

FSH_CANONICALIZE

Purpose

This function converts a path name to a canonical form by processing '.'s and '..'s, uppercasing, and prepending the current directory to non-absolute paths.

Calling Sequence

```
int far pascal FSH_CANONICALIZE(pPathName, cbPathBuf, pPathBuf, pFlags)

char far * pPathName;
unsigned short cbPathBuf;
char far * pPathBuf;
unsigned short far * pFlags;
```

Where

pPathName is a pointer to the ASCII path name to be canonicalized.

cbPathBuf is the length of path name buffer.

pPathBuf is the pointer to the buffer into which to copy the canonicalized path.

pFlags is the pointer to flags returned to the FSD.

Flags == 0x0040 indicates a non-8.3 filename format. All other values are reserved.

Returns

If no error is detected, a zero error code is returned. If an error is detected, one of the following error codes is returned:

- ERROR_PATH_NOT_FOUND - is an invalid path name-too many '.'s
- ERROR_BUFFER_OVERFLOW - the path name is too long.

Remarks

This routine processes DBCS characters properly.

The FSD is responsible for verifying the string pointers and checking for segment boundaries.

FSH_CANONICALIZE should be called for names passed into the FSD raw data packets. For example, names passed to *FS_FCTL* in the parameter area should be passed to *FSH_CANONICALIZE*. This routine does not need to be called for explicit names passed to the FSD, that is, the name passed to *FS_OPENCREATE*.

If the canonicalized name is being created as a file or directory, the non-8.3 attribute in the directory entry should be set according to the value returned in *pFlags*.

Note: OS/2 does not validate input parameters. Therefore, an FSD should call *FSH_PROBEBUF* where appropriate.

From:

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

Permanent link:

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

Last update: **2014/05/13 03:55**

