Exercise 1 – FreeBSD Installation

Announced Date: 2006/9/20

Due Date: 2005/10/4

Outline

- FreeBSD version
- Installing FreeBSD
- Update source and make world
- Rebuild kernel

FreeBSD branches

- Two parallel development branches:
  - *-STABLE*
    - **Receive only well-tested bug fixes and other small incremental enhancement**
    - Latest Release version:
      - ***5.5 May, 2006***
      - ***6.1 May, 2006***
  - *-CURRENT*
    - **Latest working sources for FreeBSD**
    - Latest Release version:
      - ***7.0-CURRENT***

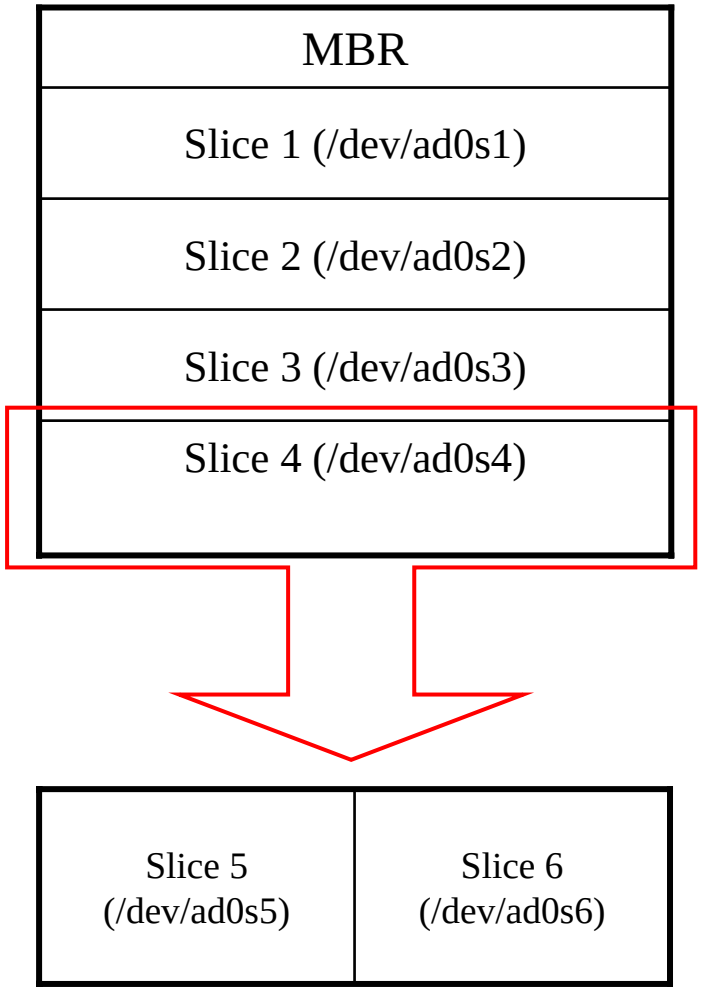
FreeBSD version

- A.B.C – Type
  - A: major version Number
  - B: minor version Number
  - C: slight patch version number
  - Type: version type
    - SNAP
    - ALPHA、BETA、GAMMA
    - RELEASE
    - RELENG
    - STABLE
    - CURRENT

Snapshot → { Alpha / Beta / Gamma } → Release → Releng → Stable

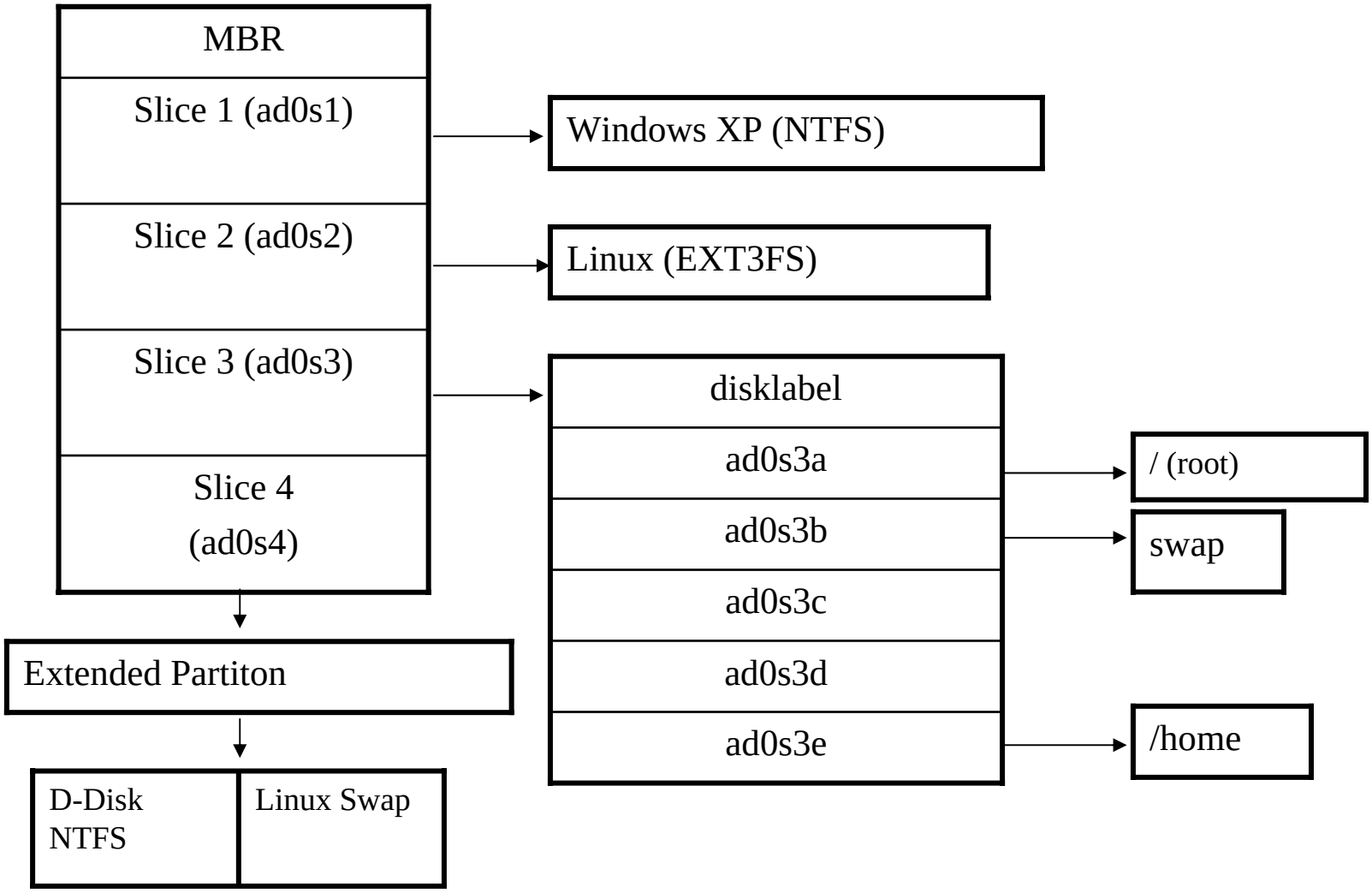
FreeBSD view of Disk (1)

- What is the meaning of **ad0s1e**
  - Disk name
    - IDE: ad
    - SCSI: da
  - Slice is equal to the partition of common use
    - Primary partition: s1 ~ s4
    - Extended partition: s5 ~ s*n*
  - Label in each slice
    - a: root partition
    - b: swap
    - c: entire disk
    - d: entire partition
    - efgh: /usr, /home, ...



FreeBSD view of Disk (2)

- An Example



Installing FreeBSD

- Steps
  1. Knowing your hardware
  2. Obtaining installation file
  3. Booting from CD
  4. sysinstall main menu
  5. Custom Installation Options
     1. Partition
     2. Label
     3. Distribution
     4. Media
     5. Commit
  6. Post Installation

Installing FreeBSD – 1. knowing your hardware

- CPU
  - 32bit or 64bit
  - Intel、AMD or other brand
  - Pentium-II, Pentium4, Xeon, XP1700+, Optron, C3
- RAM
  - Size
- HD
  - Size, amount, SCSI or IDE
- VGA
  - Brand, ram size
- Sound
  - Brand
- Network Interface Card
  - Brand
  - IP、Netmask、default gateway、Hostname、DNS
- Other Special device

Installing FreeBSD – 2. Obtaining installation file

- FreeBSD installation CD
  - ftp://freebsd.csie.nctu.edu.tw/pub/ISO-IMAGES-i386/6.1/6.1-RELEASE-i386-bootonly.iso
  - ftp://freebsd.csie.nctu.edu.tw/pub/ISO-IMAGES-i386/6.1/6.1-RELEASE-i386-disc1.iso
  - Burn!
- Boot Floppy Image
  - ftp://freebsd.csie.nctu.edu.tw/pub/releases/i386/6.1-RELEASE/floppies/boot.flp
  - ftp://freebsd.csie.nctu.edu.tw/pub/tools/fdimage.exe
  - C:\fdimage.exe boot.flp a:\

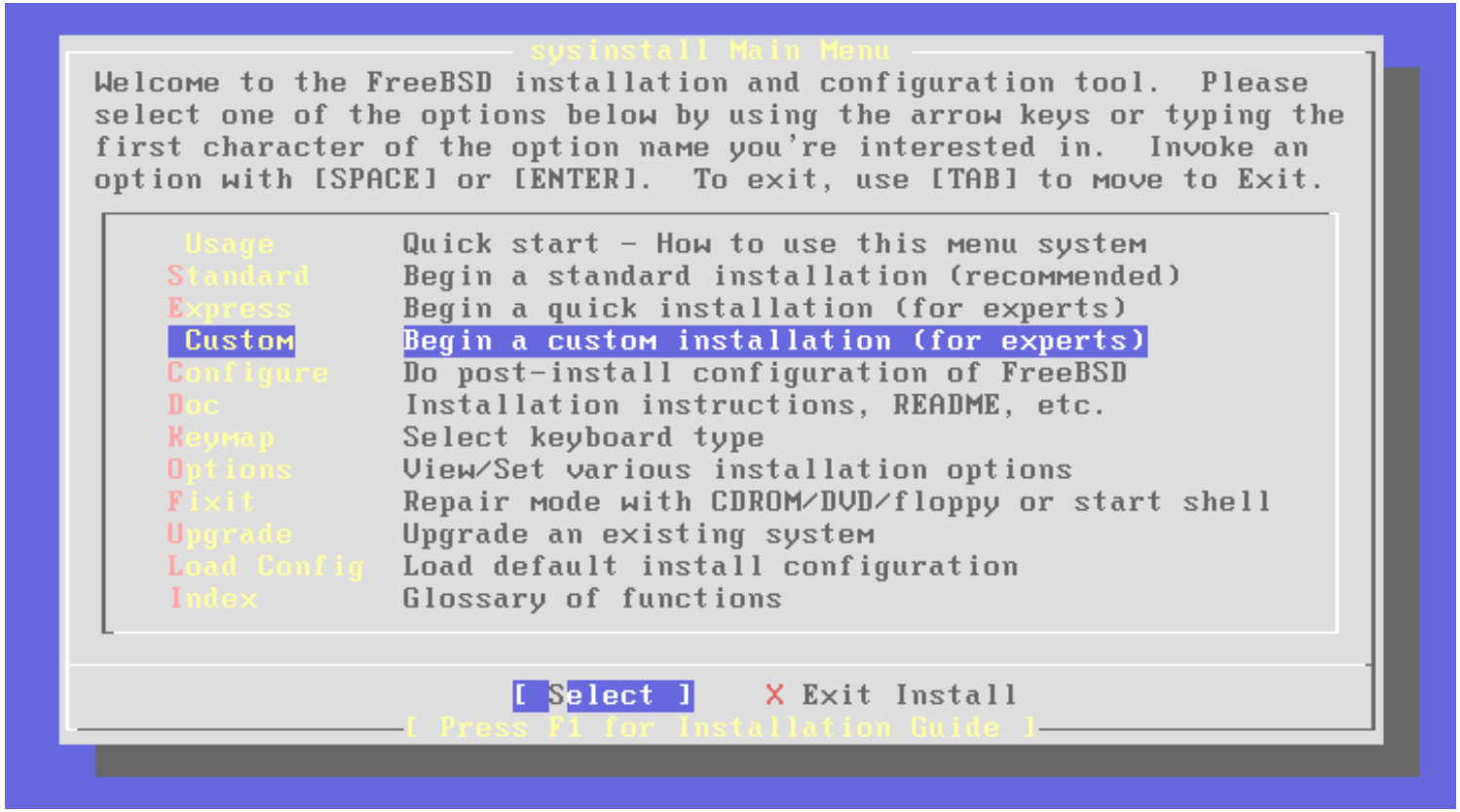
Installing FreeBSD – 3. Booting from CD

```
CD Loader 1.2

Building the boot loader arguments
Looking up /BOOT/LOADER... Found
Relocating the loader and the BTX
Starting the BTX loader

BTX loader 1.00  BTX version is 1.01
Consoles: internal video/keyboard
BIOS CD is cd0
BIOS drive A: is disk0
BIOS drive C: is disk1
BIOS 638kB/128960kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1
(root@opus.cse.buffalo.edu, Sun May  7 03:20:03 UTC 2006)
Loading /boot/defaults/loader.conf
/boot/kernel/kernel text=0x4c8c74 \
```

Installing FreeBSD – 4. sysinstall Main Menu

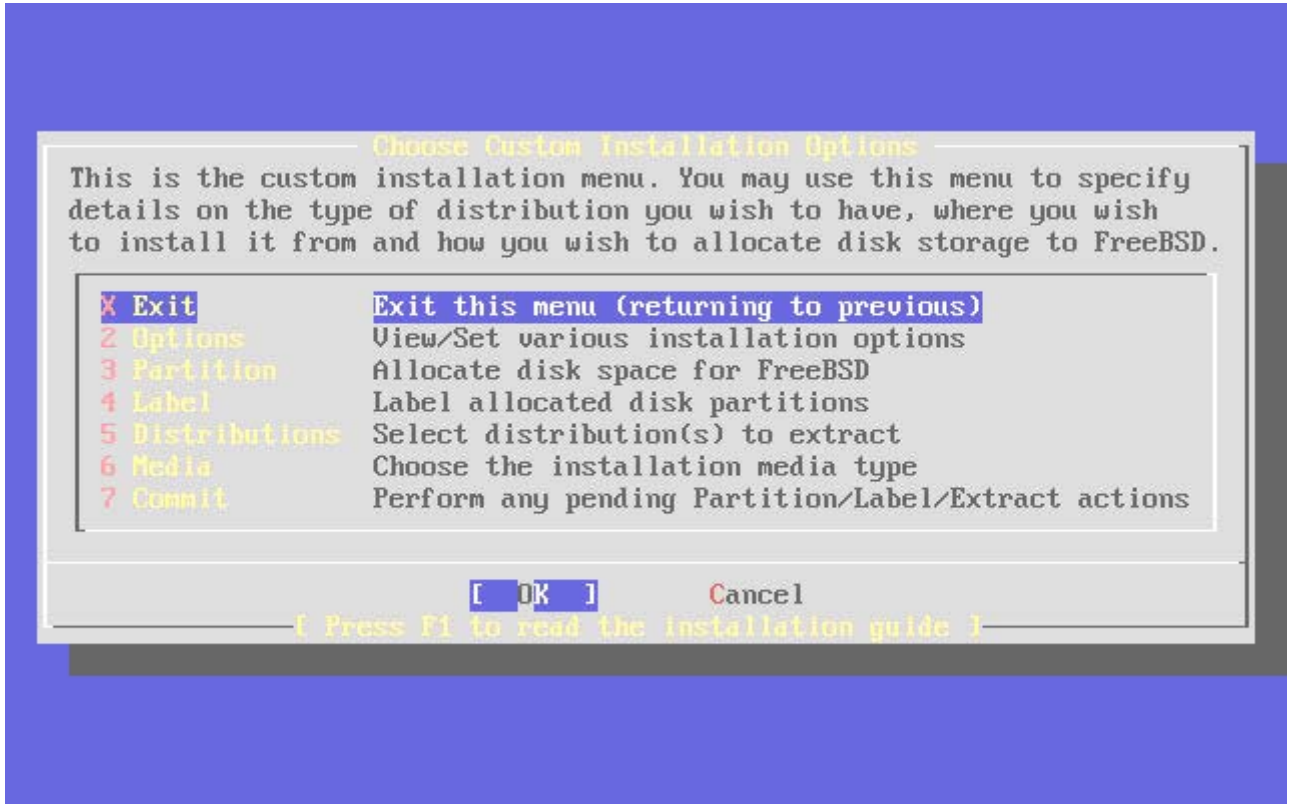
You can press “Scroll Lock” key to see probe results.

- sysinstall Main Menu
  - Contry Selection “Taiwan”
  - Console Keymap “USA ISO”
  - Choose “Custom”



Installing FreeBSD – 5. Custom Installation Options

- 5 major steps
  1. Partition your disk
  2. Label your disk
  3. Choose what to install
  4. Choose how to install
  5. Commit



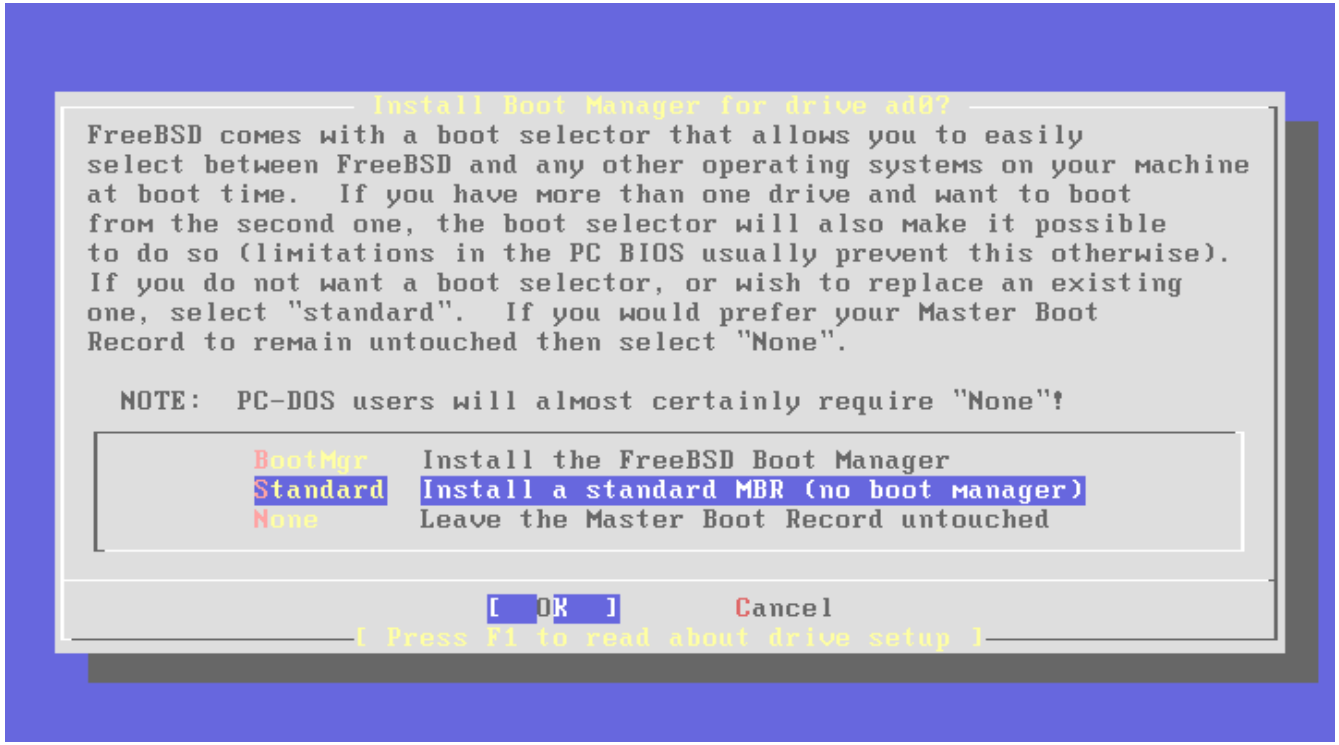
Installing FreeBSD – 5. Custom Installation – partition (1)

- Create slice and choose boot manager
  - Press “C” to create a new slice or press “A” to use entire disk
  - Press “S” to toggle ad0s1 as bootable (we will put / on this slice)
  - Press “Q” to next step (Select Boot Manager)

```
Disk name:      ad0                                FDISK Partition Editor
DISK Geometry:  43496 cyls/15 heads/63 sectors = 41103720 sectors (20070MB)

Offset       Size(ST)        End      Name  PType       Desc  Subtype    Flags

         0          63         62         -     12     unused        0
        63    41103657   41103719     ad0s1      8     freebsd     165     A
  41103720         459   41104178         -     12     unused        0


The following commands are supported (in upper or lower case):

A = Use Entire Disk   G = set Drive Geometry   C = Create Slice   F = 'DD' mode
D = Delete Slice      Z = Toggle Size Units     S = Set Bootable   ! = Wizard m.
T = Change Type       U = Undo All Changes      Q = Finish

Use F1 or ? to get more help, arrow keys to select.
```

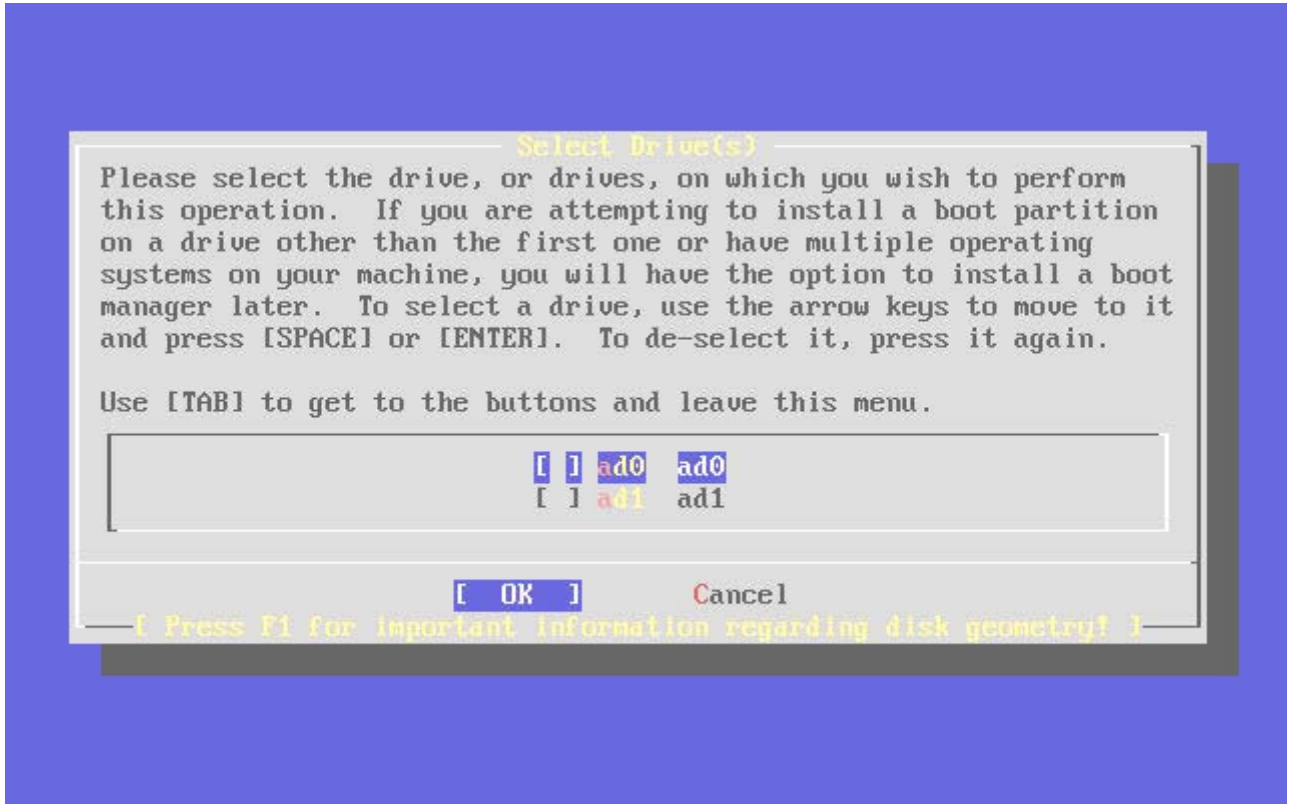
Installing FreeBSD – 5. Custom Installation – partition (2)

- Select “Standard” for ad0
  - BootMgr ➔ Multiple OS
  - Standard ➔ Single OS
  - None ➔ Other BM
- After press OK, it will back to Custom Installation Options menu



Installing FreeBSD – 5. Custom Installation – partition (3)

- If you have more than one disk...
  - You can choose whether to partition it.
  - Install “BootMgr” for first disk and “none” for rest ones

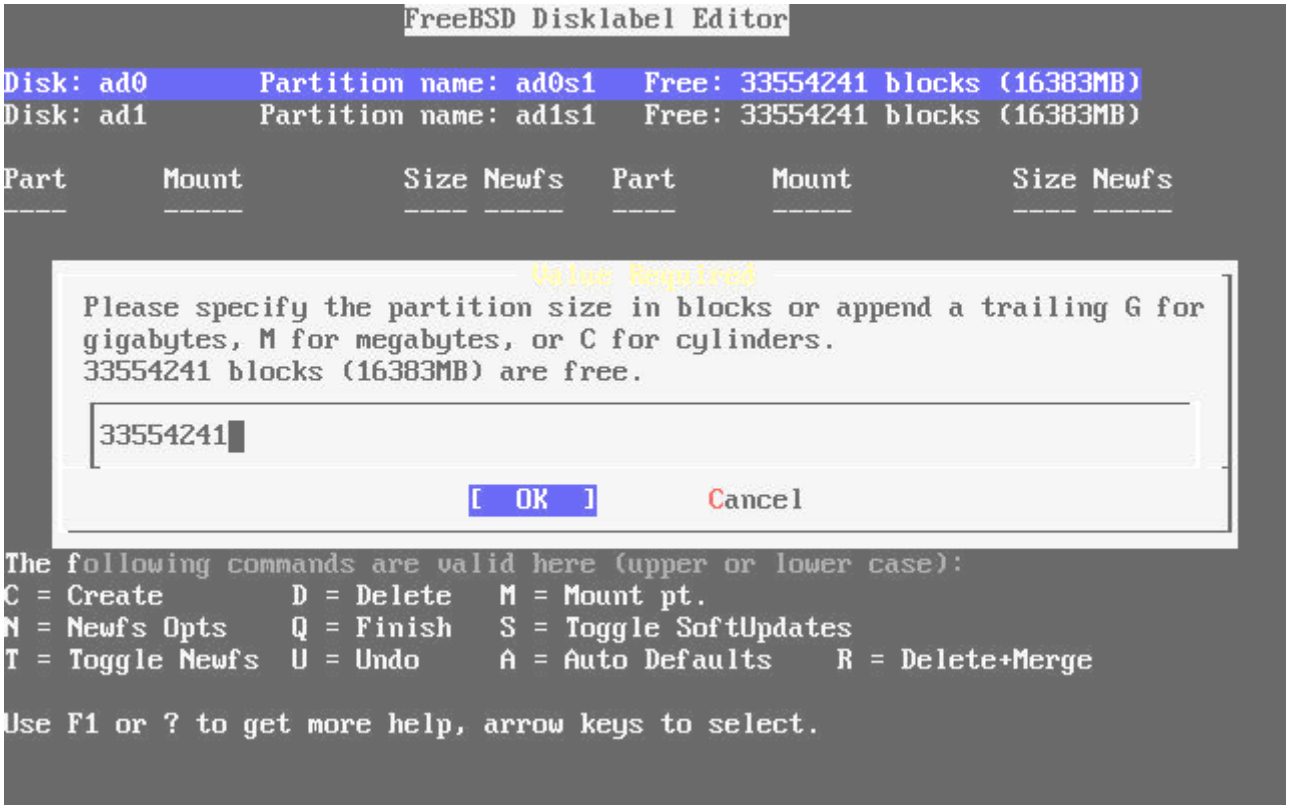


Installing FreeBSD – 5. Custom Installation – Label (1)

- Disklabel Editor
  - Move blue bar to select slice
  - Press “C” to create disk label
    - swap , / (and /home, /var, ...)
      - Specify size
      - Choose type (either swap or FS)
      - Specify mount point
  - Press “S” to toggle SoftUpdates (async written to disk)
  - Press “Q” to next step (back to custom installation options menu)

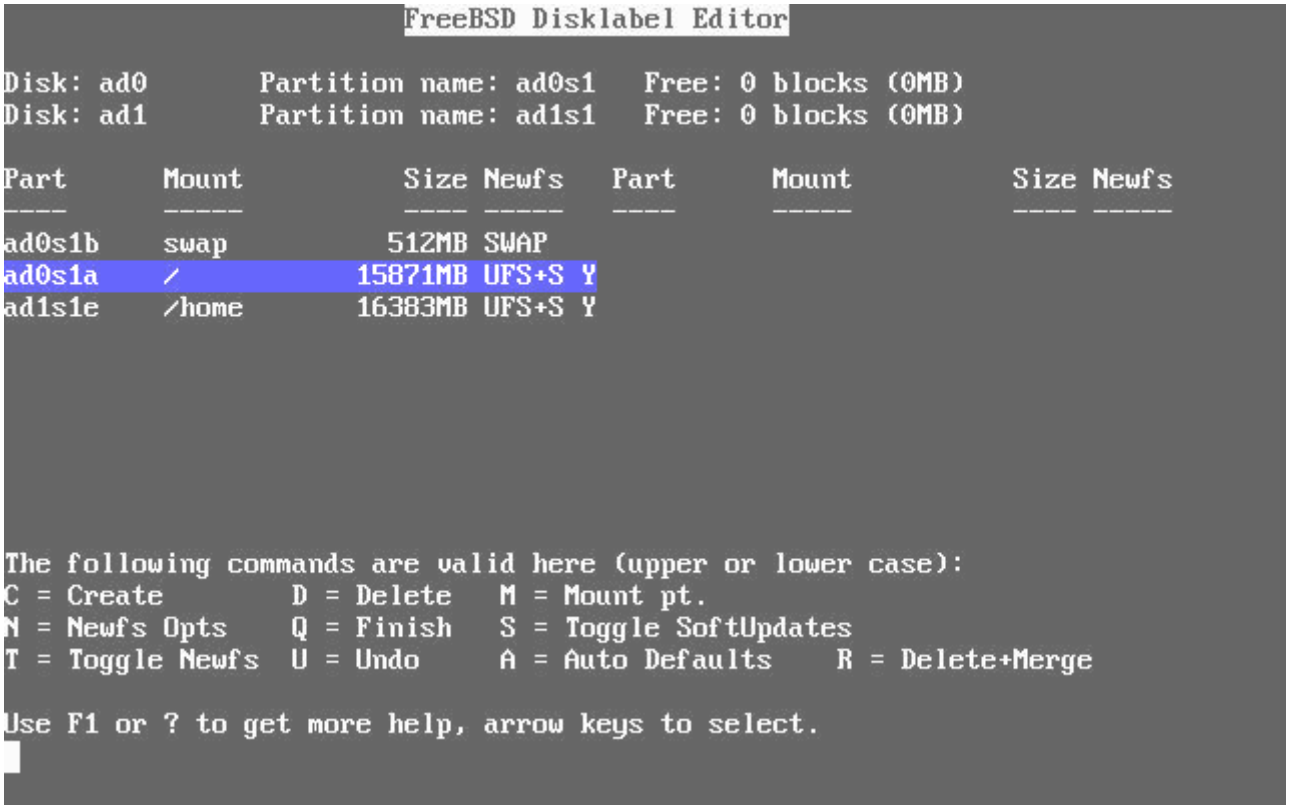
Installing FreeBSD – 6. Custom Installation – Label (2)

- Create label in ad0 and specify size



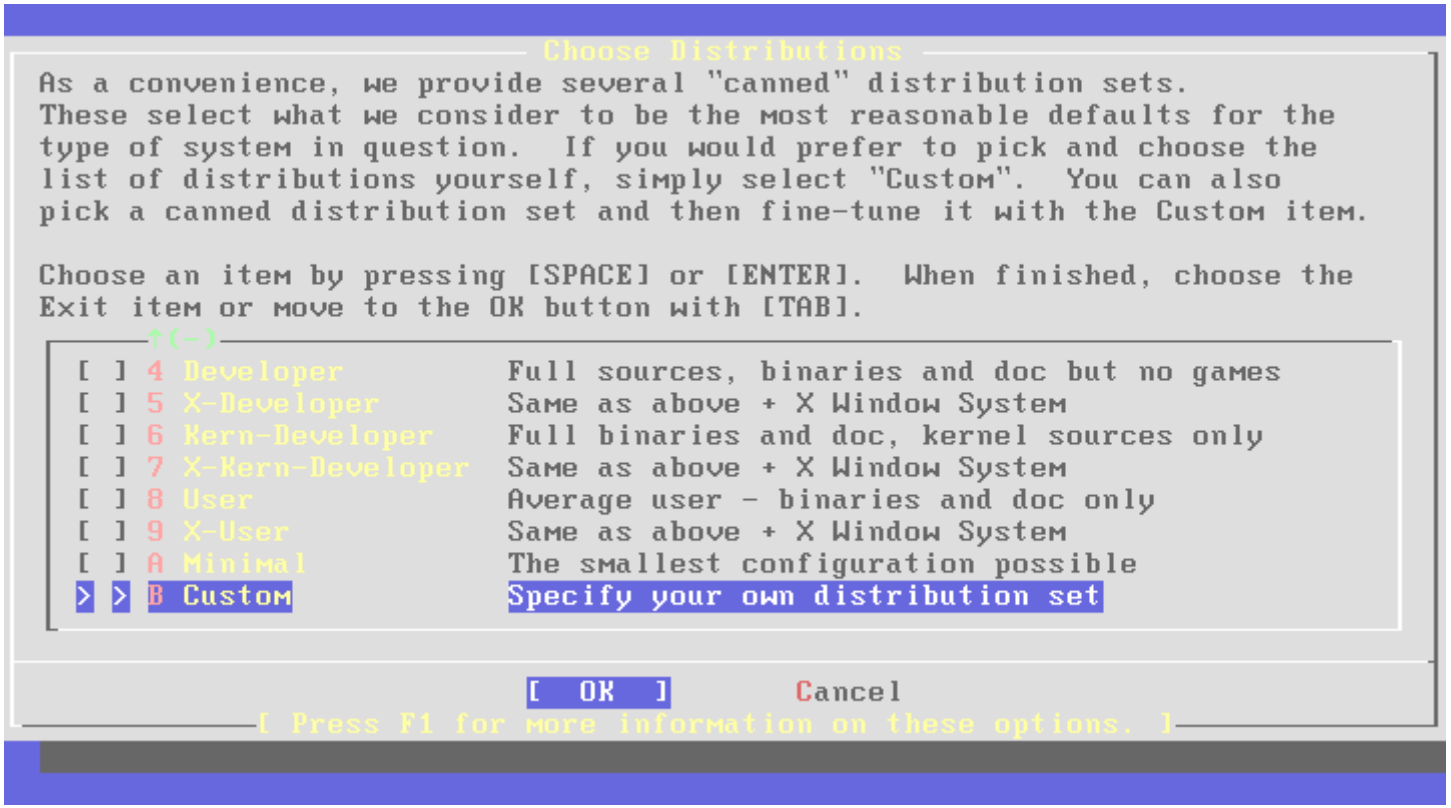
Installing FreeBSD – 6. Custom Installation – Label (3)

- Complete disklabel



Installing FreeBSD – 6. Custom Installation – distri. (1)

- Choose Distributions Menu
  - Choose “Custom”

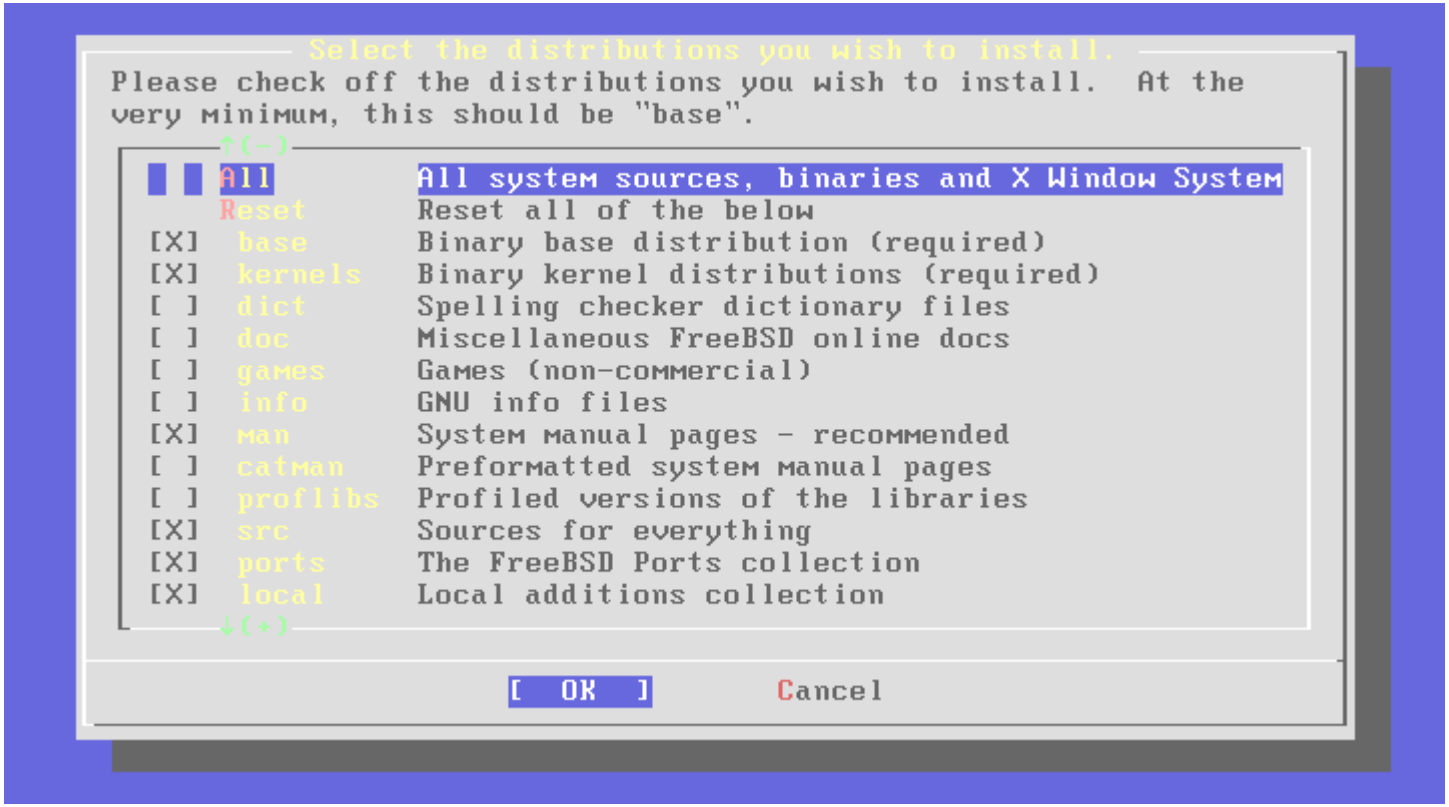


Installing FreeBSD – 6. Custom Installation – distri. (2)

- Select
  - base ; **binary**
  - kernels(all) ; **Generic kernel and SMP kernel**
  - Man ; **Manual**
  - src ; **FreeBSD Source Code**
  - ports ; **FreeBSD Software Collection**
  - local ; **local additions**

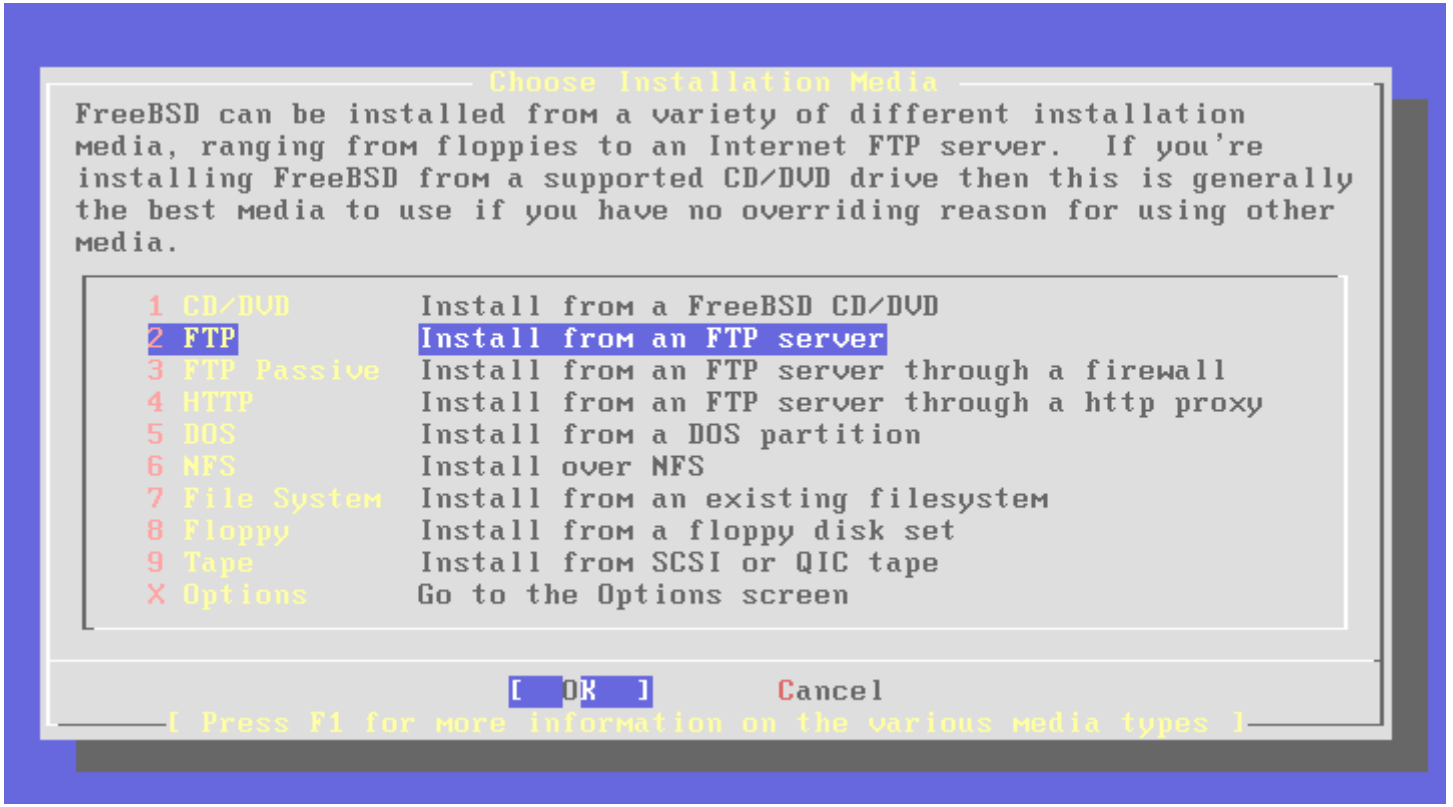
- After Selection
  - Press “OK” and it will return to “Choose Distributions menu”
  - Press “OK” again to back to “Custom Installation Options menu”
  - Select “Media”

Installing FreeBSD – 6. Custom Installation – distri. (3)



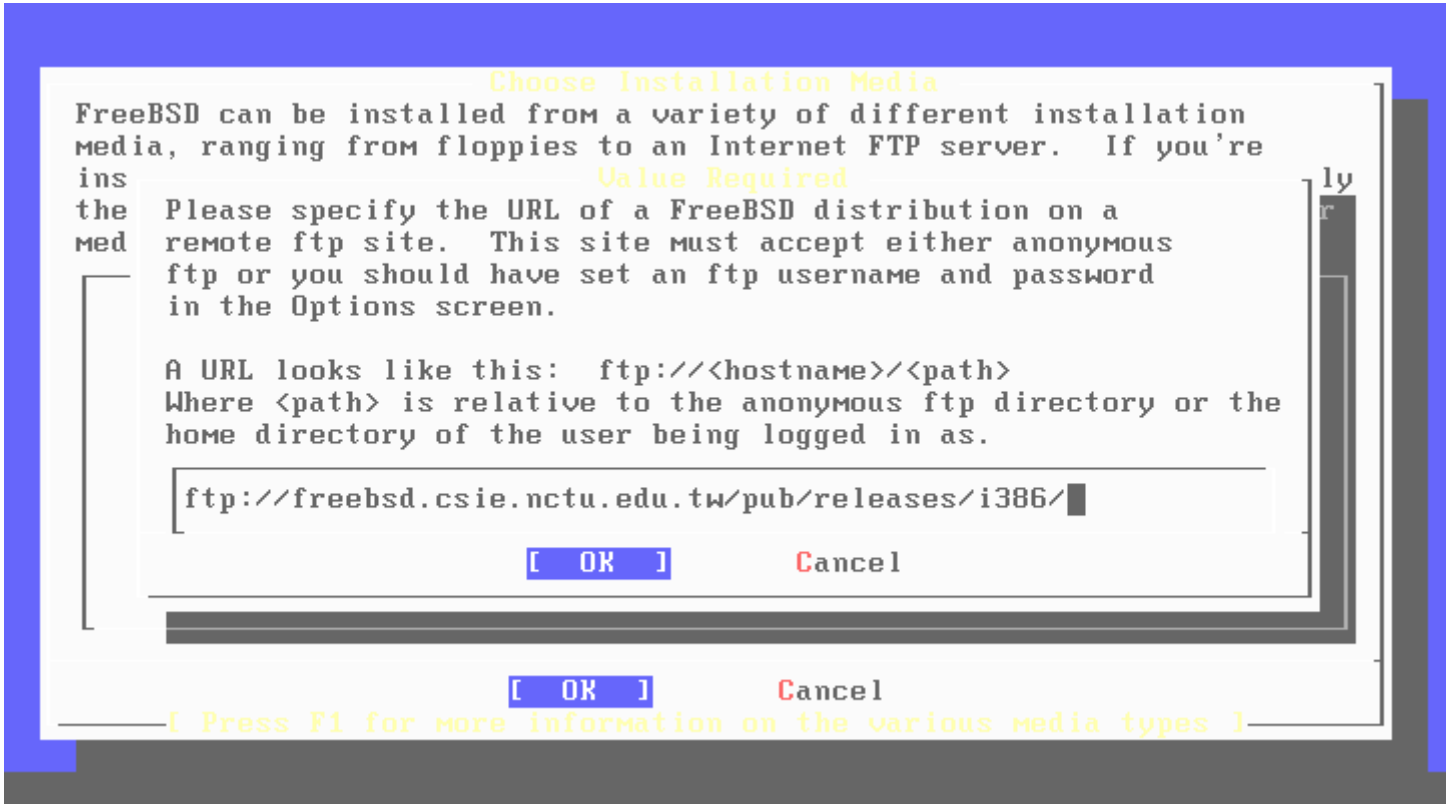
Installing FreeBSD – 6. Custom Installation – Media (1)

- Choose CD/DVD if you have 6.1 Stable CD
- Choose FTP if your NIC is detected
  - Choose FTP Passive if you in private network



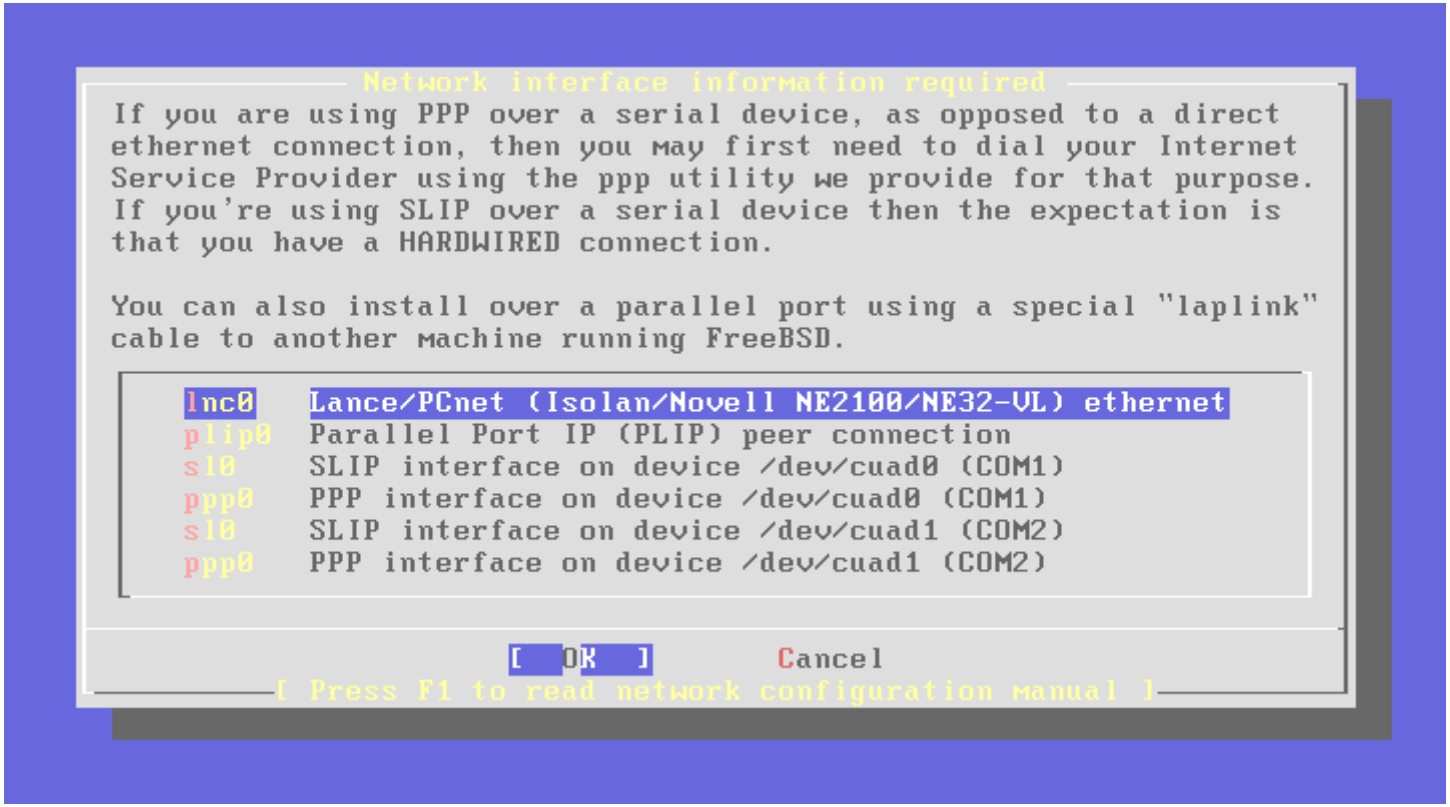
Installing FreeBSD – 6. Custom Installation – Media (2)

- Install through FTP
  - Specify ftp server and path



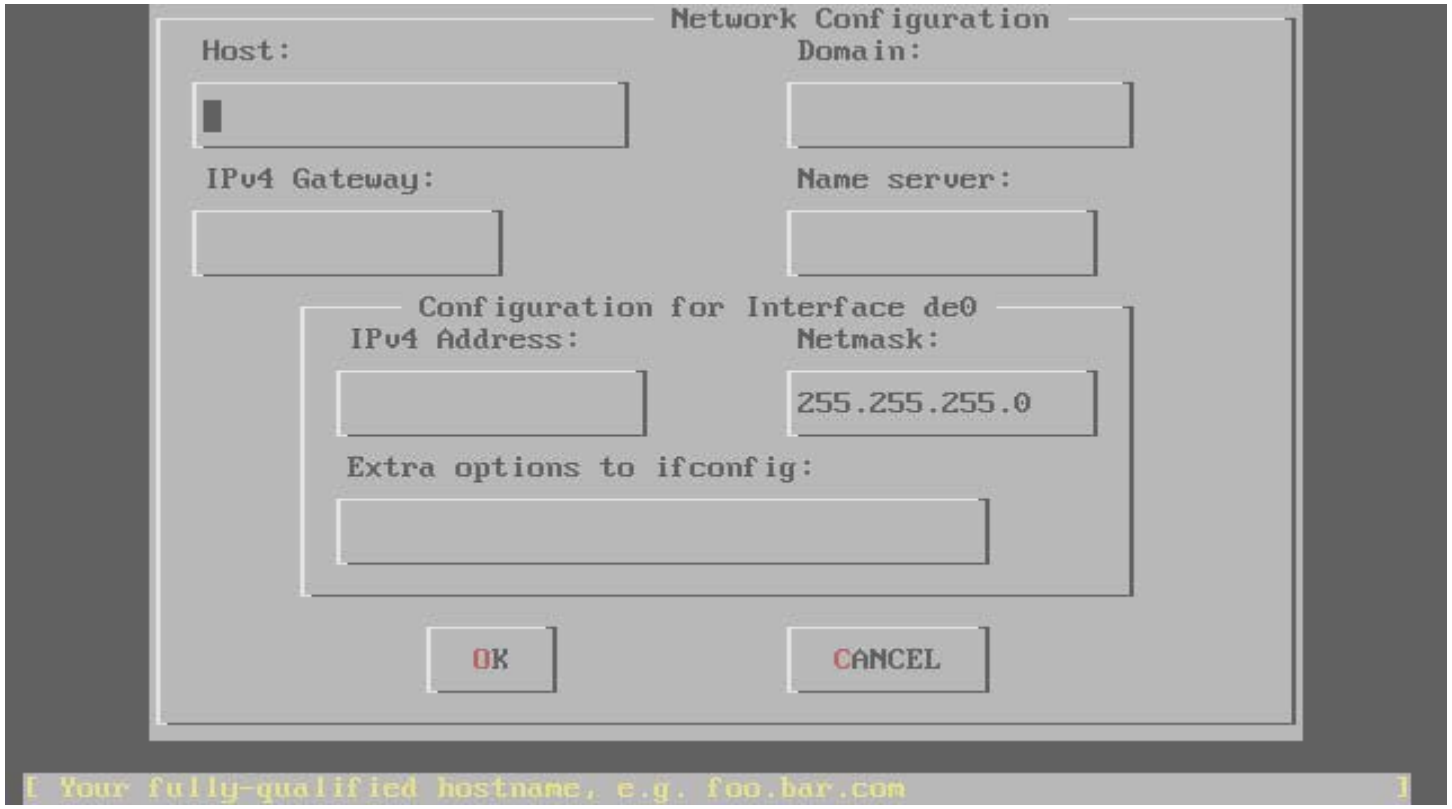
Installing FreeBSD – 6. Custom Installation – Media (3)

- Select NIC
- IPv6 and DHCP



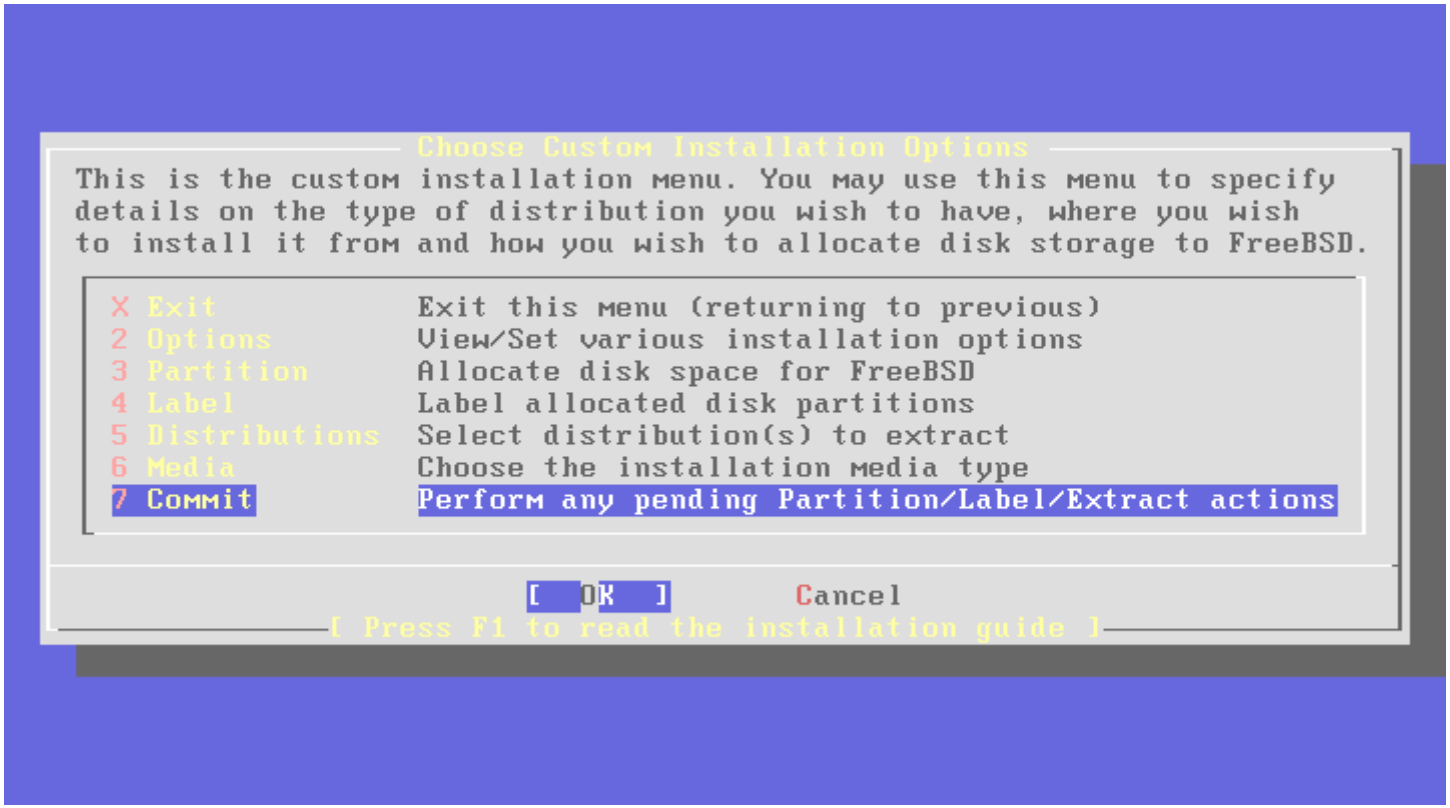
Installing FreeBSD – 6. Custom Installation – Media (4)

- Specify your IP information
- Press “OK” to next step

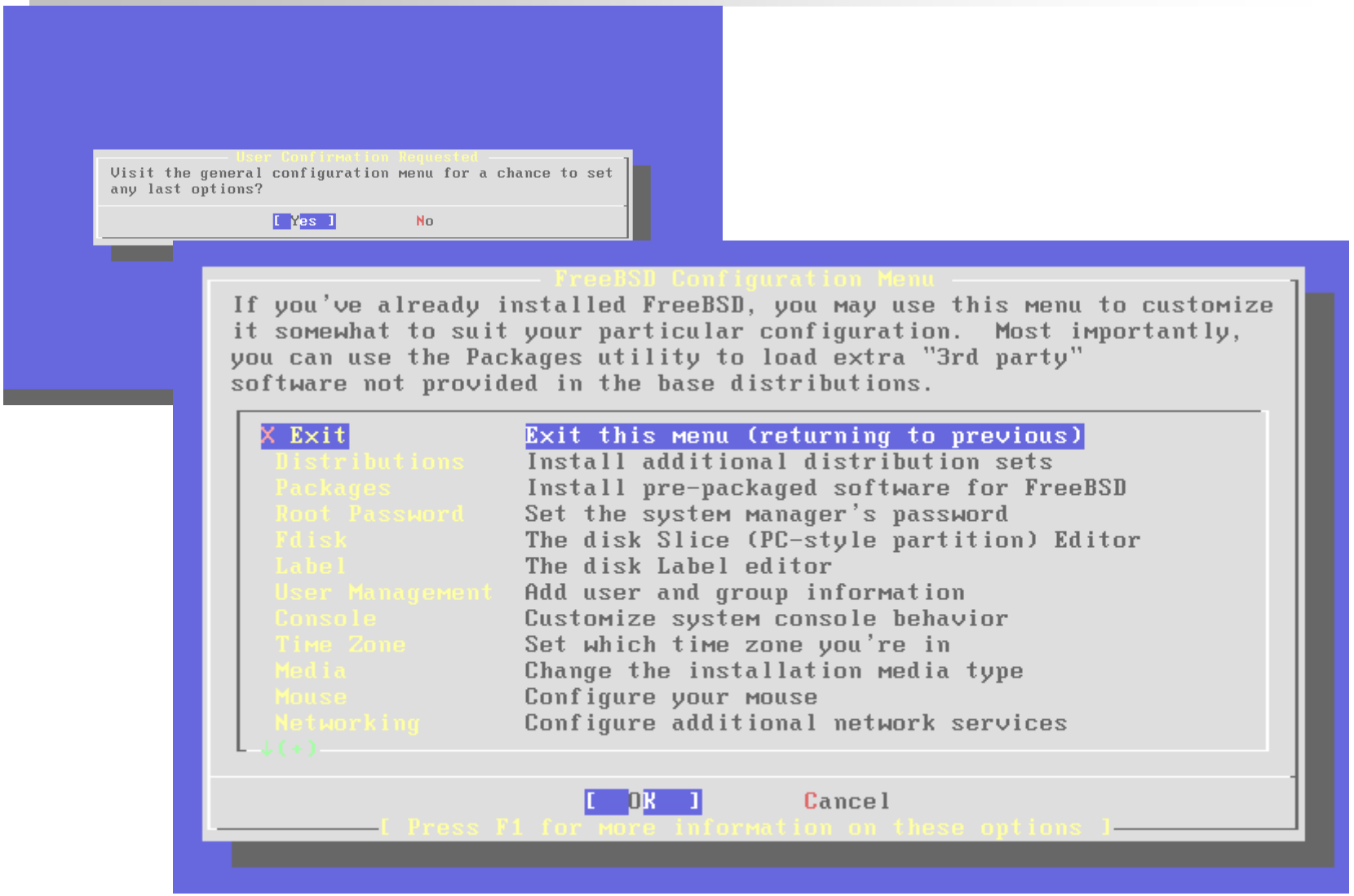


Installing FreeBSD – 6. Custom Installation – Commit

- Start to format disk、make file system and install software
- You can press “Alt + F2” to see the install detail

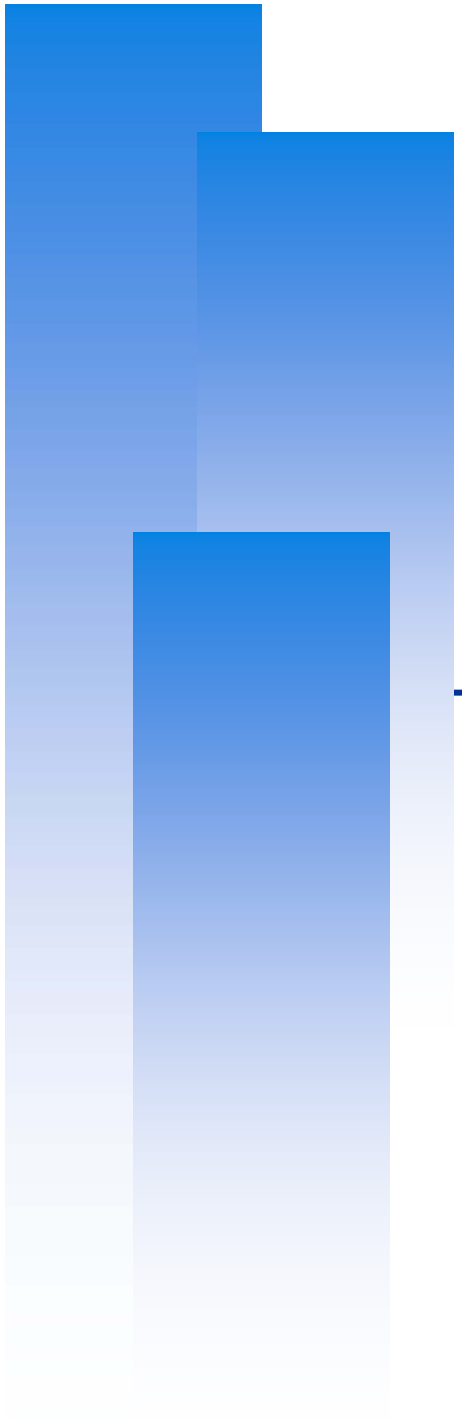


Installing FreeBSD – 7. Post Installation (1)



Installing FreeBSD – 7. Post Installation (2)

- Root Password
- Time Zone → Asia → Taiwan
- Mouse → enable
- Networking → sshd
- Packages (optional)
  - net/csup
  - editors/vim-lite



Exercise 1 –
FreeBSD build world and kernel

FreeBSD source

- Maintained in a CVS repository in California
  - Two softwares to get the latest FreeBSD source
    - CVSup
    - CSup – A rewrite of the CVSup file updating client in C
- We can use CSup keep our FreeBSD source up-to-date with any FreeBSD mirror sites
  - Install CSup
  - Edit CSup supfile
  - Update source using CSup
- Up-to-date your system
  - Build world & kernel using the updated source.
  - Install softwares.
  - Upgrade softwares.

FreeBSD source CSup Installation

- Install via pkg_add
  - **Package is pre-compiled application**
  - % pkg_add ftp://freebsd.csie.nctu.edu.tw/pub/releases/i386/6.1-RELEASE/packages/net/csup-20060223_1.tbz
  - **pkg_add** package-name
  - **pkg_delete** package-name
  - **pkg_info** package-name
  - All installed package is stored in /var/db/pkg
- The csup binary is in /usr/local/bin/csup
  - You can use “whereis” command to find something

FreeBSD source
CSup Configuration file (1)

- Example csup supfile
  - /usr/share/examples/cvsup/stable-supfile
  - /usr/share/examples/cvsup/ports-supfile
- Create your own supfile
  - Edit /usr/local/etc/cvsup-src
  - Edit /usr/local/etc/cvsup-ports

FreeBSD source
CSup Configuration file (2)

- /usr/local/etc/cvsup-src

*default host=freebsd.csie.nctu.edu.tw	Where to get source
*default base=/usr	Where to put status file
*default prefix=/usr	Where to put source
*default delete use-rel-suffix	Allow cvs to delete
*default compress	Compress before transmit
*default release=cvs tag=RELENG_6	
src-all	

FreeBSD source
CSup Configuration file (3)

- CVS tags
  - Branch Tags
    - . (FreeBSD-CURRENT line)
    - RELENG_6 (FreeBSD 6-STABLE line)
    - RELENG_5 (FreeBSD 5-STABLE line)
  - Release Tags
    - RELENG_6_1 (FreeBSD 6.1-RELEASE)
    - RELENG_5_5 (FreeBSD 5.5-RELEASE)
  - To refer to a specific point in time
    - RELENG_6_1_0_RELEASE
    - RELENG_5_5_0_RELEASE

http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/cvs-tags.html

FreeBSD source
CSup Configuration file (4)

- /usr/local/etc/cvsup-ports

```
*default host=freebsd.csie.nctu.edu.tw
*default base=/usr
*default prefix=/usr
*default delete use-rel-suffix
*default compress
*default release=cvs tag=.
ports-all
```

FreeBSD source
CSup Configuration file (5)

- you can put them all together in /usr/local/etc/cvsup-all

```
*default host=freebsd.csie.nctu.edu.tw
*default base=/usr
*default prefix=/usr
*default delete use-rel-suffix
*default compress
*default release=cvs tag=RELENG_6
src-all
ports-all tag=.
```

FreeBSD source

Update source using CSup

- Update both src and ports
  - % /usr/local/bin/csup -L 1 /usr/local/etc/cvsup-all > /var/log/csup.log

---

The “–L 1” tells cvsup to print out the details of all the file updates it is doing.
from 0 (silent) to 2

Rebuilding world & kernel

- The canonical way to update system
  - make buildworld
  - make buildkernel
  - make installkernel
  - reboot and boot in single user mode
  - mergemaster -p
  - make installworld
  - mergemaster
  - reboot

Rebuilding world & kernel – Prepare make.conf

- Example make.conf
  - /usr/share/examples/etc/make.conf 5.x ~ 6.x
  - Everything add in make.conf is used every time you run make
  - KERNCONF=CHBSD
  - CPUTYPE?=pentium4

---

man make.conf has detail descriptions of it

Available CPUTYPE:

See /usr/share/mk/bsd.cpu.mk

Rebuilding world & kernel – make buildworld

- Build FreeBSD entire system
  - % cd /usr/src
  - % make buildworld >& /var/log/world.log &

---

make –j *n* buildworld
Spawn multiple (n) processes to do make.
The compiling processes of make world is I/O bound.

Rebuilding world & kernel – make buildkernel (1)

- Why rebuild kernel?
  - Faster boot time.
    - Probe only necessary device
  - Lower memory usage
    - Smaller kernel image
  - Additional hardware support.

Rebuilding world & kernel – make buildkernel (2)

- Edit kernel config file
  - cd /usr/src/sys/i386/conf
    - GENERIC may not have all for your system
    - LINT has every options

      %use “make LINT” to generte the LINT file
  - cp GENERIC “YOUR-NAME”
    - We often use hostname to be “YOUR-NAME”
  - edit config file
    - Depend on your system
    - Be attention to related options
    - Following the explanation of http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/kernelconfig-config.html

Rebuilding world & kernel – make buildkernel (3)

- Build kernel
  - % cd /usr/src
  - % make KERNCONF=CHBSD buildkernel

If the KERNCONF="YOUR-NAME"
has been set in your make.conf
You can use "make buildkernel" instead

Rebuilding world & kernel – make installkernel

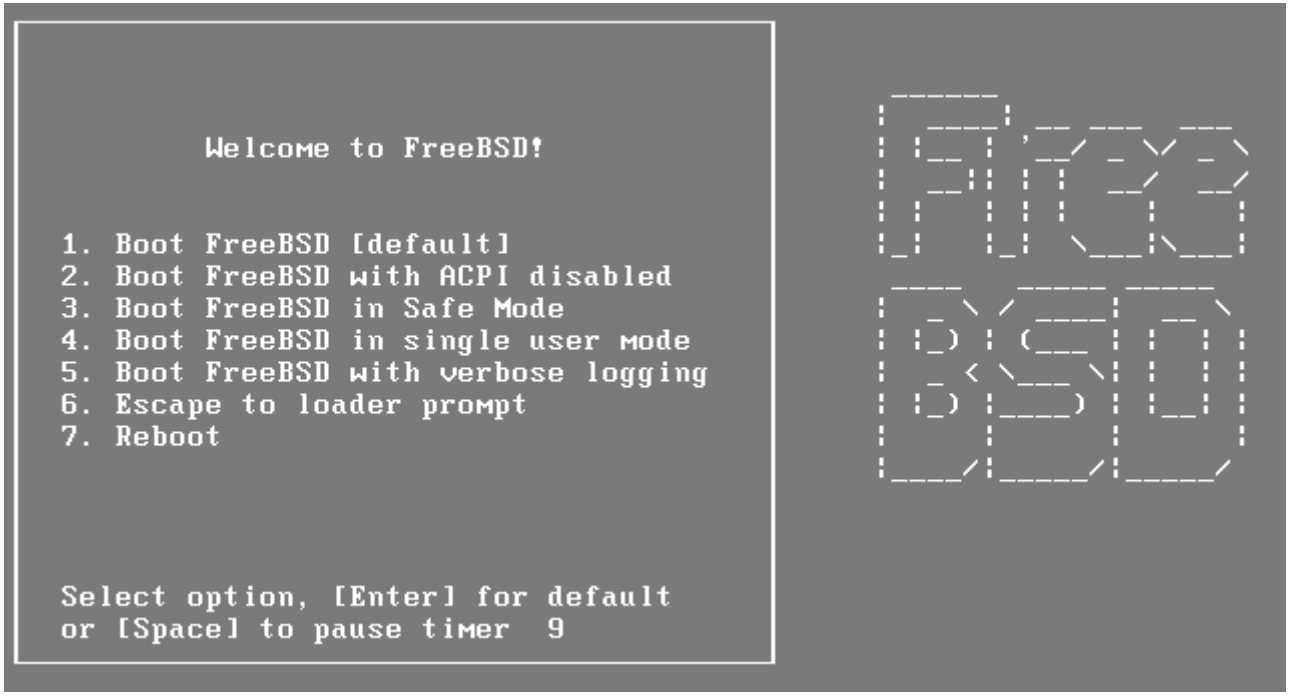
- Install kernel
  - % cd /usr/src
  - % make KERNCONF=CHBSD installkernel

---

make buildkernel && make installkernel
= make buildkernel installkernel
= make kernel

Rebuilding world & kernel – reboot into single user mode

- Boot in single user mode
  - Press “4”



Or

  - % shutdown now
    - For a running system, this will drop it to single user mode

Rebuilding world – make installworld

- mergemaster -p
  - -p

    Pre-buildworld mode. Compares only files known to be essential to the success of {build|install}world, including /etc/make.conf.
- Install the built world
  - % make installworld

Rebuilding world – mergemaster

- mergemaster
  - Synchronize /usr/src/etc with /etc
  - Choose “i” for most case, such as
    - /etc/defaults/rc.conf, ...
  - Press “enter” for certain file, such as
    - master.passwd, hosts, csh.*

Reboot

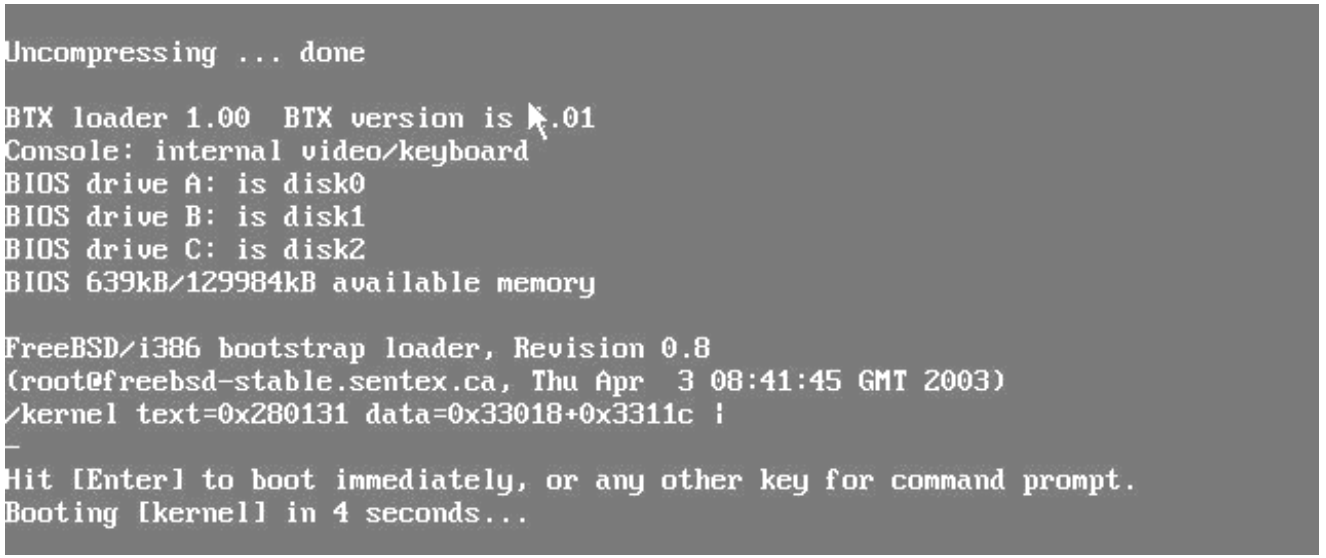
- Reboot and enjoy it
  - % reboot

If Something Goes Wrong ... (1)

- Possible errors in building new kernel
  - Configuration file
    - % cd /usr/src/sys/i386/conf/
    - % config CHBSD
  - make fail
  - Install fail
  - Kernel does not boot
    - Boot with old kernel, recompile kernel
  - Kernel works, but ps does not work
    - Build world

If Something Goes Wrong ... (2)

- Boot with old kernel
  - In 5.x ~
    - Press “6”
    - Type “boot /boot/kernel.old/kernel”
  - In 4.x
    - Hit any key other than “enter” when counting down
    - Type “unload”
    - Type “load /kernel.old”
    - Type “boot”



If Something Goes Wrong ... (3)

- Move working kernel to /boot/kernel
  - % mv /boot/kernel.old /boot/kernel
- For versions of FreeBSD prior to 5.x
  - Unlock kernel
    - % chflags noschg /kernel
    - % cp kernel.old kernel
    - % sync;sync; reboot
  - Lock kernel
    - % chflags schg /kernel
- Use ls –lo to check similar file

---

schg → set the immutable (永遠不變的) flag
ls –o → include file flags in long output

Install software

- Package
  - Pre-built ports
  - pkg_add, pkg_delete, pkg_deinstall, pkg_info, pkg_version
- Ports
  - cd /usr/ports, make search, make install clean
  - ports/sysutils/portupgrade
    - portinstall, portupgrade, portversion
    - **/usr/local/etc/pkgtools.conf**
- Source
  - Tar ball
  - tar xzvf certain-source.tar.gz
  - ./configure
  - make; make install

How to use ports

❑ Using ports

- Steps of install software

  (1) Figure out the path to the software

  – **% cd /usr/ports**

  – **% make search key=mutt**

  – **% cd /usr/ports/chinese/mutt**

  (2) Fetch and compile the source

  – **% make install**

- Uninstall

  ➢ % make deinstall

※Using portupgrade

➢ portinstall ＝ portupgrade –N

➢ pkgtools.conf

How to use ports (1)

- Try to install some software, such as:
  - vim
  - mutt
  - wget