NAME

volmgt_symname, volmgt_symdev – convert between Volume Management symbolic names, and the devices that correspond to them

SYNOPSIS

cc [*flag* . . .] *file* . . . **–lvolmgt** [*library*. . . .]

#include <volmgt.h>

char ∗**volmgt_symname(char** ∗*pathname***);**

char ∗**volmgt_symdev(char** ∗*symname***);**

MT-LEVEL

MT-Safe

DESCRIPTION

These two routines compliment each other, translating between Volume Management's symbolic name for a device, called a *symname*, and the **/dev** *pathname* for that same device.

volmgt_symname( ) converts a supplied **/dev** *pathname* to a **symname**, Volume Management's idea of that device's symbolic name (see **volfs**(7FS) for a description of Volume Management symbolic names).

volmgt_symdev( ) does the opposite conversion, converting between a *symname*, Volume Management's idea of a device's symbolic name for a volume, to the **/dev** *pathname* for that device.

RETURN VALUES

volmgt_symname( ) returns the symbolic name for the device pathname supplied, and **volmgt_symdev**( ) returns the device pathname for the supplied symbolic name.

These strings are allocated upon success, and therefore must be freed by the caller when they are no longer needed (see **free**(3C)).

ERRORS

volmgt_symname( ) can fail, returning a null string pointer, if a **stat**(2) of the supplied **pathname** fails, or if an **open**(2) of **/dev/volctl** fails, or if any of the following is true:

ENXIO Volume Management is not running.

EINTR An interrupt signal was detected while trying to convert the supplied *pathname* to a *symname*.

volmgt_symdev( ) can fail if an **open**(2) of **/dev/volctl** fails, or if any of the following is true:

ENXIO Volume Management is not running.

EINTR An interrupt signal was detected while trying to convert the supplied *symname* to a **/dev** *pathname*.

EXAMPLES

The following tests how many floppies Volume Management currently sees in floppy drives (up to 10):

```
for (i=0; i < 10; i++) {
      (void) sprintf(path, "floppy%d", i);
      if (volmgt_symdev(path) != NULL) {
            (void) printf("volume %s is in drive %d\n",
```

```
                path, i);
        }
    }
```

This code finds out what symbolic name (if any) Volume Management has for **/dev/rdsk/c0t6d0s2**:

```
if ((nm = volmgt_symname("/dev/rdsk/c0t6d0s2")) == NULL) {
    (void) printf("path not managed\n");
} else {
    (void) printf("path managed as %s\n", nm);
}
```

SEE ALSO

cc(1B), **vold**(1M), **open**(2), **stat**(2), **free**(3C), **malloc**(3C), **volmgt_check**(3X), **volmgt_inuse**(3X), **volmgt_running**(3X), **volfs**(7FS)

NOTES

These routines only work when Volume Management is running.

BUGS

There should be a straightforward way to query Volume Management for a list of all media types it’s managing, and how many of each type are being managed.