

NAME

geom_map - GEOM module that maps defined items as separate partitions

SYNOPSIS

To compile this driver into the kernel, place the following line in your kernel configuration file:

```
options geom_map
```

DESCRIPTION

The **geom_map** framework provides support for mapping defined parts of the media. Basically it is helpful for embedded devices where in the one continuous flash are loader, kernel and rootfs parts. **geom_map** allows making them available as separate parts and protects the loader from being overwritten.

At boot time **geom_map** partitions are listed (only with bootverbose) as:

```
MAP: 0x30000, data=0x30000 "/dev/map/bootloader"
MAP: 30000x10000, data=0x10000 "/dev/map/factory"
MAP: 40000x7a0000, data=0x7a0000 "/dev/map/upgrade"
MAP: search key "!.bin/sh" from 0x100000, step 0x10000
MAP: 40000x110000, data=0x110000 "/dev/map/kernel"
MAP: search key "!.bin/sh" from 0x100000, step 0x10000
MAP: 150000x690000, data=0x690000 "/dev/map/rootfs"
MAP: 7e0000x20000, data=0x20000 "/dev/map/config"
```

The current **geom_map** configuration can be accessed with the following `sysctl(8)` nodes: `kern.geom.conftxt`, `kern.geom.confxml`, and `kern.geom.confdot` or by using "geom map list".

```
# sysctl -n kern.geom.conftxt
0 MD md0 10485760 512 u 0 s 512 f 0 fs 0 1 10485760 t malloc
0 DISK cfid0 8388608 4 hd 0 sc 0
1 MAP map/config 131072 4 i 5 o 8257536 entry 0 dsize 131072
1 MAP map/rootfs 6881280 4 i 4 o 1376256 entry 0 dsize 6881280
2 UZIP map/rootfs.uzip 18677760 512
1 MAP map/kernel 1114112 4 i 3 o 262144 entry 0 dsize 1114112
1 MAP map/upgrade 7995392 4 i 2 o 262144 entry 0 dsize 7995392
1 MAP map/factory 65536 4 i 1 o 196608 entry 0 dsize 65536
1 MAP map/bootloader 196608 4 i 0 o 0 entry 0 dsize 196608
```

Driver configuration can be done in `device.hints(5)`. List of used parameters:

at select media to attach

name name of partition (will create device */dev/map/that_name*)

start offset from the beginning of the parent media to start of the mapped partition. This field can also have a special value "search:*searchstart:searchstep:searchkey*", where:

searchstart
offset from the beginning of the parent media where search will be started

searchstep
value of the increment used while searching for the partition boundary markers

searchkey
key which will be used to find partition boundary markers. The wildcard '.' can be used to match any character on that position

end offset from the beginning of the parent media to the end of the mapped partition. This field can also have the special value "search:*searchstart:searchstep:searchkey*", as described above.

offset offset where the data of the mapped partition begins

Each record contains the start address (in bytes) from the media begin, size (in bytes), offset where the data of mapped partition begins, and the name of new device.

```
MAP: 150000x690000, data=0x690000 "/dev/map/rootfs"
```

00150000 - start address

00690000 - size

00000000 - data begin from zero offset

00690000 - data size

"map/rootfs" - new media will be accessible via /dev/map/rootfs dev.

EXAMPLES

If we need to implement layout shown above, we need to define the following hints:

```
hint.map.0.at="cfid0"
```

```
hint.map.0.start=0x00000000
```

```
hint.map.0.end=0x00030000
```

```
hint.map.0.name="bootloader"
```

```
hint.map.0.readonly=1
```

This defines */dev/map/bootloader* at disk *cfid0* starting at 0x00000000 and ending at 0x00030000, it is also marked as *readonly*.

```
hint.map.1.at="cfid0"
hint.map.1.start=0x00030000
hint.map.1.end=0x00040000
hint.map.1.name="factory"
```

```
hint.map.2.at="cfid0"
hint.map.2.start=0x00040000
hint.map.2.end=0x007e0000
hint.map.2.name="upgrade"
```

```
hint.map.3.at="cfid0"
hint.map.3.name="kernel"
hint.map.3.start=0x00040000
hint.map.3.end="search:0x00100000:0x10000:./bin/sh"
```

This defines */dev/map/kernel* at disk *cfid0* starting at 0x00040000, but the end position must be searched by finding the key *./bin/sh*, from offset 0x00100000 to the end of media with step 0x10000. The real marker in this case is *#!/bin/sh*, but *#* terminates the line when the hints file is parsed, so we need to use wildcard *.* instead of *#*.

```
hint.map.4.at="cfid0"
hint.map.4.name="rootfs"
hint.map.4.start="search:0x00100000:0x10000:./bin/sh"
hint.map.4.end=0x007e0000
```

```
hint.map.5.at="cfid0"
hint.map.5.start=0x007e0000
hint.map.5.end=0x00800000
hint.map.5.name="config"
```

SEE ALSO

geom(4), *geom(8)*, *sysctl(8)*

AUTHORS

The *geom_map* driver was written by Aleksandr Rybalko <ray@ddteam.net>.