USB

Basic process:

Get a good, that either works with your existing controller or also get a controller with is. eg. If it is SCSI and you only have SATA you will need another controller, etc.

Install it in the box and boot the machine.

Observe the boot messages or use dmesg to make sure it is recognized and that you know how the system is identified. It will probably be either da1 or ad1 depending on whether it is SCSI or SATA.

Decide on how you want the new disk divided.

Use fdisk to create a FreeBSD slice on the disk

Use bsdlabel (or disklabel for 4.xxx systems and before) to create at least one partition within the slice.

Use newfs to build a filesystem on all of the partitions you create with bsdlabel.

create a mount point (for example 'mkdir /newdsk')

Edit the /etc/fstab file to add an entry to make it mount upon boot.

Mount the new disk (For example 'mount /newdsk' or just 'mount -a')

The man pages for fdisk, bsdlabel and newfs have all the information you need but can be a little confusing. In the bsdlabel man page there are some examples near the bottom that are good. They suggest using dd to overwrite anything that might preexist in the first sector. That isn't always needed, but can fix things if that sector is a problem. Actually, I usually write around the first 100 sectors just for good measure when I need it.

The only more complicated things are if you want to make more than one partition and/or slice, and if you want to be able to boot from it. More slices and partitions amounts to the same, but just require some calculations. Making it bootable requires using -B on both fdisk and bsdlabel.

All this can also be done using sysinstall, but I prefer doing it straight up with the regular tools.

Read the man pages.

Good luck,

---

http://lists.freebsd.org/pipermail/freebsd-questions/2006-June/123638.html

http://www.freebsd.org/doc/en/books/handbook/disks-adding.html 18.3.2.2 Dedicated

If you will not be sharing the new drive with another operating system, you may use the dedicated mode. Remember this mode can confuse Microsoft operating systems; however, no damage will be done by them. IBM's OS/2(R) however, will "appropriate" any partition it finds which it does not understand.

dd if=/dev/zero of=/dev/da1 bs=1k count=1 # bsdlabel -Bw da1 auto # bsdlabel -e da1 # create the 'e' partition # newfs /dev/da1e # mkdir -p /1 # vi /etc/fstab # add an entry for /dev/da1e # mount /1

Inline interpreted text or phrase reference start-string without end-string.

bsdlabel -e da0

used to say # /dev/da0: 8 partitions: # size offset fstype [fsize bsize bps/cpg]

Unexpected indentation.

a: 15646704 16 unused 0 0 c: 15646720 0 unused 0 0 # "raw" part, don't edit

now I adjusted it

 # size offset fstype [fsize bsize bps/cpg] a: 1048576 16 unused 0 0 c: 15646720 0 unused 0 0 # "raw" part, don't edit e: 10485760 1048592 4.2BSD 2048 16384 28552

actually I entered e: 5G * 4.2BSD it did the rest of the calculatoions
amazingly it worked