, one could type in ‘resume’. Resumes don’t have the word ‘resume’ in them, they only have it in their file name. However, Windows will keep looking through the contents of file for the smallest of searches. It is not optional. While content search may be useful, it sure should not be the default.

3. METHODOLOGY

In order to overcome the issue of slow search in Windows, we have implemented a different way of searching which uses the MFT file present in NTFS file systems. NTFS or New Technology File System has been the default file system on Windows since Windows NT 3.1. The NTFS file system contains a file called the master file table, or MFT. There is at least one entry in the MFT for every file on an NTFS file system volume, including the MFT itself. All information about a file, including its size, time and date stamps, permissions, and data content, is stored either in MFT entries, or in space outside the MFT that is described by MFT entries.

Whenever a search query is received, Flash Search looks for a record of the file directly in the MFT file. The benefit of searching using the MFT file is that instead of scanning the entire drive for locating a file, we only search for the record of that file in the MFT file. Hard Disks are very slow and each operation performed on them is expensive. By limiting our search to just the MFT file, we are performing only one operation on the hard disk (bringing the MFT into RAM). The actual search takes place in the memory which makes the searching instantaneous.

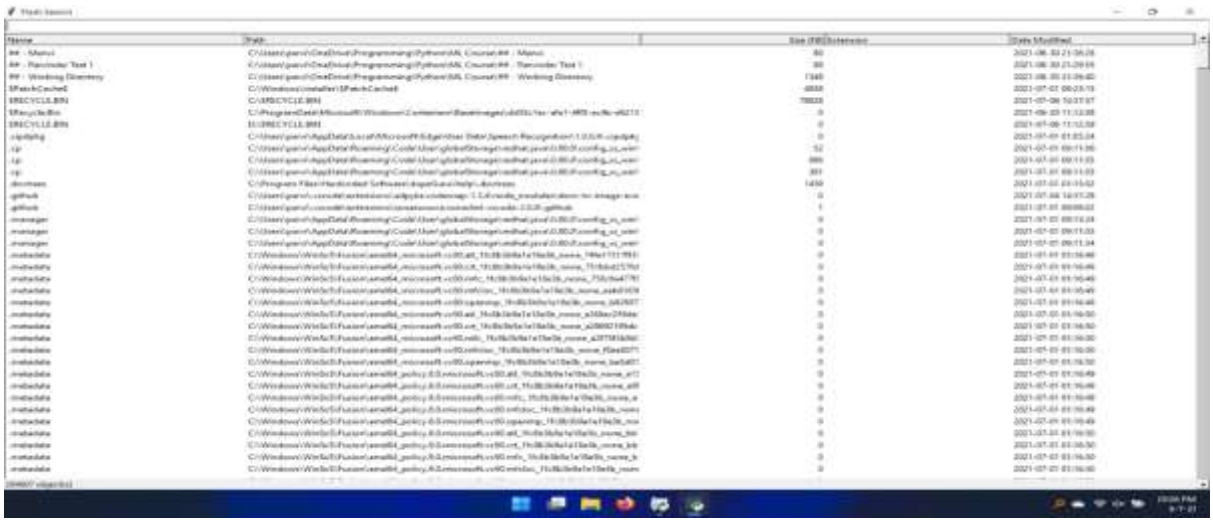
We then use a Python TKinter Treeview to display all results in a neat and clean manner which can then be easily sorted using any one of the displayed attributes of the results.

4. RESULTS

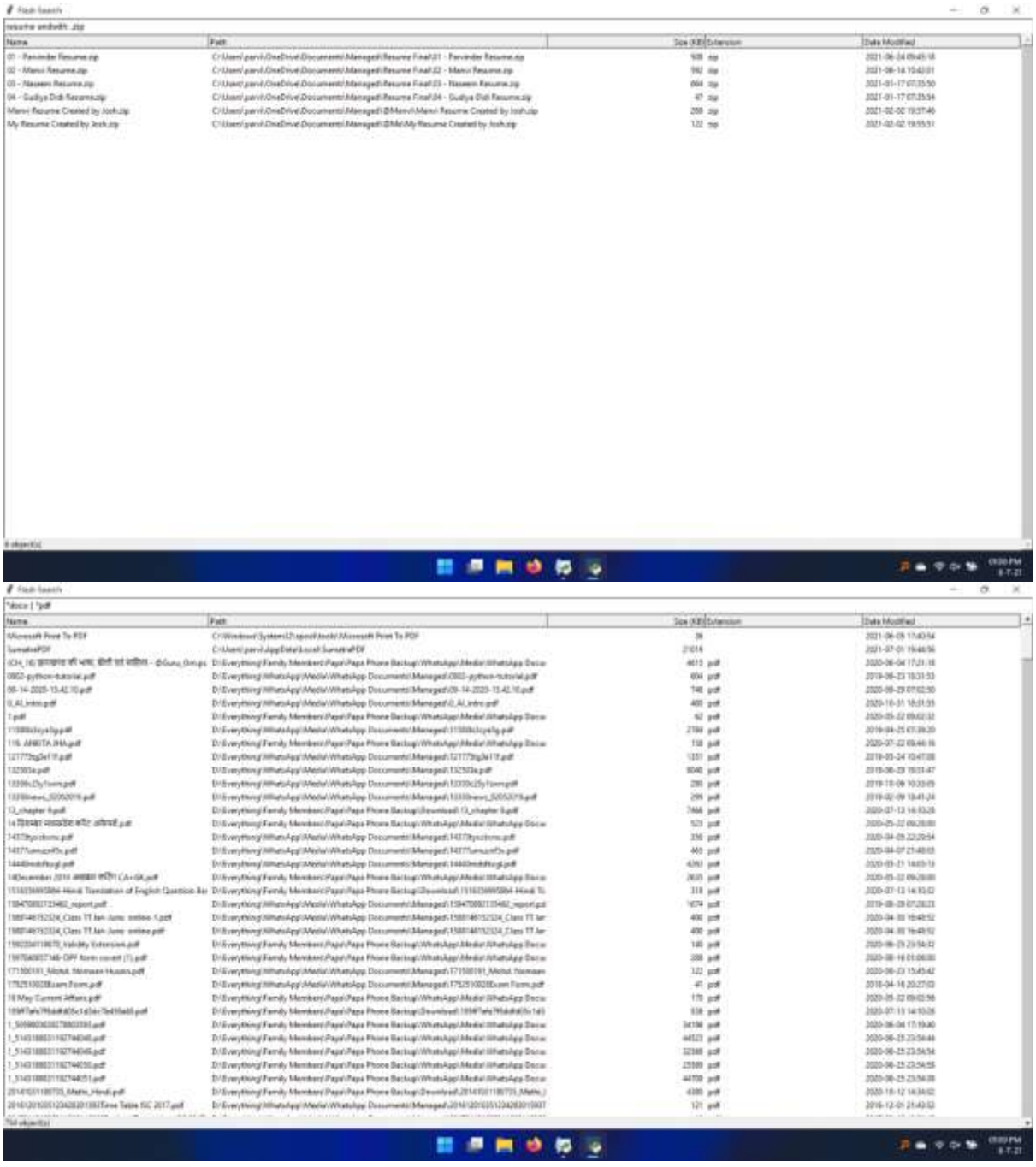
We began this paper by stating how slow Windows Search was in our own testing. We tested our own application on the same machine and the results were very simple:

i.) All queries returned results in an instant
ii.) All wildcards and boolean operators returned results in an instant
iii.) All mixing of different search syntaxes returned results in an instant
iv.) Sorting through any displayed attribute returned results in an instant

This shows that our methodology works exactly as expected and brings desired results in desired time.



International Journal of Advance Research in Science and Engineering
Volume No. 10, Issue No. 07, July 2021
www.ijarse.com
IJARSE
ISSN 2319 - 8354



5. CONCLUSION

To summarise, Flash Search does not try to be a replacement to Windows Search, but to overcome all of its common limitations such as limiting search to filenames only, allowing wildcards, allowing the use of substring-based search etc.

Apart from concurring these limitations, Flash Search also includes handy ways to make the searching experience so seamless that you basically forget about it. It can be said that Flash Search gives you what's yours in the fastest time possible whenever you ask for it.

Flash search is fast, but does not sacrifice functionality in order to reach that goal. We initially said that content-based search should be an option and that is exactly what we have done. Using a simple predefined syntax, you can instruct Flash Search to have a look inside the files in order to find what you are looking for. While this process of searching the content is slow due to the content not being indexed, the scope of a content-based search can easily be defined in order to speed up the process or make it instant.

6. ACKNOWLEDGEMENTS

The successful completion of this project marks the beginning of an ever going learning experience of converting ideas and concepts into real life, practical systems. This project was quite a learning experience for us at each and every step. At the same time, it has given us the confidence to work in a professional setup. We feel the experience gained during the project will lead us to gain a bright prospect in the future. First of all, we would like to give thanks to our guide, Mr. Vivek Kumar Mishra, for their active support and continuous guidance without which it would have been difficult for us to complete this project. We would also like to thank our respected HOD, Mr. Manish Gupta for their support.