Device Naming Conventions

Block Devices

FreeBSD creates block device names based on type, similar to the way Linux does. A device name is a 2-3 letter indicator of device type, followed by an integer showing its hierarchy in the initial search. **Note:** The device number can change if a new device of the same type is added, so use labels in fstab.

FreeBSD creates block device names based on type, similar to the way Linux does.

Identifer	Description
ada	SATA and IDE Hard Drives
da	SCSI hard drives and USB storage
cd or scd	SATA and IDE CD-ROM Drives
cd	SCSI CD-ROM Drives
fd	Floppy Drives

Devices are then broken down into slices (“partitions” in the Linux and Windows worlds), designated by the letter “s” followed by a number (max of 4). Thus, *da0s1* would be the first slice on a USB or SCSI hard drive.

Each slice can have a partition (no relationship to Linux and Windows partitions) which is designated by a single letter. *da0s1a* would be the first partition in the first slice on a USB or SCSI hard drive

By convention, partition “a” is a root file system, partition “b” is swap space, partition “c” is a special partition used by the file system management. All other partitions are available for other things (ie, /var, etc…)

Partitions can hold file systems. NOTE: by “partition” I mean a partition of a slice (ie, BSD terms, not Windows/Linux)

See https://www.freebsd.org/doc/handbook/disk-organization.html

Disks can be managed by either fdisk (old school appearantly) or gpart.

Ethernet Devices

Ethernet devices are named based on the drive needed to access it and the order in which it was discovered. So, for example, a NIC which uses the ndis drive would be called ndis0. If a second card was found with the same driver, it would be named ndis1.

You can see the ethernet cards recognized by the system with the command:

```
grep 'Ethernet address:' /var/run/dmesg.boot
```

Note that this will **not** find cards which do not have a driver loaded, although the most common ones are built into the GENERIC kernel.

RAMDisk

RAMDisks are handled a little differently in FreeBSD in that you will want to manually create a directory (say, /mnt/ramdisk) and then execute the appropriate *mdfs* command. Once that is done, you can use RAMDisks as you like. The fstab entry is different from the one used in Linux, obviously.

```
# fstab entry creating a RAMDISK mount for /tmp of 16M size.
md              /tmp            mfs     rw,-s16M,nosuid,noatime,noexec 0
0
```