PBHRename Rename a file, volume, or directory (HFS only)

#include <Files.h> **File Manager (PBxxx)**

OSErr	**PBHRename(***pb*, *async* **);**	
HParmBlkPtr	*pb* ;	address of an 80-byte HParamBlockRec union
Boolean	*async* ;	0=await completion; 1=immediate return
	returns	Error Code; 0=no error

PBHRename renames a file, volume, or directory. It works like **PBRename** except that it ignores file version numbers and allows you to specify a directory using a "hard" directory number. If a file ID exists for the file being renamed, the file ID remains with the file.

pb is the address of an 80-byte HParamBlockRec union. You must specify fields from two members of the union. The relevant fields are as follows:

Out-In	Name	Type	Size	Offset	Structure	Description
–>	ioCompletion	ProcPtr	4	12	ioParam	Completion rtn address (used only if *async* =TRUE)
–>	ioNamePtr	StringPtr	4	18	ioParam	Address of current filename
–>	ioVRefNum	short	2	22	ioParam	Volume, drive, or directory ref
–>	ioMisc	Ptr	2	28	ioParam	Address of desired new filename
–>	ioDirID	long	4	48	fileParam	"hard" dir ID (0=use ioVRefNum)
<-	ioResult	OSErr	2	16	ioParam	Error Code (0=no err,1=not done)

async is a Boolean value. Use FALSE for normal (synchronous) operation or TRUE to enqueue the request and resume control immediately. See Async I/O.

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

noErr	(0)	No error
bdNamErr	(-37)	Bad file or volume name
dirFulErr	(-33)	Directory full
dupFNErr	(-48)	Duplicate filename (new name already exists)
extFSErr	(-58)	External file system
fLckdErr	(-45)	File is locked
fnfErr	(-43)	File not found
fsRnErr	(-59)	File system rename error
ioErr	(-36)	I/O error
nsvErr	(-35)	No such volume
paramErr	(-50)	No default volume
vLckdErr	(-46)	Volume is locked
wPrErr	(-44)	Diskette is write-protected

Notes: See **PBRename** for related details. This HFS-specific variation is the same except that you can identify the directory of the file to rename via a "hard" directory ID (in ioDirID). However, you must use two different members of ParamBlockRec in order to satisfy the input requirements.

As with other **PBH**xxx functions, setting ioDirID to 0 causes the File Manager to ignore that field and look to ioVRefNum and/or ioNamePtr to specify the directory. If ioDirID is not 0, it specifies the directory in which the file resides. Some examples:

```
ParamBlockRec pb;
long           myBaseDir;        /* a hard directory ID */
```

```
/* ====== using multiple-name filenames ======
*/
pb.ioParam.ioNamePtr=(StringPtr)"\pHardDisk:Letters:Jones";
pb.ioParam.ioVRefNum = 0;          /* ignored (valid full pathname) */
pb.fileParam.ioDirID = 0;          /* 0=don't use (same reason) */
pb.ioParam.ioMisc = (Ptr)"\pHardDisk:Letters:Smith";
rc = PBHRename( &pb, FALSE );

/* ====== renaming a file in a dir whose "hard" ID is known ======
*/
pb.ioParam.ioNamePtr = (StringPtr)"\pJones";    /* old filename */
pb.ioParam.ioVRefNum = 0;             /* use default volume (disk) */
pb.fileParam.ioDirID = myBaseDir;           /* parent of the file */
pb.ioParam.ioMisc = (Ptr)"\pSmith";        /* new name */
rc = PBHRename( &pb, FALSE );
```

This function cannot be used to move a file to a different directory (use **PBCatMove** for that). The 'new name' in ioMisc must be in the same directory as in ioNamePtr (or as otherwise identified by ioVRefNum or ioDirID); that is, **PBHRename** can rename only one file or directory at a time. It would take two calls to rename "HD20:Ltrs:Smith" to "HD20:Letters:Jones".

The high-level version of this function is **Rename**. On flat volumes, you may want to use **PBRename**.