**Documentation**

**Notation**
In the Linux world, you will at times see a notation not only referring to a name, but also to a manpage category. In the BSD world, it is usual to do exactly that by appending the category number in parentheses after the name. Thus, when people would usually say *Just use sh!*, you’d *translate* it to *Just use sh(1)!* when they want you to use the shell.

In the same way, as packages in the BSD world are organized in categories, you will mostly see references to the category prepended to the package’s name, e.g., you’ll be told to install *www/firefox* instead of just *firefox*, where *www* is the pkgsrc-category of Firefox (see below).

This notation helps omitting ambiguous descriptions, and you usually can determine whether you are talking about a kernel driver, a userland tool, configuration file or a library by just looking at the category the object is meant to be in.

**The Guide**
As other BSDs, NetBSD has a central handbook, called *The NetBSD Guide*. This guide can be seen as the reference for the system, containing a description of all important subsystems and introductions on how to use them (e.g., hard disk encryption), and should be your first start if you have a problem.

**The articles**
The NetBSD website contains a huge amount of NetBSD-related documentation including articles, guides, FAQs, and how-tos. Of course, there are many other websites dedicated to NetBSD as well; the NetBSD website often provides links to them. For less static information, also have a look at the NetBSD wiki.

**Community/bug reports**
When the documentation doesn’t suffice for solving an issue you encounter, you may ask about it in the community; if it turns out to be a bug, you can report it as such. There are several methods to do so, the fastest ones being described here:

**Mailing lists**
Mostly all of NetBSD’s communication is done via mailing lists. You can subscribe to them (most of them don’t have too much traffic, and nearly no dumb traffic anyway), or read all the archives on the web or Usenet.

**Announcements**
For general information like release announcements, security advisories, etc., there is the *netbsd-announce* list. There is very little traffic on this list, as only important announcements are made through it. Security announcements are also announced on the *security-announce* list; you are strongly advised to subscribe to this list if you use NetBSD anywhere.

**Bug reports**
If you encounter a problem, you can file a problem report (PR). First take a look at the PR database to see if the issue hasn’t already been reported (it may even include a temporary workaround); if it wasn’t, you can do so using a form on the NetBSD website, or by using the *send-pr(1)* command.

**IRC**
Decentally organized, there are NetBSD channels in the larger IRC networks like freenode, EFnet, and IRCnet, generally named *#netbsd*.

**Tips and Tricks for NetBSD Newcomers**
Nowadays there are only few people who start using Unix by using a shell. Most people install their Unix-like operating system, with GNU/Linux being the most widespread one, with a graphical installer.

This introductory text is made for people who already use GNU/Linux and know about its internals and the principles of the system. It provides a few tips and tricks on how to switch to NetBSD, and what you have to consider when switching.

Note: in the following text, *Linux* actually means GNU/Linux, except where just the kernel is meant.
Generic tips

- The configuration in NetBSD is mostly the same as in Linux, or any other Unix-like system for that matter. Most configuration files of the base system are in /etc; those of additional third party software from pkgsrc are in /usr/pkg/etc. Most programs allow users to override these files with equivalent ones in their home directories, mostly having an (almost) equal name.

Default configuration files for system processes and procedures are often kept in /etc/defaults. These files are not just templates, but actually processed; configuration directives defined in them may be overridden though by files with the same name in the parent directory (i.e., /etc), then containing the same directives. See the rc.conf(5) manpage for an example hereof.

- sysctl(7) is an interface to set various parameters of the kernel, mostly equivalent to Linux’ procfs subsystem. Though Linux also has a sysctl command doing the same work, it is there mostly another interface to the procfs subsystem.

sysctl(7) in NetBSD is generally tidier, and contains descriptions of the options and parameters. With the sysctl(8) command, you can show and set these values, and show their descriptions.

An exact description of some important system information variables can be found in the sysctl(7) manpage; having a look here might even help experienced users fine-tune their kernel parameters.

- A useful equivalent to Linux’ lspci is pcictl(8), which you can use for manipulating a PCI bus, e.g., list all devices connected to it.

To see information the kernel printed out while booting (but which is now lost due to system message buffer being filled by other, later messages), you can have a look at the /var/run/dmesg.boot file. Here, all the information from the boot process, for example your processor model, devices detected on your system, etc. is listed. The system message buffer itself can always be viewed with the dmesg(8) command. When you report a hardware problem, it is almost always helpful to include the contents of the system message buffer.

- The NetBSD Guide has several chapters about building your system from source, which is an easy task to do, with NetBSD. You can build NetBSD on nearly any Unix-like operating system and any platform, for any platform; cross-compiling them without hassle is possible, so you don’t have to use your weak old machine for building its own current system.

NetBSD names most of its devices after its manufacturer, being the classical Unix way. You will notice this when, for instance, having a look at the network interfaces: instead of being called eth0, eth1 etc. (as is the case in Linux), they are named after their driver, e.g., wmi0, wm1 etc. for wm(4) Intel Gigabit Ethernet network interfaces. As another example, wd0, wd1, etc. are the device names for wd(4) Western Digital WD100x compatible hard disks.

NetBSD provides ISO-files which can be written to a CD-ROM, to install the system from there. As an extra, the sysutils/mkmemstick package can be used to convert these ISO-files to images to be written to a USB memory stick, so such device can be used as a bootable installation medium as well. Writing such new image to a memory stick can easily be done using the dd(1) command.

Equivalently to Linux’ logrotate NetBSD provides the newsyslog(8) command, which is activated by default, holding its configuration in the /etc/newsyslog.conf file.

NetBSD can be configured to download a list of known security vulnerabilities each day, which will then be compared to your installed packages. You are highly advised to do so, as security vulnerabilities in third party programs lower the overall safety of your system. You are advised to update the respective package as soon as possible.

- For various ways in which people contribute to NetBSD, see http://www.NetBSD.org/contrib

Of course, you are encouraged to contribute yourself as well!