

NAME

devfs_set_cdevpriv, **devfs_get_cdevpriv**, **devfs_clear_cdevpriv**, **devfs_foreach_cdevpriv** - manage per-open filedescriptor data for devices

SYNOPSIS

```
#include <sys/param.h>
```

```
#include <sys/conf.h>
```

```
typedef void d_priv_dtor_t(void *data);
```

```
int
```

```
devfs_get_cdevpriv(void **datap);
```

```
int
```

```
devfs_set_cdevpriv(void *priv, d_priv_dtor_t *dtr);
```

```
void
```

```
devfs_clear_cdevpriv(void);
```

```
int
```

```
devfs_foreach_cdevpriv(struct cdev *dev, int (*cb)(void *data, void *arg), void *arg);
```

DESCRIPTION

The **devfs_xxx_cdevpriv()** family of functions allows the *cdev* driver methods to associate some driver-specific data with each user process `open(2)` of the device special file. Currently, functioning of these functions is restricted to the context of the *cdevsw* switch method calls performed as `devfs(5)` operations in response to system calls that use filedescriptors.

The **devfs_set_cdevpriv()** function associates a data pointed by *priv* with current calling context (filedescriptor). The data may be retrieved later, possibly from another call performed on this filedescriptor, by the **devfs_get_cdevpriv()** function. The **devfs_clear_cdevpriv()** disassociates previously attached data from context. Immediately after **devfs_clear_cdevpriv()** finished operating, the *dtr* callback is called, with private data supplied *data* argument. The **devfs_clear_cdevpriv()** function will be also be called if the open callback function returns an error code.

On the last filedescriptor close, system automatically arranges **devfs_clear_cdevpriv()** call.

If successful, the functions return 0.

The function **devfs_set_cdevpriv()** returns the following values on error:

[ENOENT] The current call is not associated with some filedescriptor.

[EBUSY] The private driver data is already associated with current filedescriptor.

The function **devfs_get_cdevpriv()** returns the following values on error:

[EBADF] The current call is not associated with some filedescriptor.

[ENOENT] The private driver data was not associated with current filedescriptor, or **devfs_clear_cdevpriv()** was called.

The function **devfs_foreach_cdevpriv()** sequentially calls the function *cb* for each **cdevpriv** structure, currently associated with the *cdev* device. The iterated **cdevpriv** data pointer and the user-supplied context *arg* are passed to the function *cb*. If *cb* returns non-zero value, the iteration stops on that element. The **devfs_foreach_cdevpriv()** returns the return value from the last call to *cb*, or zero if no **cdevpriv** data is currently associated with the device.

Current implementation of the iterator makes it impossible to use any blockable locking inside the callback *cb*.

SEE ALSO

close(2), open(2), devfs(5)

HISTORY

The **devfs_cdevpriv()** family of functions first appeared in FreeBSD 7.1.