

PBDTSetComment Add user comment to desktop database

#include <Files.h>

Finder Interface

```
OSErr      PBDTSetComment(paramBlock, async);
DTPBPtr    paramBlock ;    pointer to a DTPB Param Block
Boolean    async;          0 = await completion; 1 = immediate return
```

Parameter block

→	12	ioCompletion	long	completion routine
←	16	ioResult	short	result code
→	18	ioNamePtr	long	pointer to file or directory name
→	24	ioDTRefNum	short	desktop database reference number
→	32	ioDTBuffer	long	pointer to comment text
→	36	ioDTReqCount	long	comment length
→	48	ioDirID	long	parent directory of file or directory

PBDTSetComment establishes the user comment associated with a file or directory in the database specified in ioDTRefNum. You specify the object name through ioNamePtr and the parent directory ID in ioDirID. You put the comment as a plain text string in a buffer pointed to by ioDTBuffer, and you specify the length of the buffer (in bytes) in ioDTReqCount. The maximum length of a comment is 200 bytes; longer comments are clipped. Since the comment is a plain text string and not a Pascal string, the **Desktop Manager** relies on the value in ioDTReqCount for determining the length of the buffer.

If the specified object already has a comment in the database, the new comment replaces the old.

Returns: an Error code. It will be one of the following:

noErr	(0)	No error
ioErr	(-36)	I/O error
fnfErr	(-43)	File or directory does not exist
wPrErr	(-44)	Volume is locked through hardware
vLckdErr	(-46)	Volume is locked through software
rfNumErr	(-51)	Reference number invalid
extFSErr	(-58)	External file system-file system identifier is nonzero

Note: There is a second, asynchronous, version of this function. It does not take a second parameter; instead, it adds the suffix "Async" to the name of the routine.

Similarly, the third (synchronous) version of the routine does not take a second parameter; instead, it adds the suffix "Sync" to the name of the routine.

Note, however, that the second and third versions of these routines do not use the glue code that the first versions use and are therefore more efficient.