NAME

INSTALL — Installation procedure for NetBSD/mac68k.

CONTENTS

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 10.2_STABLE on the *mac68k* platform. It is available in four different formats titled `INSTALL.`*`ext`*, where *`.ext`* is one of `.ps`, `.html`, `.more`, or `.txt`:

`.ps` PostScript.

`.html` Standard Internet HTML.

`.more` The enhanced text format used on UNIX-like systems by the `more`(1) and `less`(1) pager utility programs. This is the format in which the on-line *man* pages are generally presented.

`.txt` Plain old ASCII.

You are reading the *PostScript* version.

What is NetBSD?

The NetBSD Operating System is a fully functional open-source operating system derived from the University of California, Berkeley Networking Release 2 (Net/2), 4.4BSD-Lite, and 4.4BSD-Lite2 sources. NetBSD runs on many different different system architectures (ports) across a variety of distinct CPU families, and is being ported to more. The NetBSD 10.2_STABLE release contains complete binary releases for most of these system architectures, with preliminary support for the others included in source form. For more information please visit **https://www.NetBSD.org/**.

NetBSD is a completely integrated system. In addition to its highly portable, high performance kernel, NetBSD features a complete set of user utilities, compilers for several languages, the X Window System, firewall software and numerous other tools, all accompanied by full source code.

NetBSD is a creation of the members of the Internet community. Without the unique cooperation and coordination the net makes possible, NetBSD would not exist.

Changes Between The NetBSD 9 and 10 Releases

The NetBSD 10.2_STABLE release provides many significant changes, including support for many new devices, hundreds of bug fixes, new and updated kernel subsystems, and numerous userland enhancements. The result of these improvements is a stable operating system fit for production use that rivals most commercially available systems.

One important new feature in this release is the support for extended attributes and access control lists on FFS file systems.

For new installations the installer will by default disable these features, so the file system is compatible with older NetBSD releases (before 10), and allow other operating systems to mount this file systems at least in read-only mode.

If you want a new installed file system to support extended attributes, change the file system type from "FFSv2" to "FFSv2ea" in the partitioning menu. You can also convert file systems later, using the `fsck_ffs`(8) utility. More details are available in *this guide:*

https://wiki.netbsd.org/tutorials/acls_and_extended_attributes_on_ffs.

If you are upgrading from a version of NetBSD -current please also check the **Compatibility Issues With Previous NetBSD Releases**. It is impossible to completely summarize the massive development that went into the NetBSD 10.2_STABLE release. The complete list of changes can be found in
CHANGES:

https://nycdn.NetBSD.org/pub/NetBSD-daily/netbsd-10/latest/CHANGES
CHANGES-10.1:

https://nycdn.NetBSD.org/pub/NetBSD-daily/netbsd-10/latest/CHANGES-10.1

CHANGES-10.2:
https://nycdn.NetBSD.org/pub/NetBSD-daily/netbsd-10/latest/CHANGES-10.2

CHANGES-10.3:
https://nycdn.NetBSD.org/pub/NetBSD-daily/netbsd-10/latest/CHANGES-10.3

which are also present in the top level directory of the NetBSD 10.2 release tree.

Features to be removed in a later release

The following features are to be removed from NetBSD in the future:

- `groff`(1). Man pages are now handled with `mandoc`(1), and `groff`(1) can still be found in pkgsrc as `textproc/groff`.
- `pf`(4). This packet filter is obsolete and unmaintained in NetBSD. It will be eventually removed due to possible long-standing security issues and lack of multiprocessor support. New installations should use `npf`(7).

The NetBSD Foundation

The NetBSD Foundation is a tax exempt, not-for-profit 501(c)(3) corporation that devotes itself to the traditional goals and spirit of the NetBSD Project and owns the trademark of the word "NetBSD". It supports the design, development, and adoption of NetBSD worldwide. More information on the NetBSD Foundation, its composition, aims, and work can be found at: **https://www.NetBSD.org/foundation/**

Sources of NetBSD

Refer to *mirrors:* **https://www.NetBSD.org/mirrors/**

NetBSD 10.2_STABLE Release Contents

The root directory of the NetBSD 10.2_STABLE release is organized as follows:

`.../NetBSD-10.2_STABLE/`

`CHANGES`	Changes between the 9.0 and 10.0 releases.
`CHANGES-10.0`	Changes between the initial 10.0 branch and final release of 10.0.
`CHANGES-10.1`	Changes between the 10.0 and the 10.1 release.
`CHANGES-10.2`	Changes between the 10.1 and the 10.2 release.
`CHANGES-10.3`	Changes after the release of 10.2.
`CHANGES.prev`	Changes in previous NetBSD releases.
`README.files`	README describing the distribution's contents.
`images/`	Images (ISO 9660 or USB) for installing NetBSD. Depending on your system, these may be bootable.
`source/`	Source distribution sets; see below.

In addition to the files and directories listed above, there is one directory per architecture, for each of the architectures for which NetBSD 10.2_STABLE has a binary distribution.

The source distribution sets can be found in subdirectories of the `source` subdirectory of the distribution tree. They contain the complete sources to the system. The source distribution sets are as follows:

gnusrc This set contains the "gnu" sources, including the source for the compiler, assembler, groff, and the other GNU utilities in the binary distribution sets.

sharesrc This set contains the "share" sources, which include the sources for the man pages not associated with any particular program; the sources for the typesettable document set; the dictionaries; and more.

src This set contains all of the base NetBSD 10.2_STABLE sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.

syssrc This set contains the sources to the NetBSD 10.2_STABLE kernel for all architectures as well as the `config`(1) utility.

xsrc This set contains the sources to the X Window System.

All the above source sets are located in the `source/sets` subdirectory of the distribution tree.

The source sets are distributed as compressed tar files. Except for the **pkgsrc** set, which is traditionally unpacked into `/usr/pkgsrc`, all sets may be unpacked into `/usr/src` with the command:

```
# cd / ; tar -zxpf set_name.tgz
```

In each of the source distribution set directories, there are files which contain the checksums of the files in the directory:

`MD5` MD5 digests in the format produced by the command:
cksum -a MD5 *file*.

`SHA512` SHA512 digests in the format produced by the command:
cksum -a SHA512 *file*.

The SHA512 digest is safer, but MD5 checksums are provided so that a wider range of operating systems can check the integrity of the release files.

NetBSD/mac68k subdirectory structure

The mac68k-specific portion of the NetBSD 10.2_STABLE release is found in the `mac68k` subdirectory of the distribution: `.../NetBSD-10.2_STABLE/mac68k/`. It contains the following files and directories:

`INSTALL.html`
`INSTALL.ps`
`INSTALL.txt`
`INSTALL.more` Installation notes in various file formats, including this file. The `.more` file contains underlined text using the `more`(1) conventions for indicating italic and bold display.

`binary/`
- `kernel/`
  - `netbsd-GENERIC.gz`
    A kernel containing code for everything supported in this release using the standard SCSI driver.
  - `netbsd-GENERICSBC.gz`
    A kernel containing code for everything supported in this release using the SBC variant of the SCSI driver.
- `sets/` mac68k binary distribution sets; see below.

`installation/`
- `instkernel/`
  mac68k boot and installation kernels; see installation section (Sysinst Method), below.
- `misc/` Miscellaneous mac68k installation utilities; see the Traditional method installation section below.

Binary distribution sets

The NetBSD mac68k binary distribution sets contain the binaries which comprise the NetBSD 10.2_STABLE release for mac68k. The binary distribution sets can be found in the `mac68k/binary/sets` subdirectory of the NetBSD 10.2_STABLE distribution tree, and are as follows:

base The NetBSD 10.2_STABLE mac68k **base** binary distribution. You *must* install this distribution set. It contains the base NetBSD utilities that are necessary for the system to run and be minimally functional.

comp Things needed for compiling programs. This set includes the system include files (`/usr/include`) and the various system libraries (except the shared libraries, which are included as part of the **base** set). This set also includes the manual pages for all of the utilities it contains, as well as the system call and library manual pages.

debug This distribution set contains debug information for all base system utilities. It is useful when reporting issues with binaries or during development. This set is huge, if the target disk is small, do not install it.

etc This distribution set contains the system configuration files that reside in `/etc` and in several other places. This set *must* be installed if you are installing the system from scratch, but should *not* be used if you are upgrading.

games This set includes the games and their manual pages.

kern-GENERIC
This set contains a NetBSD/mac68k 10.2_STABLE `GENERIC` kernel, named `/netbsd`. You *must* install either this distribution set or kern-GENERICSBC.

kern-GENERICSBC
This set contains a NetBSD/mac68k 10.2_STABLE `GENERICSBC` kernel, named `/netbsd`. You *must* install either this distribution set or kern-GENERIC.

man This set includes all of the manual pages for the binaries and other software contained in the **base** set. Note that it does not include any of the manual pages that are included in the other sets.

misc This set includes the system dictionaries, the typesettable document set, and other files from `/usr/share`.

modules This set includes kernel modules to add functionality to a running system.

rescue This set includes the statically linked emergency recovery binaries installed in `/rescue`.

text This set includes NetBSD's text processing tools, including `groff`(1), all related programs, and their manual pages.

NetBSD maintains its own set of sources for the X Window System in order to assure tight integration and compatibility. These sources are based on XFree86 4.5.0. Binary sets for the X Window System are distributed with NetBSD. The sets are:

xbase The basic files needed for a complete X client environment. This does not include the X servers.

xcomp The extra libraries and include files needed to compile X source code.

xdebug This distribution set contains debug information for all X11 binaries. It is useful when reporting issues with these binaries or during development. This set is huge, if the target disk is small, do not install it.

xfont Fonts needed by the X server and by X clients.

xetc Configuration files for X which could be locally modified.

xserver The X server. This includes the Xmac68k *monochrome* server with man pages.

The mac68k binary distribution sets are distributed as gzipped tar files named with the extension **.tgz**, e.g. `base.tgz`.

The instructions given for extracting the source sets work equally well for the binary sets, but it is worth noting that if you use that method, the filenames stored in the sets are relative and therefore the files are extracted *below the current directory*. Therefore, if you want to extract the binaries into your system, i.e. replace the system binaries with them, you have to run the **`tar -xzpf`** command from the root directory (/) of your system.

Kernels suitable for booting from an AppleShare server may be found in the `mac68k/binary/kernels` subdirectory of the NetBSD 10.2_STABLE distribution tree. These kernels are generally named something like `netbsd-GENERIC.gz` and can be booted as-is by the NetBSD/mac68k Booter utility, if desired. Please note that these kernels are simply gzipped and are not in tar archives.

The Mac OS based utilities necessary for installing and running NetBSD can be found in the `mac68k/installation/misc` subdirectory of the NetBSD 10.2_STABLE distribution tree. The important files in this directory are as follows:

`Booter.sea.hqx` The NetBSD/mac68k Booter utility. This program is used to boot the NetBSD kernel from within Mac OS.
141 KB archived

`Installer.sea.hqx` The NetBSD/mac68k Installer utility. This program is used to install the distribution sets onto your NetBSD partition(s). This utility is used only in a Traditional method installation; it is not used or required for an installation using the **`sysinst`** method.
147 KB archived

`Mkfs.sea.hqx` The Mkfs utility. This program is used to format your chosen partitions so that they can be used with NetBSD. This utility is used only in a Traditional method installation; it is not used or required for an installation using the **`sysinst`** method.
76 KB archived

These files are all BinHexed, self-extracting archives. If you need them, the sources for these utilities are in the `src` subdirectory.

Note: Each directory in the mac68k binary distribution also has its own checksum files, just as the source distribution does.

NetBSD/mac68k System Requirements and Supported Devices

NetBSD/mac68k 10.2_STABLE runs on several of the older Macintosh computers. About 4 MB of RAM is sufficient to boot a stripped-down custom kernel, and a subset of the system can be squeezed onto a 40 MB hard disk with considerable creativity and persistence. However, 140 MB of disk should be considered a practical minimum, and to do anything more interesting than booting at least 8 MB of RAM and more disk space is recommended.

Please note that to install NetBSD/mac68k 10.2_STABLE using the **`sysinst`** method, your system must have a minimum of 6 MB of RAM and 60 MB of available disk space (i.e. not part of an in-use HFS partition).

Supported models

- Mac II, IIx, IIcx, SE/30, IIci, IIsi, IIvx, IIvi
- Performa 400/405/410/430, Performa 450, Performa 460/466/467
- Performa 520, Performa 550/560, Performa 600/600CD
- LC II, LC III, LC III+, LC 520, LC 550
- MacTV
- Classic II, Color Classic
- Centris 650
- Quadra 610, Quadra 630, Quadra 650, Quadra 700, Quadra 800
- Quadra/Centris 660AV, Quadra 840AV
- PowerBook 140, PowerBook 145/145B, PowerBook 170
- PowerBook 160, PowerBook 165, PowerBook 180
- PowerBook 165c, PowerBook 180c, PowerBook 550c

Supported devices

- Onboard SCSI bus and most SCSI tapes, hard drives, and CD-ROMs
- Internal sound — enough to beep on some machines, anyway
- Most basic NuBus video cards (there have been some problems with some 24-bit color cards and with most QuickDraw accelerators)
- Both internal serial ports
- ADB keyboards and mice (both Apple and a number of third party multi-button mice and track-balls are supported)
- Ethernet cards based on the National Semiconductor 8390 and the SONIC (DP83932) chips (Asante, Apple, and a few others — problems still with Ethernet and many NuBus video cards)
- Ethernet cards based on the SMC 91c92 and 91c100 (FEAST) chips. This includes the Asante-FAST 10/100 cards
- Onboard Ethernet based on the SONIC chip for Quadra-series Macs
- Onboard Ethernet based on the MACE (Am79C940) chip for the Quadra AV-series Macs
- Ethernet port on Asante NetDock and Newer Ether MicroDock, for PowerBook Duo series
- Comm-slot Ethernet should be working for most machines/cards

If your 68030 system is not listed above, it may be because of a problem with accessing onboard video, and it may still work with a serial console. Some of the known ones in this category:

- Mac Classic series
- PowerBook Duo series

If your 68LC040 system is not listed above, it is due to a problem with floating point emulation (FPE) for this type of processor. Machines in this category include:

- Newer LC-series machines (47x, 57x)
- Newer Performa-series machines (47x, 57x, 58x, 63x, 640)
- Some PowerBook 500-series Macs

Unsupported models

- Macintosh IIfx

  This machine has unusual custom chips for the ADB and serial interfaces which make support for it difficult. Work is in progress on this, though.

- Quadra 900/950

  These machines have I/O processor chips for their ADB interfaces similar to those used in the IIfx and thus face similar support problems. Note that you can use a serial console on these systems.

- PowerPC-based Macs

  This is a separate effort from the mac68k port. PowerMacs use hardware that is quite different from that of the mac68k port. See the NetBSD/macppc port webpage at **http://www.NetBSD.org/ports/macppc/** for more information.

Known hardware issues with this release

- Real Time Clock

  Due to oddities of the Macintosh hardware interrupt priority scheme, NetBSD/mac68k keeps very poor time. Under a high interrupt load (e.g. SCSI or serial port activity), a machine can lose several minutes per hour. A consequence of this problem is that attempting to run **ntpd** is generally rather pointless.

- SCSI difficulties

  The NetBSD/mac68k SCSI drivers are not quite as robust as their Mac OS counterparts. Symptoms of these problems are that some SCSI disks will not work under NetBSD that work fine under Mac OS. Other problems include occasional file system corruption with some types of drives and the general unreliability of removable SCSI media. Keep in mind that there are no clear patterns with these problems, and they do not appear to affect the majority of users.

Getting the NetBSD System on to Useful Media

There are currently two installation methods available for initial installation of NetBSD on Apple Macintosh 68000-based systems. Neither supports all installation media types at this time, so the one you select must be compatible with the media you have available on your system.

- The **sysinst** method of installation uses an Installation Kernel which is a minimal NetBSD system with a memory resident set of utilities that are capable of partitioning the disk, initializing the file systems, and loading them from the archive files. Since the installation kernel does not currently support access to Mac OS HFS file systems this method requires that the Binary Distribution Sets be accessible from CD-ROM, remote NFS partition, or via FTP access.
- The Traditional method of installation uses Mac OS hosted utilities to partition your disk, initialize the partitions for use by NetBSD, and load the file systems from archive files stored on the Mac OS HFS file system. This method requires that the Binary Distribution Sets reside on a local Macintosh hard drive, a CD-ROM, or an AppleShare volume.

The Traditional method of installation is currently supported from the local Macintosh hard drive, from a CD-ROM, or from an AppleShare volume (however, you may upgrade a system from within NetBSD; see the **Upgrading a previously-installed NetBSD System** section for more details). If you are installing from a local hard drive, this means that you'll need at least enough room for the largest file that you will have to install.

If the install is being done from an AppleShare-mounted volume, the NetBSD/mac68k Installer must be in the same folder as the binary distribution sets.

Each distribution file is in *raw* archive format.

- Distribution files must be downloaded in binary mode. Common web browsers may not be suitable for this task; FTP clients such as Fetch and Anarchie work fine, but be sure to specify a binary file transfer.
- The files should not be unpacked. If you have the Internet Config extension installed, you can disable this in the "Helpers" dialog by removing the entry associated with ".tgz" files. Other FTP clients may require separate changes; consult your package's documentation.

- If you are installing from a CD-ROM, the distribution sets are already in the proper format. No special handling is required.

You will also need to collect the Mac OS installation tools from the `mac68k/installation/misc` subdirectory of the NetBSD 10.2_STABLE distribution: Mkfs, NetBSD/mac68k Installer, and NetBSD/mac68k Booter. These three are in BinHexed, self-extracting archives as Mkfs.sea.hqx, Installer.sea.hqx, and Booter.sea.hqx, respectively. Extract them as you would any other Macintosh application.

Note: The Booter is the only Mac OS application needed if the **sysinst** method of installation is used. The Traditional method of installation is deprecated and will be removed in a future release, and the Mkfs and Installer tools will be retired.

Preparing your System for NetBSD installation

No matter which installation method you use, there is some planning and preparation that is required beforehand. First and foremost, before beginning the installation process, make sure you have a reliable backup of any data on your hard disk that you wish to keep. *Mistakes in partitioning your hard disk may lead to data loss.*

NetBSD/mac68k uses the same disk mapping scheme as Mac OS: the Apple Disk Partition Map. This permits both systems to reside on the same disk, but introduces some installation problems unique to the Macintosh. There are very few, if any, reliable ways to reduce the size of an existing Mac OS disk partition, so partitioning a disk that currently contains Mac OS will almost always require a backup and reload step under Mac OS.

If you are using the **sysinst** method of installation you will be able to do most, if not all, of your disk partitioning during the install process. Partitioning the disk with **sysinst** will destroy any partition that is resized, deleted, converted, or designated for use by NetBSD. All space not planned to be used for Mac OS HFS partitions may be used by NetBSD and can be sub-divided by the **sysinst** process. This space may be defined within one or more existing disk partitions of any type, including HFS partitions that are no longer needed for Mac OS. However it is best if this space is physically contiguous on the disk as **sysinst** is not capable of merging non-contiguous disk partitions. If you are using the **sysinst** method and have sufficient disk space in one or more disk partitions you should skip forward to the section labeled **Installing the NetBSD System (Sysinst Method)** in this document.

If you are using the Traditional method of installation you must use a disk partitioning utility to designate the different partitions you will want in your final NetBSD configuration. It is not necessary to create NetBSD (or AU/X) type partitions at this stage; the Mkfs utility can convert a partition of any type to one usable for NetBSD.

If disk partitioning is required because you've selected the Traditional method of installation, or because disk space needs to be freed up for use for the **sysinst** method of installation, follow the directions in the remainder of this section.

Find your favorite disk partitioning utility. Any formatter capable of partitioning a SCSI disk should work. Some of the ones that have been tried and seem to work are:

- Apple HD SC Setup
- Hard Disk ToolKit from FWB
- SCSI Director Lite
- Disk Manager Mac from OnTrack
- Silverlining from LaCie
- APS Disk Tools

Apple's HD SC Setup is probably the easiest to use and the most commonly available. Instructions for patching HD SC Setup so that it will recognize non-Apple drives is available at:

http://www.euronet.nl/users/ernstoud/patch.html

First, you need to choose a drive on which to install NetBSD. Try to pick a drive with a low SCSI target number (or "SCSI ID"), especially if you are likely to add or remove drives to your SCSI chain in the future.

Note: Be certain you have a reliable backup of any data that you may want to keep. Repartitioning your hard drive is an excellent way to destroy important data.

Second, decide how you want to set up your partitions. At minimum, you need a partition to hold the NetBSD installation (the root partition — `/`) and a partition to serve as swap space. You may choose to use more than one partition to hold the installation. This allows you to separate the more vital portions of the file system (such as the kernel and the `/etc` directory) from the more volatile parts of the file system. Typical setups place the `/usr` directory on a separate partition from the root partition (`/`). Generally, `/` can be fairly small while the `/usr` partition should be fairly large. If you plan to use this machine as a server, you may also want a separate `/var` partition.

Once you have decided how to lay out your partitions, you need to calculate how much space to allocate to each partition. A minimal install of NetBSD (i.e. `base.tgz`, `etc.tgz`, and either `kern-GENERIC.tgz` or `kern-GENERICSBC.tgz`) requires about 140MB. A general rule of thumb for sizing the swap partition is to allocate twice as much swap space as you have real memory. Having your swap + real memory total at least 20 MB is also a good idea. Systems that will be heavily used or that are low on real memory should have more swap space allocated. Systems that will be only lightly used or have a very large amount of real memory can get away with less.

Keep in mind that NetBSD currently requires Mac OS in order to boot, so it is likely that you will want to keep at least a minimal install of Mac OS around on an HFS partition for this purpose. The size of this partition may vary depending on the size requirements for the version of Mac OS you are using. Of course, if you have Mac OS on another hard drive or can boot from a floppy, feel free to dedicate the entire drive to NetBSD.

Next, use your favorite partitioning utility to make partitions of the necessary sizes. You can use any type of partition, but partitions of type `Apple_Free` might save you some confusion in the future.

You are now set to install NetBSD on your hard drive.

Installing the NetBSD System (Sysinst Method)

Running the sysinst installation program

1. *Introduction*

   Using **sysinst**, installing NetBSD is a relatively easy process. Still, you should read this document and have it available during the installation process. This document tries to be a good guide to the installation, and as such, covers many details for the sake of completeness. Do not let this discourage you; the install program is not hard to use.

2. *Possible hardware-specific issues*

   - *SCSI driver problems*

     The SCSI driver used in the kernel on many older Macintosh systems is, by default, the ncrscsi driver. It contains a recognized but as yet unfixed bug that affects some disk drive/controller combinations, usually Quantum disks. Under heavy load these systems may hang or corrupt the file system; or, you may experience frequent *Segmentation fault* and *Illegal instruction* errors that may or may not be consistently repeatable. This latter condition is particularly prevalent on systems with minimal RAM installed.

     If either of these problems occur on your system you are advised to use the SBC variants of the Kernel and Installation Kernel. However, be aware that this issue does not affect e.g. Centris or Quadra systems.

- *The 68LC040 processor*

  NetBSD has known but unresolved problems running on the 68LC040 processor, the variant of the 68040 that does not contain the floating point unit (FPU). The kernel is thus forced to emulate the missing operations in software. Unfortunately the 68LC040 processor has a design problem that causes the emulation to fail intermittently. We hope to provide a solution for this issue in a future NetBSD release.

  Software emulation of floating point operations is not a problem on the 68020 and 68030 processors.

3. *General*

   The following is a walk-through of the steps you will take while installing NetBSD on your hard disk. **sysinst** is a menu-driven program that guides you through the installation process. Sometimes questions will be asked, and in many cases the default answer will be displayed in brackets ("[ ]") after the question. If you wish to stop the installation, you may press `CONTROL-C` at any time, but if you do, you'll have to begin the installation process again from scratch by running the `/sysinst` program from the command prompt. It is not necessary to reboot.

4. *Quick install*

   First, let's describe a quick install. The other sections of this document go into the installation procedure in more detail, but you may find that you do not need this. If you want detailed instructions, skip to the next section. This section describes a basic installation, using a CD / DVD as the install medium.

- What you need.
  - The distribution sets (in this example, they are on the CD or DVD).
  - The Mac OS Booter application and an Installation Kernel
  - A Macintosh with a 68020 and MMU, 68030 or 68RC040 processor. An FPU is not required but will be used if present (but see the note above regarding the 68LC040).
  - A minimum of 8 MB of memory installed.
  - An optical drive.
  - A hard drive with at least 500 MB of free space for a complete base install, not including room for swap. If you wish to install the X Window System as well, you will need at least 225 MB more.
- The NetBSD Boot Tools folder.
  - Create a Folder on your Mac OS disk for the NetBSD/mac68k components.
  - Copy the Booter application into the newly created Folder. Expand the file if necessary to create the Mac OS executable.
  - Copy the Installation Kernels into the newly created Folder. It is not necessary to **gunzip** compressed kernel files.
  - Single-click on the Booter application icon then select the "Get Info" from the File Menu list. Increase the memory allocation for the Booter to as much as possible for your system. Having a large number of fonts, extensions or sounds installed on your system can cause memory exhaustion problems for the Booter if you don't do this. Also, the extra memory is needed by the Booter to expand compressed kernels while booting.

- The Quick Installation
  - Double-click on the Booter application icon to start executing it. From the **Options** pull-down menu, select **Monitors**, then select **Change Monitor Depth** and make sure B&W is highlighted. Close the window using the **Close** button.
  - From the **Options** pull-down menu select **Boot Options**. This will bring up an option panel. Set the **Auto-set GMT Bias** checkbox in the lower left and then select the **Boot from Mac OS** option at the top of the window. The **Set** button on the right will become active. Use it to locate and select the Installation Kernel file appropriate for your hardware. This will be either `netbsd-INSTALL.gz` or `netbsd-INSTALLSBC.gz`. Close the window using the **Close** button.
  - From the **Options** pull-down menu select the **Boot Now**, or use the Apple-B (Command-B) key combination to start the NetBSD boot process. Do not move the mouse while the boot operation is in progress as this may leave the keyboard locked to NetBSD.

    The main menu will be displayed. Insert the first boot floppy you just created and boot the computer. After language selection, the main menu will be displayed.

```
.***********************************************.
* NetBSD-10.2_STABLE Install System             *
*                                               *
*>a: Install NetBSD to hard disk                *
* b: Upgrade NetBSD on a hard disk              *
* c: Re-install sets or install additional sets *
* d: Reboot the computer                        *
* e: Utility menu                               *
* f: Config menu                                *
* x: Exit Install System                        *
.***********************************************.
```

  - If you wish, you can configure some network settings immediately by choosing the **Utility menu** and then **Configure network**. It isn't actually required at this point, but it may be more convenient. Go back to the main menu.
  - Choose **Install**.
  - You will be guided through the setup of your disk.
  - You will be asked to choose which distribution sets to install.
  - When prompted, choose **CD-ROM** as the install medium if booted from CD-ROM. The default values for the path and device should be OK.
  - After the installation process has completed, you will be brought back to the main menu, where you should select **Reboot.**
  - NetBSD will now boot. If you didn't set a password for the `root` user when prompted by **sysinst**, logging in as `root` and setting a password should be your first task. You are also advised to read `afterboot`(8).

5. *Booting NetBSD*

   Prior to attempting to boot NetBSD/mac68k verify that all the following are done:

   - Enable 32-bit addressing in the Memory Control Panel [1].

- Disable all forms of virtual memory (the Memory Control Panel, RAM Doubler, or other software-based memory enhancement products).
- Place the system in B&W Mode (1-bit color or grayscale) as shown in the Monitors Control Panel or in the Monitors options dialog of the Booter. You may choose to have the Booter do this for you automatically by selecting the appropriate check box and radio button in the **Monitors** dialog on the **Options** menu.

It is probably best to boot your machine with all extensions turned off [1]. You can do this by booting into Mac OS with the SHIFT key held down. You may have to restart your Macintosh for changes to take effect before proceeding.

[1] If your version of the Memory control panel does not have a 32-bit addressing mode radio button, this means that your system is already 32-bit clean and is running in 32-bit addressing mode by default. If the Booter complains that your are not in 32-bit mode, it may be necessary for you to press the "Use Defaults" button in the Memory control panel to restore 32-bit addressing. You should probably reboot after doing so. If you have an older II-class system (including the II, IIx, IIcx, and SE/30), it is necessary to install Connectix's MODE32 to work around ROM issues which prevent you from enabling 32-bit addressing. Please see the NetBSD/mac68k *FAQ:*
https://www.NetBSD.org/ports/mac68k/faq/
for more information.

Double-click on the NetBSD/mac68k Booter icon to start the application. Select **Booting** from the **Options** menu. Select the Kernel Location to be from Mac OS with the filename corresponding to the name of the Installation Kernel you are using. Typically this will be netbsd-INSTALL.gz.

If you haven't already put your Macintosh into B&W mode, select the **Monitor Options** from the **Options** menu and check the box for B&W mode.

Try booting NetBSD by selecting **Boot Now** from the **Options** menu.

If the system does not come up, send mail to `port-mac68k@NetBSD.org` describing your software, your hardware, and as complete a description of the problem as you can. As an alternative, try using the Traditional method of installation described in the next section.

6. *Preparing your hard disk*

You are now at the point of no return. Nothing has been written to your disk yet, but if you confirm that you want to install NetBSD, your hard drive will be modified. If you are sure you want to proceed, select **yes**.

The install program will now label your disk and create the file systems you specified. The file systems will be initialized to contain NetBSD bootstrapping binaries and configuration files. You will see messages on your screen from the various NetBSD disk preparation tools that are running. There should be no errors in this section of the installation. If there are, restart from the beginning of the installation process. Otherwise, you can continue the installation program after pressing the return key.

7. *Getting the distribution sets*

The NetBSD distribution consists of a number of *sets* that come in the form of gzipped tar files. At this point, you will be presented with a menu which enables you to choose from one of the following methods of installing the sets. Some of these methods will first transfer the sets to your hard disk, others will extract the sets directly.

For all these methods, the first step is to make the sets available for extraction. The sets can be made available in a few different ways. The following sections describe each of the methods. After reading about the method you will be using, you can continue to the section labeled 'Extracting the distribution sets'.

8. *Installation from CD-ROM*

   When installing from a CD-ROM, you will be asked to specify the device name for your CD-ROM drive (usually `cd0`) and the directory name on the CD-ROM where the distribution files are.

   sysinst will then check that the files are actually present in the specified location and proceed to the extraction of the sets.

9. *Installation using FTP*

   To install using FTP, you first need to configure your network setup if you haven't already done so. **sysinst** will help you with this, asking if you want to use DHCP. If you do not use DHCP, you can enter network configuration details yourself. If you do not have DNS set up for the machine that you are installing on, you can just press RETURN in answer to this question, and DNS will not be used.

   You will also be asked to specify the host that you want to transfer the sets from, the directory on that host, the account name and password used to log into that host using FTP, and optionally a proxy server to use. If you did not set up DNS, you will need to specify an IP address instead of a hostname for the FTP server.

   sysinst will then transfer the set files from the remote site to your hard disk.

10. *Installation using NFS*

    To install using NFS, you first need to configure your network setup if you haven't already done so. **sysinst** will do this for you, asking you if you want to use DHCP. If you do not use DHCP, you can enter network configuration details yourself. If you do not have DNS set up for the machine that you are installing on, you can just press RETURN in answer to this question, and DNS will not be used.

    You will also be asked to specify the host that you want to transfer the sets from and the directory on that host that the files are in. This directory should be mountable by the machine you are installing on, i.e., correctly exported to your machine.

    If you did not set up DNS, you will need to specify an IP address instead of a hostname for the NFS server.

11. *Installation from Mac OS file systems*

    NetBSD/mac68k does not currently have in-kernel support for Mac OS HFS/HFS+ or AppleShare filesystems. **sysinst** therefore can not access the file sets if they are on these filesystems.

12. *Installation from an unmounted file system*

    In order to install from a local file system, you will need to specify the device that the file system resides on (for example `wd1e`), the type of the file system, and the directory on the specified file system where the sets are located. **sysinst** will then check if it can indeed access the sets at that location.

13. *Installation from a local directory*

    This option assumes that you have already done some preparation yourself. The sets should be located in a directory on a file system that is already accessible. **sysinst** will ask you for the name of this directory.

14. *Extracting the distribution sets*

    A progress bar will be displayed while the distribution sets are being extracted.

    After all the files have been extracted, the device node files will be created. If you have already configured networking, you will be asked if you want to use this configuration for normal operation. If so, these values will be installed in the network configuration files.

15. *Configure additional items*

    The next menu will allow you to select a number of additional items to configure, including the time zone that you're in, to make sure your clock has the right offset from UTC, the root user's shell, and the initial root password.

    You can also enable installation of binary packages, which installs the `pkgin`(1) tool for managing binary packages for third-party software. This will feel familiar to users of package tools such as **apt-get** or **yum**. If you prefer to install third-party software from source, you can install the `pkgsrc`(7) tree.

    Finally, you can enable some daemons such as `sshd`(8), `ntpd`(8), or `mdnsd`(8).

16. *Finalizing your installation*

    Congratulations, you have successfully installed NetBSD 10.2_STABLE. You can now reboot the machine and boot NetBSD from hard disk.

Installing the NetBSD System (Traditional Method)

The Traditional method of installation can be broken down into three basic steps:

- Run Mkfs to build a file system or file systems.
- Run the Installer to load the files onto your file systems.
- Run the Booter to boot the system.

Preparing the file system(s)

Double-click on the Mkfs application icon to start it up. It will ask you for the SCSI-ID (SCSI target number) of the drive that you are installing NetBSD on. Once this is selected, it will present a list of the partitions on that disk. You must first convert the partitions to a type which NetBSD can understand. Select each partition on which you wish to build a file system and click on the **Change** button. If you are placing the entire installation on a single partition, select the **NetBSD Root&Usr** radio button. If you are using multiple partitions, select **NetBSD Root** for the root partition (/) and **NetBSD Usr** for all the other partitions. You should select **NetBSD Swap** for the swap partition.

When you have finished converting each partition, select each partition and click on the **Format** button. You will now be asked for a bunch of parameters for the hard drive and the file system. Usually, you can just take the defaults. If you are installing onto removable media (e.g. a Zip, Jaz, or Syquest), please see the FAQ. Note that although this dialog only has the **OK** button, you are not committed, yet. Once you get the values you want, press the **OK** button. A dialog will be presented at this point with two options: **Format** and **Cancel**. If you choose **Cancel**, nothing will be written to your drive. If you choose **Format**, the program will proceed to make a file system.

Mkfs is not a well-behaved Macintosh application. It will not allow any other tasks to run while it does (cooperative multitasking at its best). When it's finished, the program will put up a dialog to ask if you have scanned the output for any error messages. Usually there won't have been any errors, but do scan the output to make sure. Simply click on the **I Read It** button and the program will quit.

Repeat as necessary for any extra partitions that you wish to make file systems on. Note that you do *not* need a file system on your swap partition.

When you are finished, click on the **Done** button and choose **Quit** from the **File** menu to exit Mkfs.

Installing the files

Before using the Installer, it is probably a good idea to increase its memory allocation. Select the Installer icon by clicking on it and choose **Get Info** from the File menu. Increase both the Minimum and Preferred sizes to as much as you can spare.

Double-click on the Installer icon to start it up. The Installer will present the same SCSI-ID menu that Mkfs did. Select the same SCSI-ID (SCSI target number) that you did for Mkfs - i.e., the one you are installing NetBSD on.

If you are installing onto a single root partition (/), proceed to the **Installation of base files** section, below.

If you have not created file systems for / (root), usr, and any other file systems, go back to **Preparing the file system(s)** above.

When you started the Installer, it mounted your root partition (/). Just before it printed

```
Mounting partition 'A' as /
```

it printed lines like:

```
sd1 at scsi ID 5
```

This means that the device for SCSI target 5 ("SCSI ID 5") is sd1. The partitions are signified by a trailing letter. For instance, sd1a would be the root partition (/) of the second SCSI disk in the chain, and sd0g would be the first Usr partition on the first SCSI disk.

You will need to know the proper device to mount the remaining partition(s) by hand:

1. Select **Build Devices** from the **File** menu.
2. Select **Mini Shell** from the **File** menu.
3. You can use the **disklabel** command to get a listing of the available partitions and their types and sizes.
4. Create the directory mount point(s) with the command:

   # **mkdir path**

   E.g. for the /usr partition type:

   # **mkdir /usr**
5. Mount the file systems you wish with the command:

   # **mount device path**
   For example, if you wish to mount a /usr partition from the first SCSI disk sd0, on /usr, you would type:

   # **mount /dev/sd0g /usr**
6. Type
   # **fstab force**
   to create a proper /etc/fstab file.
7. Type **quit** after you have mounted all the file systems.

Installation of base files

Select the **Install** menu item from the **File** menu and install `base.tgz`, `etc.tgz`, `either kern-GENERIC.tgz or kern-GENERICSBC.tgz`, and any other sets you wish to install at this time (see the **NetBSD 10.2_STABLE Release Contents** for information about what's in each set). The Installer will print out the filename of each file as it is installed, and will take quite some time to install everything (the base package alone can take over two hours on a slow hard drive).

As is the case with Mkfs, this is not a particularly well-behaved Macintosh application and the machine will be completely tied up while the installation takes place.

At some point after installing the base set, select the **Build Devices** option from the **File** menu if you have not already done so. This will create a bunch of device nodes for you and will create your initial `/etc/fstab`. The Installer program also has an option to give you a mini-shell. Do not use this unless you are sure know what you are doing.

When you are finished installing all of the sets you wish to install, exit the Installer by choosing **Quit** from the **File** menu.

Booting the system

Prior to attempting to boot NetBSD/mac68k, please verify that all of the following are true:

1. 32-bit addressing is enabled [2] in the Memory control panel;
2. All forms of virtual memory are disabled (the Memory control panel, RAM Doubler, or other software-based memory enhancement products); and
3. Your system is in B&W mode (1-bit color or grayscale) as shown by the Monitors control panel. You may choose to have the Booter do this for you automatically by selecting the appropriate check box and radio button in the **Monitors** dialog on the **Options** menu.

It is probably best to boot your machine with all extensions turned off [2]. You can do this by booting into Mac OS with the SHIFT key held down. You may have to restart your Macintosh for changes to take effect before proceeding.

[2] If your version of the Memory control panel does not have a 32-bit addressing mode radio button, this means that your system is already 32-bit clean and is running in 32-bit addressing mode by default. If the Booter complains that you are not in 32-bit mode, it may be necessary for you to press the **Use Defaults** button in the Memory control panel to restore 32-bit addressing. You should probably reboot after doing so. If you have an older II-class system (including the II, IIx, IIcx, and SE/30), it is necessary to install Connectix's MODE32 to work around ROM issues which prevent you from enabling 32-bit addressing. Please see the NetBSD/mac68k FAQ at

http://www.NetBSD.org/ports/mac68k/faq/

for more information.

Double-click on the NetBSD/mac68k Booter icon to start the application. Select **Booting** from the **Options** menu. Check that all of the items in the resulting dialog look sane - especially the SCSI target number. If not, correct them to your preference (the SCSI target number, or "SCSI ID", should be the only thing you need to change). When you are satisfied with your choices, try booting NetBSD by selecting **Boot Now** from the **Options** menu.

If you wish to save your preferences, choose **Save Options** from the **File** menu before Booting (your preferences will not be saved if you forget to do this).

If the system does not come up, send mail to `port-mac68k@NetBSD.org` describing your software, your hardware, and as complete a description of the problem as you can.

If the system does come up, congratulations, you have successfully installed NetBSD 10.2_STABLE.

Post installation steps

Once you've got the operating system running, there are a few things you need to do in order to bring the system into a properly configured state. The most important steps are described below.

1. Before all else, read `postinstall`(8).

2. Configuring `/etc/rc.conf`

   If you or the installation software haven't done any configuration of `/etc/rc.conf` (**sysinst** normally will), the system will drop you into single user mode on first reboot with the message

   ```
   /etc/rc.conf is not configured. Multiuser boot aborted.
   ```

   and with the root file system (`/`) mounted read-only. When the system asks you to choose a shell, simply press RETURN to get to a `/bin/sh` prompt. If you are asked for a terminal type, respond with **vt220** (or whatever is appropriate for your terminal type) and press RETURN. You may need to type one of the following commands to get your delete key to work properly, depending on your keyboard:

   ```
   # stty erase '^h'
   # stty erase '^?'
   ```

   At this point, you need to configure at least one file in the `/etc` directory. You will need to mount your root file system read/write with:

   ```
   # /sbin/mount -u -w /
   ```

   Change to the `/etc` directory and take a look at the `/etc/rc.conf` file. Modify it to your tastes, making sure that you set `rc_configured=YES` so that your changes will be enabled and a multi-user boot can proceed. Default values for the various programs can be found in `/etc/defaults/rc.conf`, where some in-line documentation may be found. More complete documentation can be found in `rc.conf`(5).

   When you have finished editing `/etc/rc.conf`, type **exit** at the prompt to leave the single-user shell and continue with the multi-user boot.

   Other values that may need to be set in `/etc/rc.conf` for a networked environment are *hostname* and possibly *defaultroute*. You may also need to add an *ifconfig_int* for your ⟨int⟩ network interface, where your on-board, NuBus or PDS interface may be `ae0`, `mc0` or `sn0`. For example:

   ```
   ifconfig_sn0="inet 192.0.2.123 netmask 255.255.255.0"
   ```

   or, if you have *myname.my.dom* in `/etc/hosts`:

   ```
   ifconfig_sn0="inet myname.my.dom netmask 255.255.255.0"
   ```

   To enable proper hostname resolution, you will also want to add an `/etc/resolv.conf` file or (if you are feeling a little more adventurous) run `named`(8). See `resolv.conf`(5) or `named`(8) for more information.

   Instead of manually configuring networking, DHCP can be used by setting `dhcpcd=YES` in `/etc/rc.conf`.

3. Logging in

   After reboot, you can log in as `root` at the login prompt. If you didn't set a password in **sysinst**, there is no initial password. You should create an account for yourself (see below) and protect it and the "root" account with good passwords. By default, root login from the network is disabled (even via `ssh`(1)). One way to become root over the network is to log in as a different user that belongs to group "wheel" (see `group`(5)) and use `su`(1) to become root.

4. Adding accounts

   Use the `useradd`(8) command to add accounts to your system. Do not edit `/etc/passwd` directly! See `vipw`(8) and `pwd_mkdb`(8) if you want to edit the password database.

5. The X Window System

   If you installed the X Window System, you may want to read the chapter about X in the *NetBSD Guide:* **https://www.NetBSD.org/docs/guide/en/chap-x.html**:

   [Color_X] Some systems are capable of supporting a color X Server in either thousands or millions of colors. The mac68k port does not support resolution switching at this time. If your system is capable of running the color X Server it can be installed at this point and the Booter options can be modified to match the resolution depth.

6. Installing third-party packages

   If you wish to install any of the software freely available for UNIX-like systems you are strongly advised to first check the NetBSD package system, pkgsrc. pkgsrc automatically handles any changes necessary to make the software run on NetBSD. This includes the retrieval and installation of any other packages the software may depend upon.

   – More information on the package system is available at
     https://www.NetBSD.org/docs/software/packages.html

   – A list of available packages suitable for browsing is at
     https://cdn.NetBSD.org/pub/pkgsrc/current/pkgsrc/README.html

   – Precompiled binaries can be found at
     https://cdn.NetBSD.org/pub/pkgsrc/packages/NetBSD/
     usually in the `mac68k/10.2_STABLE/All` subdir. If you installed `pkgin`(1) in the **sysinst** post-installation configuration menu, you can use it to automatically install binary packages over the network. Assuming that `/usr/pkg/etc/pkgin/repositories.conf` is correctly configured, you can install them with the following commands:

     ```
     # pkgin install tcsh bash perl apache xfce4 firefox
     ...
     ```

     The above command will install the Tenex-csh and Bourne Again shells, the Perl programming language, Apache web server, Xfce desktop environment and the Firefox web browser as well as all the packages they depend on.

     If it was not automatically installed, `pkgin`(1) can be installed on a fresh NetBSD system with `pkg_add`(1):

     ```
     export PKG_PATH=https://cdn.netbsd.org/pub/pkgsrc/packages/NetBSD/$(uname -p)/$
     pkg_add pkgin
     ```

   – If you wish to use the `pkgsrc`(7) framework for compiling packages and did not install it from the `sysinst`(8) post-installation configuration menu, you can obtain it by retrieving the file
     https://cdn.NetBSD.org/pub/pkgsrc/stable/pkgsrc.tar.gz.
     It is typically extracted into `/usr/pkgsrc` (though other locations work fine) with the commands:

     ```
     # cd /usr
     # tar -zxpf pkgsrc.tar.gz
     ```

     After extracting, see the `doc/pkgsrc.txt` file in the extraction directory (e.g., `/usr/pkgsrc/doc/pkgsrc.txt`) for more information.

7. Misc

   – Edit `/etc/mail/aliases` to forward root mail to the right place. Don't forget to run `newaliases`(1) afterwards.

   – Edit `/etc/rc.local` to run any local daemons you use.

   – Many of the `/etc` files are documented in section 5 of the manual; so just invoking

     ```
     # man 5 filename
     ```

     is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

The easiest way to upgrade to NetBSD 10.2_STABLE is with binaries, and that is the method documented here.

To do the upgrade, you must boot the install kernel using one of the methods described above. You must also have at least the **base** and **kern** binary distribution sets available. Finally, you must have sufficient disk space available to install the new binaries. Since files already installed on the system are overwritten in place, you only need additional free space for files which weren't previously installed or to account for growth of the sets between releases.

Since upgrading involves replacing the kernel, boot blocks, and most of the system binaries, it has the potential to cause data loss. You are strongly advised to *back up* any important data on the NetBSD partition or on another operating system's partition on your disk before beginning the upgrade process.

The upgrade procedure is similar to an installation, but without the hard disk partitioning.

Fetching the binary sets is done in the same manner as the installation procedure; refer to the installation part of the document for help. File systems are checked before unpacking the sets.

After a new kernel has been copied to your hard disk, your machine is a complete NetBSD 10.2_STABLE system. However, that doesn't mean that you're finished with the upgrade process. You will probably want to update the set of device nodes you have in `/dev`. If you've changed the contents of `/dev` by hand, you will need to be careful about this, but if not, you can just cd into `/dev`, and run the command:

```
# sh MAKEDEV all
```

`sysinst` will attempt to merge the settings stored in your `/etc` directory with the new version of NetBSD using the `postinstall`(8) utility. However, `postinstall`(8) is only able to deal with changes that are easily automated. It is recommended that you use the `etcupdate`(8) tool to merge any remaining configuration changes.

Compatibility Issues With Previous NetBSD Releases

Users upgrading from previous versions of NetBSD may wish to bear the following problems and compatibility issues in mind when upgrading to NetBSD 10.2_STABLE.

Note that **`sysinst`** will automatically invoke

```
postinstall fix
```

and thus all issues that are fixed by **`postinstall`** by default will be handled.

In NetBSD 9 and earlier, filesystems listed in `/etc/fstab` would be mounted before non-legacy **`zfs`** filesystems. Starting from NetBSD 10 this order has been reversed.

If you have ever run a version of NetBSD -current between April 18, 2020 and September 23, 2022 (the version numbers used in the affected time range are between 9.99.56 and 9.99.106) your FFS file systems might have broken extended attributes stored.

You should follow this *guide:*

https://wiki.netbsd.org/features/UFS2ea/

before booting the updated system multi-user for the first time.

Note that you do not need to do anything special if you never did run any affected kernel, especially if you have never run NetBSD -current.

The display drivers used for modern GPUs and the whole subsystem supporting it (DRM/KMS) have been updated to a newer version. Unfortunately not all issues with this have been resolved before the NetBSD 10.0 release. You can find a list of issues in the *Open issues with new DRM/KMS:*

https://wiki.netbsd.org/releng/netbsd-10/

section of the release engineering wiki page.

A number of things have been removed from the NetBSD 10.2_STABLE release. See the "Components removed from NetBSD" section near the beginning of this document for a list.

Using online NetBSD documentation

Documentation is available if you installed the manual distribution set. Traditionally, the "man pages" (documentation) are denoted by '`name(section)`'. Some examples of this are

- `intro`(1),
- `man`(1),
- `apropos`(1),
- `passwd`(1), and
- `passwd`(5).

The section numbers group the topics into several categories, but three are of primary interest: user commands are in section 1, file formats are in section 5, and administrative information is in section 8.

The *man* command is used to view the documentation on a topic, and is started by entering **man** [*section*] *topic*. The brackets [] around the section should not be entered, but rather indicate that the section is optional. If you don't ask for a particular section, the topic with the lowest numbered section name will be displayed. For instance, after logging in, enter

```
# man passwd
```

to read the documentation for `passwd`(1). To view the documentation for `passwd`(5), enter

```
# man 5 passwd
```

instead.

If you are unsure of what man page you are looking for, enter

```
# apropos subject-word
```

where *subject-word* is your topic of interest; a list of possibly related man pages will be displayed.

Administrivia

If you've got something to say, do so! We'd like your input. There are various mailing lists available via the mailing list server at `majordomo@NetBSD.org`. See **https://www.NetBSD.org/mailinglists/** for details.

There are various mailing lists set up to deal with comments and questions about this release. Please send comments to: `netbsd-comments@NetBSD.org`.

To report bugs, use the `send-pr`(1) command shipped with NetBSD, and fill in as much information about the problem as you can. Good bug reports include lots of details.

Bugs also can be submitted and queried with the web interface at
https://www.NetBSD.org/support/send-pr.html

There are also port-specific mailing lists, to discuss aspects of each port of NetBSD. Use majordomo to find their addresses, or visit **https://www.NetBSD.org/mailinglists/**

If you're interested in doing a serious amount of work on a specific port, you probably should contact the 'owner' of that port (listed below).

If you'd like to help with NetBSD, and have an idea as to how you could be useful, send us mail or subscribe to: `netbsd-users@NetBSD.org`.

As a favor, please avoid mailing huge documents or files to these mailing lists. Instead, put the material you would have sent up for FTP or WWW somewhere, then mail the appropriate list about it. If you'd rather not do that, mail the list saying you'll send the data to those who want it.

Thanks go to

– The former members of UCB's Computer Systems Research Group, including (but not limited to):

Keith Bostic
Ralph Campbell
Mike Karels
Marshall Kirk McKusick

for their work on BSD systems, support, and encouragement.

– The Internet Systems Consortium, Inc. for hosting the NetBSD FTP, CVS, AnonCVS, mail, mail archive, GNATS, SUP, Rsync and WWW servers.

– The Internet Research Institute in Japan for hosting the server which runs the CVSweb interface to the NetBSD source tree.

– The Columbia University Computer Science Department for hosting the build cluster.

– The many organizations that provide NetBSD mirror sites.

– Without CVS, this project would be impossible to manage, so our hats go off to Brian Berliner, Jeff Polk, and the various other people who've had a hand in making CVS a useful tool.

– We list the individuals and organizations that have made donations or loans of hardware and/or money, to support NetBSD development, and deserve credit for it at **https://www.NetBSD.org/donations/** (If you're not on that list and should be, tell us! We probably were not able to get in touch with you, to verify that you wanted to be listed.)

– Finally, we thank all of the people who've put sweat and tears into developing NetBSD since its inception in January, 1993. (Obviously, there are a lot more people who deserve thanks here. If you're one of them, and would like to be mentioned, tell us!)

Legal Mumbo-Jumbo

All product names mentioned herein are trademarks or registered trademarks of their respective owners.

The following notices are required to satisfy the license terms of the software that we have mentioned in this document:

NetBSD is a registered trademark of The NetBSD Foundation, Inc.
This product includes software developed by the University of California, Berkeley and its contributors.
This product includes software developed by the NetBSD Foundation.
This product includes software developed by The NetBSD Foundation, Inc. and its contributors.

This product includes software developed for the NetBSD Project. See https://www.NetBSD.org/ for information about NetBSD.
This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)
This product includes cryptographic software written by Eric Young (eay@mincom.oz.au)
This product includes software designed by William Allen Simpson.
This product includes software developed at Ludd, University of Luleå.
This product includes software developed at Ludd, University of Luleå, Sweden and its contributors.
This product includes software developed at the Information Technology Division, US Naval Research Laboratory.
This product includes software developed by Aaron Brown and Harvard University.
This product includes software developed by Adam Ciarcinski for the NetBSD project.
This product includes software developed by Adam Glass.
This product includes software developed by Adam Glass and Charles M. Hannum.
This product includes software developed by Alex Zepeda.
This product includes software developed by Alex Zepeda, and Colin Wood for the NetBSD Project.
This product includes software developed by Allen Briggs.
This product includes software developed by Amancio Hasty and Roger Hardiman
This product includes software developed by Ben Gray.
This product includes software developed by Berkeley Software Design, Inc.
This product includes software developed by Bill Paul.
This product includes software developed by Bodo Möller.
This product includes software developed by Boris Popov.
This product includes software developed by Brini.
This product includes software developed by Bruce M. Simpson.
This product includes software developed by Causality Limited.
This product includes software developed by Charles Hannum.
This product includes software developed by Charles M. Hannum.
This product includes software developed by Charles M. Hannum, by the University of Vermont and State Agricultural College and Garrett A. Wollman, by William F. Jolitz, and by the University of California, Berkeley, Lawrence Berkeley Laboratory, and its contributors.
This product includes software developed by Christian E. Hopps.
This product includes software developed by Christian E. Hopps, Ezra Story, Kari Mettinen, Markus Wild, Lutz Vieweg and Michael Teske.
This product includes software developed by Christopher G. Demetriou.
This product includes software developed by Christopher G. Demetriou for the NetBSD Project.
This product includes software developed by Chuck Silvers.
This product includes software developed by Cisco Systems, Inc.
This product includes software developed by Colin Wood.
This product includes software developed by Colin Wood for the NetBSD Project.
This product includes software developed by Computing Services at Carnegie Mellon University (http://www.cmu.edu/computing/).
This product includes software developed by Daan Vreeken.
This product includes software developed by Daishi Kato
This product includes software developed by Daniel Widenfalk and Michael L. Hitch.
This product includes software developed by Daniel Widenfalk for the NetBSD Project.
This product includes software developed by David Jones and Gordon Ross
This product includes software developed by David Miller.
This product includes software developed by Dean Huxley.
This product includes software developed by Emmanuel Dreyfus
This product includes software developed by Eric S. Hvozda.
This product includes software developed by Eric S. Raymond
This product includes software developed by Eric Young (eay@cryptsoft.com)
This product includes software developed by Eric Young (eay@mincom.oz.au)
This product includes software developed by Ezra Story.
This product includes software developed by Ezra Story and by Kari Mettinen.
This product includes software developed by Ezra Story, by Kari Mettinen and by Bernd Ernesti.
This product includes software developed by Ezra Story, by Kari Mettinen, and Michael Teske.
This product includes software developed by Ezra Story, by Kari Mettinen, Michael Teske and by Bernd Ernesti.
This product includes software developed by Frank van der Linden for the NetBSD Project.
This product includes software developed by Gardner Buchanan.
This product includes software developed by Garrett D'Amore.
This product includes software developed by Gary Thomas.
This product includes software developed by Gordon Ross
This product includes software developed by Harvard University.
This product includes software developed by Harvard University and its contributors.
This product includes software developed by Hellmuth Michaelis and Joerg Wunsch
This product includes software developed by Henrik Vestergaard Draboel.
This product includes software developed by Herb Peyerl.
This product includes software developed by Hidetoshi Shimokawa.

This product includes software developed by Hubert Feyrer for the NetBSD Project.
This product includes software developed by Ian W. Dall.
This product includes software developed by Intel Corporation and its contributors.
This product includes software developed by Internet Initiative Japan Inc.
This product includes software developed by Internet Research Institute, Inc.
This product includes software developed by James R. Maynard III.
This product includes software developed by Jared D. McNeill.
This product includes software developed by Jason L. Wright
This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com/
This product includes software developed by Joachim Koenig-Baltes.
This product includes software developed by Jochen Pohl for The NetBSD Project.
This product includes software developed by Joerg Wunsch
This product includes software developed by John Birrell.
This product includes software developed by John P. Wittkoski.
This product includes software developed by John Polstra.
This product includes software developed by Jonathan R. Stone for the NetBSD Project.
This product includes software developed by Jonathan Stone.
This product includes software developed by Jonathan Stone and Jason R. Thorpe for the NetBSD Project.
This product includes software developed by Jonathan Stone for the NetBSD Project.
This product includes software developed by Julian Highfield.
This product includes software developed by K. Kobayashi
This product includes software developed by K. Kobayashi and H. Shimokawa
This product includes software developed by Kazuhisa Shimizu.
This product includes software developed by Kazuki Sakamoto.
This product includes software developed by Kenneth Stailey.
This product includes software developed by Kiyoshi Ikehara.
This product includes software developed by Klaus Burkert,by Bernd Ernesti, by Michael van Elst, and by the University of California, Berkeley and its contributors.
This product includes software developed by Kyma Systems.
This product includes software developed by Leo Weppelman and Waldi Ravens.
This product includes software developed by Lloyd Parkes.
This product includes software developed by Lutz Vieweg.
This product includes software developed by Marc Horowitz.
This product includes software developed by Marcus Comstedt.
This product includes software developed by Mark Brinicombe.
This product includes software developed by Mark Brinicombe for the NetBSD Project.
This product includes software developed by Mark Tinguely and Jim Lowe
This product includes software developed by Markus Wild.
This product includes software developed by Marshall M. Midden.
This product includes software developed by Masanobu Saitoh.
This product includes software developed by Masaru Oki.
This product includes software developed by Matt DeBergalis
This product includes software developed by Matthew Fredette.
This product includes software developed by Michael Smith.
This product includes software developed by Microsoft
This product includes software developed by Mika Kortelainen
This product includes software developed by Mike Pritchard.
This product includes software developed by Mike Pritchard and contributors.
This product includes software developed by Minoura Makoto.
This product includes software developed by MINOURA Makoto, Takuya Harakawa.
This product includes software developed by Niels Provos.
This product includes software developed by Niklas Hallqvist.
This product includes software developed by Niklas Hallqvist, Brandon Creighton and Job de Haas.
This product includes software developed by Paolo Abeni.
This product includes software developed by Paul Kranenburg.
This product includes software developed by Paul Mackerras.
This product includes software developed by Paul Mackerras <paulus@samba.org>.
This product includes software developed by Pedro Roque Marques <pedro_m@yahoo.com>
This product includes software developed by Per Fogelstrom.
This product includes software developed by Peter Galbavy.
This product includes software developed by Phase One, Inc.
This product includes software developed by Philip A. Nelson.
This product includes software developed by QUALCOMM Incorporated.
This product includes software developed by Ravikanth.
This product includes software developed by RiscBSD.
This product includes software developed by Roar Thronæs.

In the following statement, the phrase ‘‘this text’’ refers to portions of the system documentation.

The End

Contributions

The following people have made contributions of various sorts specifically for the Macintosh port (in alphabetical order):

- All of the users who have supplied us with good bug reports and moral support.
- The Alice Group (Allen K. Briggs, Chris P. Caputo, Michael L. Finch, Bradley A. Grantham, and Lawrence A. Kesteloot), without whom there would be no NetBSD port for the Macintosh.

– Steven R. Allen for keeping our snapshot distributions up-to-date.
– Stephen C. Brown for maintaining the Installer application.
– Denton Gentry and Yanagisawa Takeshi for their work on the SONIC Ethernet driver.
– Paul Goyette, Taras Ivanenko, Ken Nakata, and Michael R. Zucca for invaluable work towards supporting color X.
– Takashi Hamada and John Wittkoski beating the direct ADB hardware driver into submission.
– David Huang for getting MACE Ethernet and basic DMA working on the AV Macs.
– Scott Jann for acquiring a IIx and a IIci, used for building and testing release sets.
– Scott Kaplan for lending his IIci and Kensington Turbo Mouse for IIci/IIsi banked memory and internal video as well as non-Apple ADB devices.
– Noah M. Kieserman for lending a PowerBook 520C for tracking down several bugs on that platform.
– Markus Krummenacker for monetary donations.
– Glan Lalonde for an invaluable IIci page table dump.
– Dan McMahill for lending a PowerBook 165 to tweak ADB support on the PowerBook 160 and 180 family laptops.
– Bob Nestor for (unofficially) maintaining the Mkfs utility, and providing a lot of useful information about the ROM vectors used by different systems.
– Brad Parker for serial and Ethernet drivers/improvements.
– Brian R. Gaeke and Nigel Pearson for tweaking, polishing, and performing the occasional major refit on the Booter application.
– Scott Redman for lending Brad Grantham a PowerBook 160.
– Craig Ruff for assembling an '030 `pmove ttx` instruction.
– Brad Salai for lending an Ethernet card to help resolve interrupt conflicts.
– Larry Samuels for monetary donations.
– Peter Siebold for lending his IIvx in support of ADB and IIvx internal video.
– Glen Stewart for lending a Carrera040 accelerator which, while still unsupported, helped to track down memory management bugs for '030-based machines.
– Bill Studenmund for providing a stable front end to the machine- independent serial driver.
– Schuyler Stultz for the loan of his Macintosh II when we desperately needed another machine on which to compile and test during the '93 Xmas vacation.
– Tenon Intersystems for monetary donations, MachTen, and Brad's access to several machines and documentation after hours.
– Virginia Tech English Department for loan of a IIci w/ NuBus video and 32 MB of RAM — the first IIci to run NetBSD/mac68k.
– Rob Windsor for donating a variety of Macintosh II-family systems, a Centris 650, a Quadra 700, and several boxes full of miscellaneous peripherals and parts in the interest of ensuring adequate testing and working out minor (and not-so-minor) problems.

– Colin Wood for maintaining a host of NetBSD/mac68k documentation, including the FAQ, Meta-FAQ, and OS Info documents.