NAME

sbget, **sbsearch**, **sbput**, **sbread**, **sbfind**, **sbwrite** - read and write superblocks of a UFS file system

LIBRARY

UFS File System Access Library (libufs, -lufs)

SYNOPSIS

```
#include <sys/param.h>
#include <sys/mount.h>
#include <ufs/ufs/ufsmount.h>
#include <ufs/ufs/dinode.h>
#include <ufs/ffs/fs.h>
#include <libufs.h>

int
sbget(int devfd, struct fs **fsp, off_t sblockloc, int flags);

int
sbsearch(int devfd, struct fs **fsp, int flags);

int
sbput(int devfd, struct fs *fs, int numaltwrite);

int
sbread(struct uufsd *disk);

int
sbfind(struct uufsd *disk, int flags);

int
sbwrite(struct uufsd *disk, int all);
```

DESCRIPTION

The **sbget**(), **sbsearch**(), **sbread**(), and **sbfind**() functions provide superblock reads for libufs(3) consumers. The **sbput**() and **sbwrite**() functions provide superblock writes for libufs(3) consumers.

The **sbget**() and **sbsearch**() functions first allocate a buffer to hold the superblock. Using the *devfd* file descriptor that references the filesystem disk, **sbget**() reads the superblock located at the byte offset specified by *sblockloc* into the allocated buffer. The value **UFS_STDSB** may be specified for *sblockloc* to request that the standard location for the superblock be read. The **sbsearch**() function uses the *devfd*

file descriptor that references the filesystem disk, to search first for the superblock at the standard location. If it is not found or is too damaged to use **sbsearch**() will attempt to find one of the filesystem's alternate superblocks. Flags are specified by *or*'ing the following values:

UFS_NOCSUM Causes only the superblock itself to be returned, but does not read in any auxiliary data structures like the cylinder group summary information.

UFS_NOMSG Indicates that superblock inconsistency error messages should not be printed.

If successful, **sbget**() and **sbsearch**() functions return a pointer to the buffer containing the superblock in *fsp*. The **sbget**() and **sbsearch**() functions are safe to use in threaded applications.

The **sbput**() function writes the superblock specified by *fs* to the location from which it was read on the disk referenced by the *devfd* file descriptor. Additionally, the **sbput**() function will update the first *numaltwrite* alternate superblock locations. To update all the alternate superblocks, specify a *numaltwrite* value of *fs->fs_ncg*. The **sbput**() function is safe to use in threaded applications. Note that the **sbput**() function needs to be called only if the superblock has been modified and the on-disk copy needs to be updated.

The **sbread**() function reads the standard filesystem superblock. The **sbfind**() function tries to find a usable superblock. It searchs first for the superblock at the standard location. If it is not found or is too damaged to use **sbfind**() will attempt to find one of the filesystem's alternate superblocks. If successful **sbread**() and **sbfind**() return a superblock in the *d_sb*, structure embedded in the given user-land UFS disk structure.

The **sbwrite**() function writes the superblock from the *d_sb*, structure embedded in the given user-land UFS disk structure to the location from which it was read. Additionally, the **sbwrite**() function will write to all the alternate superblock locations if the *all* value is non-zero.

RETURN VALUES

The **sbread**() and **sbwrite**() functions return the value 0 if successful; otherwise the value -1 is returned and the global variable *errno* is set to indicate the error. The **sbget**(), **sbsearch**(), and **sbput**() functions return the value 0 if successful; otherwise they return one of the errors described below.

ERRORS

The errors returned by **sbget**(), **sbsearch**(), **sbread**(), and **sbfind**(), include any of the errors specified for the library function bread(3). Additionally, they may follow the libufs(3) error methodologies in situations where no usable superblock could be found.

The errors returned by **sbput**() and **sbwrite**() include any of the errors specified for the library function

bwrite(3).

SEE ALSO

bread(3), bwrite(3), libufs(3)

HISTORY

These functions first appeared as part of libufs(3) in FreeBSD 5.0.

AUTHORS

Juli Mallett <*jmallett@FreeBSD.org*>
Marshall Kirk McKusick <*mckusick@FreeBSD.org*>