

NAME

getfsent, **getfsspec**, **getfsfile**, **setfsent**, **endfsent** - get file system descriptor file entry

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

```
#include <fstab.h>
```

```
struct fstab *  
getfsent(void);
```

```
struct fstab *  
getfsspec(const char *spec);
```

```
struct fstab *  
getfsfile(const char *file);
```

```
int  
setfsent(void);
```

```
void  
endfsent(void);
```

```
void  
setfstab(const char *file);
```

```
const char *  
getfstab(void);
```

DESCRIPTION

The **getfsent()**, **getfsspec()**, and **getfsfile()** functions each return a pointer to an object with the following structure containing the broken-out fields of a line in the file system description file, <*fstab.h*>.

```
struct fstab {  
    char    *fs_spec; /* block special device name */  
    char    *fs_file; /* file system path prefix */  
    char    *fs_vfstype; /* File system type, ufs, nfs */  
    char    *fs_mntops; /* Mount options ala -o */  
    char    *fs_type; /* FSTAB_* from fs_mntops */  
};
```

```

        int      fs_freq; /* dump frequency, in days */
        int      fs_passno; /* pass number on parallel fsck */
};

```

The fields have meanings described in `fstab(5)`.

The **setfsent()** function opens the file (closing any previously opened file) or rewinds it if it is already open.

The **endfsent()** function closes the file.

The **setfstab()** function sets the file to be used by subsequent operations. The value set by **setfstab()** does not persist across calls to **endfsent()**.

The **getfstab()** function returns the name of the file that will be used.

The **getfsspec()** and **getfsfile()** functions search the entire file (opening it if necessary) for a matching special file name or file system file name.

For programs wishing to read the entire database, **getfsent()** reads the next entry (opening the file if necessary).

All entries in the file with a type field equivalent to `FSTAB_XX` are ignored.

RETURN VALUES

The **getfsent()**, **getfsspec()**, and **getfsfile()** functions return a NULL pointer on EOF or error. The **setfsent()** function returns 0 on failure, 1 on success. The **endfsent()** function returns nothing.

ENVIRONMENT

PATH_FSTAB If the environment variable `PATH_FSTAB` is set, all operations are performed against the specified file. `PATH_FSTAB` will not be honored if the process environment or memory address space is considered "tainted". (See `issetugid(2)` for more information.)

FILES

/etc/fstab

SEE ALSO

`fstab(5)`

HISTORY

The **getfsent()** function appeared in 4.0BSD; the **endfsent()**, **getfsfile()**, **getfsspec()**, and **setfsent()** functions appeared in 4.3BSD; the **setfstab()** and **getfstab()** functions appeared in FreeBSD 5.1.

BUGS

These functions use static data storage; if the data is needed for future use, it should be copied before any subsequent calls overwrite it.