

FInfo structure

```
#include <Files.h>
```

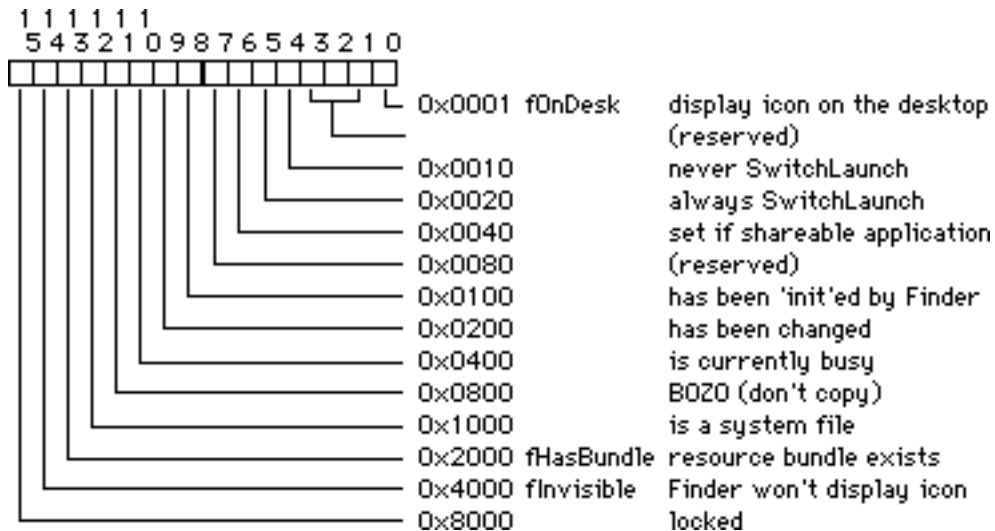
typedef struct FInfo {		<u>Size</u>	<u>Offset</u>	<u>Description</u>
<u>OSType</u>	fdType;	4	0	File type, eg 'TEXT'
<u>OSType</u>	fdCreator;	4	4	File creator's signature; eg,'MSWD'
<u>unsigned short</u>	fdFlags;	2	8	Flag bits used by Finder (see Notes)
<u>Point</u>	fdLocation;	4	10	Position of top-left corner of file's icon
<u>short</u>	fdFldr;	2	14	Folder number or other code (not used in HFS)
} FInfo;		16		

Notes: The fdType field helps applications figure out what to do with a file. It is also used by **SFGetFile** et al. to let Standard File show a subset of files in a directory. The 'APPL' (executable application) and 'TEXT' (CR-delimited text file) are standard types; others are defined by applications and can be any 32-bit value.

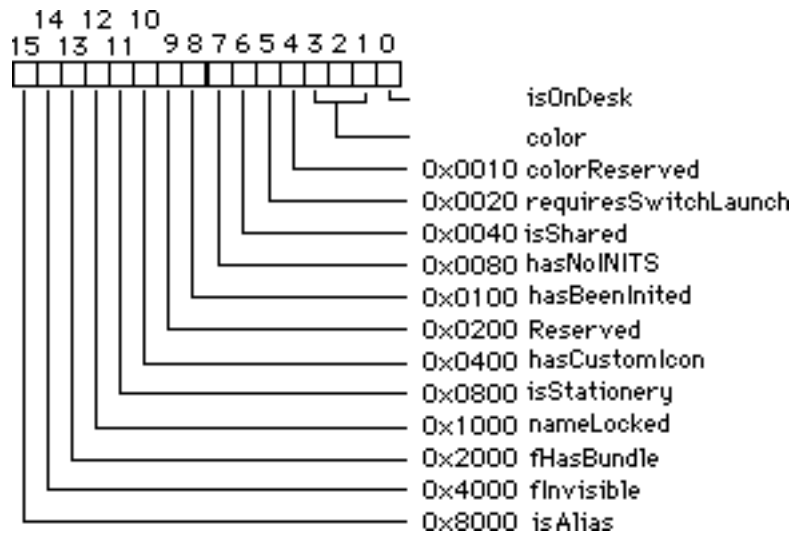
The fdCreator field specifies which application should be executed when the file's icon is double clicked. a value of '????' indicates it is not to be opened that way. Other values identify the application by its Apple-approved unique application signature. See **SetFInfo**.

The meaning of the flag bits in this structure have changed **significantly** in System 7.0. Make sure you look at both diagrams below. The first one shows the old flags, and the second shows the new flags.

The **old** fdFlags word is a set of bit flags, formatted as follows:



The **new** fdFlags word is a set of bit flags, formatted as follows:



The new meanings of the flags are as follows:

Flag name	Bit number	Description
isAlias	15	The file is an alias file. Reserved for directories - in which case, set to 0.
isInvisible	14	The file or directory is invisible from the Finder and from the Standard File Package dialog boxes.
hasBundle	13	The file contains a bundle resource. Reserved for directories - in which case, set to 0.
nameLocked	12	The file or directory can't be renamed from the Finder, and the icon can't be changed.
isStationery	11	The file is a stationery pad. Reserved for directories - in which case, set to 0.
hasCustomIcon	10	The file or directory contains a customized icon.
Reserved	9	Reserved; set to 0.
hasBeenInitd	8	The Finder has recorded information from the file's bundle resource into the desktop database and given the file or folder a position on the desktop.
hasNoINITS	7	The file contains no ' <u>INIT</u> ' resources; set to 0. Reserved for directories, set to 0.
isShared	6	The application is available to multiple users. Defined for applications only; otherwise, set to 0.
requiresSwitchLaunch	5	Unused and reserved for System 7.0; set to 0.

colorReserved	4	Unused and reserved for System 7.0; set to 0.
color	1-3	Three bits of color coding.
isOnDesk	0	Unused and reserved for System 7.0; set to 0.

Masks for two of these bits are available as:

<u>fHasBundle</u>	8192	set if the file has a bundle
<u>fInvisible</u>	16384	set if the icon is invisible

Of these Finder flags, the only ones that you might ever want to set are these:

isInvisible This flag specifies that a file is invisible from the Finder and from the **Standard File Package** dialog boxes. Making a file invisible is generally not recommended. Not even temporary files need to be invisible because the Temporary Items folder into which they should be written is invisible.

hasBundle This flag specifies that a file has a '**BNDL**' resource that associates the file with your own icons. When the Finder displays or manipulates a file, it checks the file's hasBundle bit (also called the bundle bit). If that bit is not set, the Finder displays a default icon for that file type. If the hasBundle bit is set, the Finder checks the hasBeenInitd bit. If the hasBeenInitd bit is set, the Finder uses the information in the desktop database to display that file's icon. If the hasBeenInitd bit is not set, the Finder installs the information from the bundle resource in the desktop database and sets the hasBeenInitd bit.

nameLocked This flag specifies that a file cannot be renamed from the Finder and that the file cannot have customized icons assigned to it by users.

isStationery This flag specifies that a file is a stationery pad. To support **Stationery Pads** your application should check this bit for every document passed to it by either the Finder or **Standard File Package**. StandardGetFile and CustomGetFile return this flag in the sfType field. If the isStationery bit is set for a file that a user wants to open, your application should copy the template's contents into a new document and open the document in an untitled window.

hasCustomIcon This flag specifies that a file has a customized icon. See **Customized Icons** for more information.

The fdFldr field is not used in HFS volumes. In flat volumes, it may contain one of the following values or constants, defined in Files.h:

folder number (>0)	Icon goes in this folder
fDisk (0)	Icon is in the main disk window

fDeskTop (-2) Icon is on the desktop
fTrash (-3) Icon is in the trash window