NAME

getvfsent, getvfsfile, getvfsspec, getvfsany – get vfstab file entry

SYNOPSIS

#include <stdio.h>
#include <sys/vfstab.h>

int getvfsent(FILE **fp*, **struct vfstab** **vp***);**

int getvfsfile(FILE **fp*, **struct vfstab** **vp*, **char** **file***);**

int getvfsspec(FILE *, struct vfstab **vp*, **char** **spec***);**

int getvfsany(FILE *, struct vfstab **vp*, **vfstab** **vref***);**

MT-LEVEL

Safe

DESCRIPTION

getvfsent( ), **getvfsfile( )**, **getvfsspec( )**, and **getvfsany( )** each fill in the structure pointed to by *vp* with the broken-out fields of a line in the **/etc/vfstab** file. Each line in the file contains a **vfstab** structure, declared in the **<sys/vfstab.h>** header:

```
char    *vfs_special;
char    *vfs_fsckdev;
char    *vfs_mountp;
char    *vfs_fstype;
char    *vfs_fsckpass;
char    *vfs_automnt;
char    *vfs_mntopts;
```

The fields have meanings described in **vfstab**(4).

getvfsent( ) returns a pointer to the next **vfstab** structure in the file; so successive calls can be used to search the entire file. **getvfsfile( )** searches the file referenced by *fp* until a mount point matching *file* is found and fills *vp* with the fields from the line in the file. **getvfsspec( )** searches the file referenced by *fp* until a special device matching *spec* is found and fills *vp* with the fields from the line in the file. *spec* will try to match on device type (block or character special) and major and minor device numbers. If it cannot match in this manner, then it compares the strings. **getvfsany( )** searches the file referenced by *fp* until a match is found between a line in the file and *vref*. *vref* matches the line if all non-null entries in *vref* match the corresponding fields in the file.

Note that these routines do not open, close, or rewind the file.

RETURN VALUES

If the next entry is successfully read by **getvfsent( )** or a match is found with **getvfsfile( )**, **getvfsspec( )**, or **getvfsany( )**, 0 is returned. If an end-of-file is encountered on reading, these functions return –1. If an error is encountered, a value greater than 0 is returned. The possible error values are:

VFS_TOOLONG A line in the file exceeded the internal buffer size of **VFS_LINE_MAX**.

VFS_TOOMANY A line in the file contains too many fields.

VFS_TOOFEW A line in the file contains too few fields.

NOTES

The members of the **vfstab** structure point to information contained in a static area, so it must be copied if it is to be saved.

FILES

/etc/vfstab

SEE ALSO

vfstab(4)