FindNextFile function

Continues a file search from a previous call to the **FindFirstFile**, **FindFirstFileEx**, or **FindFirstFileTransacted** functions.

Syntax

C++

```
BOOL WINAPI FindNextFile(
  _In_  HANDLE             hFindFile,
  _Out_ LPWIN32_FIND_DATA lpFindFileData
);
```

Parameters

hFindFile [in]
The search handle returned by a previous call to the **FindFirstFile** or **FindFirstFileEx** function.

lpFindFileData [out]
A pointer to the **WIN32_FIND_DATA** structure that receives information about the found file or subdirectory.

Return value

If the function succeeds, the return value is nonzero and the *lpFindFileData* parameter contains information about the next file or directory found.

If the function fails, the return value is zero and the contents of *lpFindFileData* are indeterminate. To get extended error information, call the **GetLastError** function.

If the function fails because no more matching files can be found, the **GetLastError** function returns **ERROR_NO_MORE_FILES**.

Remarks

This function uses the same search filters that were used to create the search handle passed in the *hFindFile* parameter. For additional information, see **FindFirstFile** and **FindFirstFileEx**.

The order in which the search returns the files, such as alphabetical order, is not guaranteed, and is dependent

on the file system. If the data must be sorted, the application must do the ordering after obtaining all the results.

> **Note** In rare cases or on a heavily loaded system, file attribute information on NTFS file systems may not be current at the time this function is called. To be assured of getting the current NTFS file system file attributes, call the **GetFileInformationByHandle** function.

The order in which this function returns the file names is dependent on the file system type. With the NTFS file system and CDFS file systems, the names are usually returned in alphabetical order. With FAT file systems, the names are usually returned in the order the files were written to the disk, which may or may not be in alphabetical order. However, as stated previously, these behaviors are not guaranteed.

If the path points to a symbolic link, the **WIN32_FIND_DATA** buffer contains information about the symbolic link, not the target.

In Windows 8 and Windows Server 2012, this function is supported by the following technologies.

Technology	Supported
Server Message Block (SMB) 3.0 protocol	Yes
SMB 3.0 Transparent Failover (TFO)	Yes
SMB 3.0 with Scale-out File Shares (SO)	Yes
Cluster Shared Volume File System (CsvFS)	Yes
Resilient File System (ReFS)	Yes

Transacted Operations

If there is a transaction bound to the file enumeration handle, then the files that are returned are subject to transaction isolation rules.

Examples

For an example, see **Listing the Files in a Directory**.

Requirements

Minimum supported client	Windows XP [desktop apps | Windows Store apps]
Minimum supported server	Windows Server 2003 [desktop apps | Windows Store apps]
Minimum supported phone	Windows Phone 8
Header	FileAPI.h (include Windows.h); WinBase.h on Windows Server 2008 R2, Windows 7, Windows Server 2008, Windows Vista, Windows Server 2003, and Windows XP (include Windows.h)
Library	Kernel32.lib
DLL	Kernel32.dll
Unicode and ANSI names	**FindNextFileW** (Unicode) and **FindNextFileA** (ANSI)

See also

File Management Functions
FindClose
FindFirstFile
FindFirstFileEx
GetFileAttributes
SetFileAttributes
Symbolic Links
WIN32_FIND_DATA

© 2016 Microsoft