

NAME

bsdlabel - read and write BSD label

SYNOPSIS

bsdlabel [-A] *disk* | -f *file*

bsdlabel -w [-An] [-B [-b *boot*]] [-m *machine*] *disk* | -f *file* [*type*]

bsdlabel -e [-An] [-B [-b *boot*]] [-m *machine*] *disk* | -f *file*

bsdlabel -R [-An] [-B [-b *boot*]] [-m *machine*] [-f] *disk* | -f *file* *protofile*

DESCRIPTION

The **bsdlabel** utility installs, examines or modifies the BSD label on a disk partition, or on a file containing a partition image. In addition, **bsdlabel** can install bootstrap code.

Disk Device Name

When specifying the device (i.e., when the **-f** option is not used), the */dev/* path prefix may be omitted; the **bsdlabel** utility will automatically prepend it.

General Options

The **-A** option enables processing of the historical parts of the BSD label. If the option is not given, suitable values are set for these fields.

The **-f** option tells **bsdlabel** that the program will operate on a file instead of a disk partition.

The **-n** option stops the **bsdlabel** program right before the disk would have been modified, and displays the result instead of writing it.

The **-m** *machine* argument forces **bsdlabel** to use a layout suitable for a different architecture. Current valid values are **i386** and **amd64**. If this option is omitted, **bsdlabel** will use a layout suitable for the current machine.

Reading the Disk Label

To examine the label on a disk drive, use the form

bsdlabel [-A] [-m *machine*] *disk*

disk represents the disk in question, and may be in the form *da0* or */dev/da0*. It will display the partition layout.

Writing a Standard Label

To write a standard label, use the form

bsdlabel -w [-An] [-m *machine*] *disk* [*type*]

If the drive *type* is specified, the entry of that name in the `disktab(5)` file is used; otherwise, or if the type is specified as 'auto', a default layout is used.

Editing an Existing Disk Label

To edit an existing disk label, use the form

bsdlabel -e [-An] [-m *machine*] *disk*

This command opens the disk label in the default editor, and when the editor exits, the label is validated and if OK written to disk.

Restoring a Disk Label From a File

To restore a disk label from a file, use the form

bsdlabel -R [-An] [-m *machine*] *disk* *protofile*

The **bsdlabel** utility is capable of restoring a disk label that was previously saved in a file in ASCII format. The prototype file used to create the label should be in the same format as that produced when reading or editing a label. Comments are delimited by '#' and newline.

Installing Bootstraps

If the **-B** option is specified, bootstrap code will be read from the file `/boot/boot` and written to the disk. The **-b** *boot* option allows a different file to be used.

FILES

`/boot/boot` Default boot image.

`/etc/disktab` Disk description file.

SAVED FILE FORMAT

The **bsdlabel** utility uses an ASCII version of the label when examining, editing, or restoring a disk label. The format is:

8 partitions:

```
#   size  offset  fstype  [fsize bsize bps/cpg]
a: 81920   16  4.2BSD  2048 16384 5128
b: 1091994 81936  swap
c: 1173930   0  unused   0   0   # "raw" part, don't edit
```

If the **-A** option is specified, the format is:

```
# /dev/da1c:
type: SCSI
disk: da0s1
label:
flags:
bytes/sector: 512
sectors/track: 51
tracks/cylinder: 19
sectors/cylinder: 969
cylinders: 1211
sectors/unit: 1173930
rpm: 3600
interleave: 1
trackskew: 0
cylinderskew: 0
headswitch: 0      # milliseconds
track-to-track seek: 0 # milliseconds
drivedata: 0

8 partitions:
#   size offset fstype [fsize bsize bps/cpg]
a: 81920   16 4.2BSD  1024 8192  16
b: 160000 81936  swap
c: 1173930 0  unused    0  0    # "raw" part, don't edit
```

Lines starting with a '#' mark are comments.

The partition table can have up to 8 entries. It contains the following information:

The partition identifier is a single letter in the range 'a' to 'h'. By convention, partition 'c' is reserved to describe the entire disk.

size The size of the partition in sectors, **K** (kilobytes - 1024), **M** (megabytes - 1024*1024), **G** (gigabytes - 1024*1024*1024), **%** (percentage of free space *after* removing any fixed-size partitions other than partition 'c'), or ***** (all remaining free space *after* fixed-size and percentage partitions). For partition 'c', a size of ***** indicates the entire disk. Lowercase versions of suffixes **K**, **M**, and **G** are allowed. Size and suffix should be specified without any spaces between them.

Example: 2097152, 1G, 1024M and 1048576K are all the same size (assuming 512-byte sectors).

offset The offset of the start of the partition from the beginning of the drive in sectors, or * to have **bsdlabell** calculate the correct offset to use (the end of the previous partition plus one, ignoring partition 'c'). For partition 'c', * will be interpreted as an offset of 0. The first partition should start at offset 16, because the first 16 sectors are reserved for metadata.

fstype Describes the purpose of the partition. The above example shows all currently used partition types. For UFS file systems and ccd(4) partitions, use type **4.2BSD**. For Vinum drives, use type **vinum**. Other common types are **swap** and **unused**. By convention, partition 'c' represents the entire slice and should be of type **unused**, though **bsdlabell** does not enforce this convention. The **bsdlabell** utility also knows about a number of other partition types, none of which are in current use. (See the definitions starting with FS_UNUSED in <sys/disklabel.h> for more details.)

fsize For **4.2BSD** file systems only, the fragment size; see newfs(8).

bsize For **4.2BSD** file systems only, the block size; see newfs(8).

bps/cpg

For **4.2BSD** file systems, the number of cylinders in a cylinder group; see newfs(8).

EXAMPLES

Display the label for the first slice of the *da0* disk, as obtained via */dev/da0s1*:

```
bsdlabell da0s1
```

Save the in-core label for *da0s1* into the file *savedlabel*. This file can be used with the **-R** option to restore the label at a later date:

```
bsdlabell da0s1 > savedlabel
```

Create a label for *da0s1*:

```
bsdlabell -w /dev/da0s1
```

Read the label for *da0s1*, edit it, and install the result:

```
bsdlabell -e da0s1
```

Read the on-disk label for *da0s1*, edit it, and display what the new label would be (in sectors). It does

not install the new label either in-core or on-disk:

```
bsdlabel -e -n da0s1
```

Write a default label on *da0s1*. Use another **bsdlabel -e** command to edit the partitioning and file system information:

```
bsdlabel -w da0s1
```

Restore the on-disk and in-core label for *da0s1* from information in *saveditlabel*:

```
bsdlabel -R da0s1 saveditlabel
```

Display what the label would be for *da0s1* using the partition layout in *label_layout*. This is useful for determining how much space would be allotted for various partitions with a labeling scheme using %-based or * partition sizes:

```
bsdlabel -R -n da0s1 label_layout
```

Install a new bootstrap on *da0s1*. The boot code comes from */boot/boot*:

```
bsdlabel -B da0s1
```

Install a new label and bootstrap. The bootstrap code comes from the file *newboot* in the current working directory:

```
bsdlabel -w -B -b newboot /dev/da0s1
```

Completely wipe any prior information on the disk, creating a new bootable disk with a DOS partition table containing one slice, covering the whole disk. Initialize the label on this slice, then edit it. The `dd(1)` commands are optional, but may be necessary for some BIOSes to properly recognize the disk:

```
dd if=/dev/zero of=/dev/da0 bs=512 count=32
gpart create -s MBR da0
gpart add -t freebsd da0
gpart set -a active -i 1 da0
gpart bootcode -b /boot/mbr da0
dd if=/dev/zero of=/dev/da0s1 bs=512 count=32
bsdlabel -w -B da0s1
bsdlabel -e da0s1
```

This is an example disk label that uses some of the new partition size types such as %, M, G, and *, which could be used as a source file for "bsdlabel -R ada0s1 new_label_file":

```
# /dev/ada0s1:

8 partitions:
#   size  offset  fstype  [fsize bsize bps/cpg]
a: 400M   16    4.2BSD  4096 16384  75 # (Cyl. 0 - 812*)
b: 1G     *     swap
c: *     *     unused
e: 204800 *    4.2BSD
f: 5g    *    4.2BSD
g: *     *    4.2BSD
```

DIAGNOSTICS

The kernel device drivers will not allow the size of a disk partition to be decreased or the offset of a partition to be changed while it is open.

COMPATIBILITY

Due to the use of an *uint32_t* to store the number of sectors, BSD labels are restricted to a maximum of $2^{32}-1$ sectors. This usually means 2TB of disk space. Larger disks should be partitioned using another method such as `gpart(8)`.

The various BSDs all use slightly different versions of BSD labels and are not generally compatible.

SEE ALSO

`ccd(4)`, `geom(4)`, `md(4)`, `disktab(5)`, `boot0cfg(8)`, `gpart(8)`, `newfs(8)`

HISTORY

The **disklabel** utility appeared in 4.3BSD-Tahoe.