Query information about an attached file system (local or remote), or about a character device or pseudo-character device attached to the file system.

Syntax

DosQFSAttach (DeviceName, Ordinal, FSAInfoLevel, DataBuffer, DataBufferLen, Reserved)

Parameters

;DeviceName (PSZ) - input : Address of a drive letter or the name of a character or pseudo-character device. If DeviceName is a drive, it is an ASCIIZ string having the form of drive letter followed by a colon. If DeviceName is a character or pseudo-character device name, its format is that of an ASCIIZ string in the format of an OS/2 file name in a subdirectory called \DEV\. This parameter is ignored if level 2 or 3 is specified for FSAInfoLevel. ;Ordinal (USHORT) - output : Index into the list of character or pseudo-character devices, or the set of drives. Ordinal always starts at 1. The Ordinal position of an item in a list has no significance at all. Ordinal is used strictly to step through the list. The mapping from Ordinal to item is volatile and may change from one call to DosQFSAttach to the next. This parameter is ignored if level 1 is specified for FSAInfoLevel. ;FSAInfoLevel (USHORT) - input : Level of information returned in DataBuffer: ::Level 0001H returns data for the specific drive or device name referred to by DeviceName. The Ordinal field is ignored. ::Level 0002H returns data for the entry in the list of character or pseudo-character devices selected by Ordinal. The DeviceName field is ignored. ::Level 0003H returns data for the entry in the list of drives selected by Ordinal. The DeviceName field is ignored. ;DataBuffer (PBYTE) - output : Address of the return information buffer, has the following format: ::iType (USHORT) Type of item. :::1 - Resident character device :::2 - Pseudo-character device :::3 - Local drive :::4 - Remote drive attached to FSD. ::cbName (USHORT) : Length of item name, not counting null. ::szName (UCHAR) : Item name, ASCIIZ string. ::cbFSDName (USHORT) : Length of FSD name, not counting null. ::szFSDName (UCHAR) : Name of FSD item is attached to, ASCIIZ string. ::cbFSAData (USHORT) : Length of FSD Attach data returned by FSD. ::rgFSAData (UCHAR) : FSD Attach data returned by FSD. :Note: The szFSDName is the FSD name exported by the FSD, which is not necessarily the same as the FSD name in the boot sector. :For local character devices (iType = 1), cbFSDName = 0 and szFSDName contains only a terminating NULL byte, and cbFSAData = 0. :For local drives (iType = 3), szFSDName contains the name of the FSD attached to the drive at the time of the call. This information changes dynamically. If the drive is attached to the kernel's resident file system, szFSDName contains FAT or unknown. Since the resident file system gets attached to any disk that other FSDs refuse to mount, it is possible to have a disk that does not contain a recognizable file system, but yet gets attached to the resident file system. In this case, it is possible to detect the difference, and this information would help programs in not destroying data on a disk that was not properly recognized. ;DataBufferLen (PUSHORT) - input/output : Address of the byte length of the return buffer. Upon return, it is the length of the data returned in DataBuffer by the FSD. This field must be initialized. ;Reserved (ULONG) - input : Reserved, must be set to zero.

Return Code

;rc (USHORT) - return:Return code descriptions are: * 0 NO_ERROR * 15 ERROR_INVALID_DRIVE *111 ERROR_BUFFER_OVERFLOW *124 ERROR_INVALID_LEVEL *259 ERROR_NO_MORE_ITEMS

Remarks

Information about all block devices and all character and pseudo-character devices is returned by this call.

The information returned by this call is highly volatile. Calling programs should be aware that the returned information may have already changed by the time it's returned to them.

The information returned for disks that are attached to the kernel's resident file system can be used to determine if the kernel definitely recognized the disk as one with its file system on it, or if the kernel just attached its file system to it because no other FSDs mounted the disk. This can be important information for a program that needs to know what file system is attached to the drive. It is quite easy to get into a situation where the FSD that recognizes a certain disk has not been installed into the system. In such a case, there is a potential for the data on the disk to be destroyed since the wrong file system is attached to the disk by default.

Bindings

C

<PRE> #define INCL_DOSFILEMGR

USHORT rc = DosQFSAttach(DeviceName, Ordinal, FSAInfoLevel, DataBuffer,

```
                                DataBufferLen, 0);
```

PSZ DeviceName; /* Device name or drive letter string */ USHORT Ordinal; /* Ordinal of entry in name list */ USHORT FSAInfoLevel; /* Type of attached FSD data required */ PBYTE DataBuffer; /* Returned data buffer */ PUSHORT DataBufferLen; /* Buffer length */ ULONG 0; /* Reserved (must be zero) */

USHORT rc; /* return code */ </PRE>

MASM

<PRE> EXTRN DosQFSAttach:FAR INCL_DOSFILEMGR EQU 1

PUSH@ ASCIIZ DeviceName ;Device name or drive letter string PUSH WORD Ordinal ;Ordinal of entry in name list PUSH WORD FSAInfoLevel ;Type of attached FSD data required PUSH@ OTHER DataBuffer ;Data buffer (returned) PUSH@ WORD DataBufferLen ;Buffer length (returned) PUSH DWORD 0 ;Reserved (must be zero) CALL DosQFSAttach

Returns WORD </PRE>

Dos16

From:
https://ftp.osfree.org/doku/ - **osFree wiki**

Permanent link:
https://ftp.osfree.org/doku/doku.php?id=en:docs:fapi:dosqfsattach&rev=1632030830

Last update: **2021/09/19 05:53**