

NAME

g_access - control access to GEOM consumers and their providers

SYNOPSIS

```
#include <geom/geom.h>
```

```
int
```

```
g_access(struct g_consumer *cp, int dcr, int dcw, int dce);
```

DESCRIPTION

The **g_access()** function allows to open, close, and generally change access to the provider which is attached to the given consumer *cp*. The arguments *dcr*, *dcw*, and *dce* represent relative read, write, and exclusive access count changes. Read and write access counts are self explanatory, and exclusive access counts deny write access to other interested parties. A provider's access count is the sum of the access counts of all attached consumers.

After attaching a consumer to a provider with **g_attach(9)**, the **g_access()** function has to be called on the consumer before starting I/O requests.

RESTRICTIONS/CONDITIONS

The consumer has to be attached to a provider.

The intended change must not result in a negative access count.

No-operation is not permitted (*dcr* = *dcw* = *dce* = 0).

The provider's geom must have an access method defined (e.g., *gp->access*).

The topology lock has to be held.

RETURN VALUES

The **g_access()** function returns 0 if successful; otherwise an error code is returned. Note that **g_access()** cannot fail when the arguments *dcr*, *dcw*, and *dce* are less than or equal to 0.

EXAMPLES

Create a consumer, attach it to a given provider, gain read access and read first sector.

```
void
some_function(struct g_geom *mygeom, struct g_provider *pp)
{
```

```
struct g_consumer *cp;
void *ptr;
int error;

g_topology_assert();

/* Create new consumer on 'mygeom' geom. */
cp = g_new_consumer(mygeom);
/* Attach newly created consumer to given provider. */
if (g_attach(cp, pp) != 0) {
    g_destroy_consumer(cp);
    return;
}
/* Open provider for reading through our consumer. */
error = g_access(cp, 1, 0, 0);
if (error != 0) {
    printf("Cannot access provider: %s\n", error);
    g_detach(cp);
    g_destroy_consumer(cp);
    return;
}

/*
 * Don't hold topology lock while reading.
 */
g_topology_unlock();
ptr = g_read_data(cp, 0, pp->sectorsize, &error);
if (ptr == NULL)
    printf("Error while reading: %d\n", error);

/*
 * Do something useful with data.
 */
g_topology_lock();

/* Disconnect from provider (release access count). */
g_access(cp, -1, 0, 0);
/* Detach from provider. */
g_detach(cp);
/* Destroy consumer. */
g_destroy_consumer(cp);
```

```
}
```

ERRORS

Possible errors:

[EPERM] The function is trying to open a provider with an exclusive access count, but it is already open for writing.

[EPERM] The function is trying to open a provider for writing, but it is already exclusively open.

Any other error that can be returned by the provider's access method.

SEE ALSO

geom(4), DECLARE_GEOM_CLASS(9), g_attach(9), g_bio(9), g_consumer(9), g_data(9), g_event(9), g_geom(9), g_provider(9), g_provider_by_name(9), g_wither_geom(9)

AUTHORS

This manual page was written by Pawel Jakub Dawidek <pjd@FreeBSD.org>.