

## FS\_PATHINFO

### Purpose

Returns information for a specific path or file.

### Calling Sequence

```
int far pascal FS_PATHINFO(flag, pcdfsi, pcdfsd, pName, iCurDirEnd, level,
                           pData, cbData)

unsigned short flag;
struct cdfsi far * pcdfsi;
struct cdfsd far * pcdfsd;
char far * pName;
unsigned short iCurDirEnd;
unsigned short level;
char far * pData;
unsigned short cbData;
```

### Where

*flag* indicates retrieval or setting of information.

<i>flag</i> == 0x0000	indicates retrieving information
<i>flag</i> == 0x0001	indicates setting information on the media
<i>flag</i> == 0x0010	indicates that the information being set must be written-through onto the disk before returning. This bit is never set when retrieving information.

All other values are reserved.

*pcdfsi* is a pointer the file-system-independent working directory structure.

*pcdfsd* is a pointer to the file-system-dependent working directory structure.

*pName* is a pointer to the ASCIIZ name of the file or directory for which information is to be retrieved or set.

The FSD does not need to verify this pointer.

*iCurDirEnd* is the index of the end of the current directory in *pName*.

This is used to optimize FSD path processing. If *iCurDirEnd* == -1, there is no current directory relevant to the name text, that is a device.

*level* is the information level to be returned.

*Level* selects among a series of data structures to be returned or set.

*pData* is the address of the application data area.

Addressing of this data area is not validated by the kernel (see *FSH\_PROBEBUF*). When retrieval (*flag*

== 0) is specified, the FSD places the information into the buffer. When outputting information to a file (*flag* == 1), the FSD retrieves that data from the application buffer.

*cbData* is the length of the application data area.

For *flag* == 0, this is the length of the data the application wishes to retrieve. If there is not enough room for the entire level of data to be returned, the FSD returns a BUFFER OVERFLOW error. For *flag* == 1, this is the length of data to be applied to the file.

## Remarks

See the descriptions of *DosQPathInfo* and *DosSetPathInfo* in the OS/2 Version 2.0 Control Program Programming Reference for details on information levels.

The FSD will not be called for *DosQPathInfo* level 5.

FSDs that are case-preserving (like HPFS) can decide to accept level 7 requests. A level 7 *DosQueryPathInfo* request asks the FSD to fill the *pData* buffer with the case-preserved path and filename of the path/filename passed in *pName*. Routing of level 7 requests will be determined by the kernel by checking the LV7 bit in a FSD's attribute double word.

From:

<http://www.osfree.org/doku/> - **osFree wiki**

Permanent link:

<http://www.osfree.org/doku/doku.php?id=en:ibm:ifs:routines:pathinfo>

Last update: **2014/05/13 02:08**

