FSMakeFSSpec Convert a file or directory spec into an FSSpec record

#include <Files.h> **File Manager**

```
OSErr       FSMakeFSSpec(vRefNum, dirID, fileName, spec);
short       vRefNum ;     volume reference number
long        dirID ;       parent directory ID
Str255      fileName ;    a full or partial pathname
FSSpecPtr   spec;         a pointer to an FSSpec record
            returns       Error Code; 0=no error
```

You use the **FSMakeFSSpec** function to convert a conventional file or directory specification into an **FSSpec** record.

vRefNum is the volume reference number, a working directory reference number, a drive number, or 0 for the default volume.

dirID is usually the parent directory ID of the target object. If the directory is sufficiently specified by either vRefNum or fileName, dirID can be 0. If you explicitly specify dirID (that is, if it is any value other than 0), and if vRefNum is a working directory reference number, dirID overrides the directory ID included in vRefNum. If the fileName parameter is an empty string, **FSMakeFSSpec** creates an **FSSpec** record for a directory specified by either the dirID or vRefNum parameter.

fileName is a full or partial pathname. If it is a full pathname, **FSMakeFSSpec** ignores vRefNum and dirID. A partial pathname might identify only the final target, or it might include one or more parent directory names. If fileName is a partial pathname, vRefNum, dirID , or both must be valid.

spec is a pointer to an **FSSpec** record, which **FSMakeFSSpec** fills in

Returns: an operating system Error Code. It will be one of:

noErr	(0)	No error
fnfErr	(-43)	File or directory does not exist (**FSSpec** is still valid)

Notes: **FSMakeFSSpec** places the specification in the *spec* parameter. Call **FSMakeFSSpec** whenever you want to create an FSSpec record.

You can pass the input to **FSMakeFSSpec** in any of the four ways described in **File Specification Strategies** under the section entitled **Identifying Files, Directories, and Volumes**. See the table in **Using FSSpec Records** under **Using the File Manager** for details of how **FSMakeFSSpec** interprets input.

If the specified volume is mounted and the specified parent directory exists, but the target file or directory doesn't exist in that location, **FSMakeFSSpec** fills in the record and then returns fnfErr instead of noErr. The record is valid, but it describes a target that doesn't exist. You can use the record for other operations, such as creating a file with the **FSpCreate** function.

In addition to the result codes listed here, **FSMakeFSSpec** can return a number of different **File Manager** error codes. If you receive any result code other than noErr or fnfErr, **FSMakeFSSpec** returns a NIL **FSSpecPtr**.