

# Easy Software Installation with pkgsrc

*One 3rd Party Software collection  
for \*BSD, Solaris, Linux ... and more!*

Prepared by Hubert Feyrer <hubertf@NetBSD.org>

Presented by D'Arcy J. M. Cain <darcy@NetBSD.org>

The NetBSD Foundation



Installation of Open Source software on Unix and Unix-like systems has a number of problems:

- Many programs and lots of version changes
- Compilation costs time
- Software often is not written with portability in mind (but we don't want to give a coding lesson here...)
- Installation is not trivial



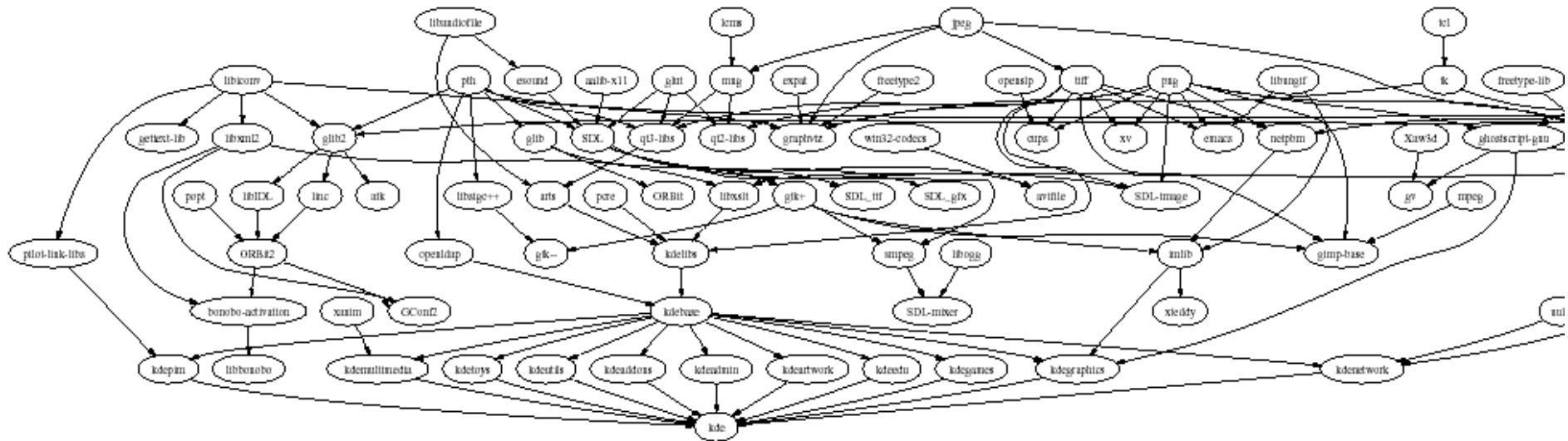
# Problems (cont'd)

- Installation is not trivial:
  - ◆ Some basic knowledge about tools is necessary
  - ◆ Various ways to configure things (GNU autoconf, Imake, ...)
  - ◆ Side effects (depending on other packages, compiler, ...)
  - ◆ Many inter-dependent packages
  - ◆ Troubleshooting requires expert knowledge



# Problems (cont'd)

## Illustration of complexity of inter-dependending packages:

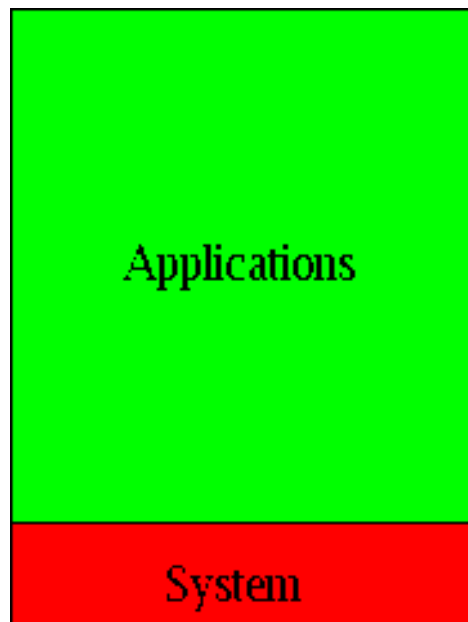


(created from a pkgsrc system running NetBSD, using pkgdepgraph and dot/graphviz)



# Solution: It depends! (1/2)

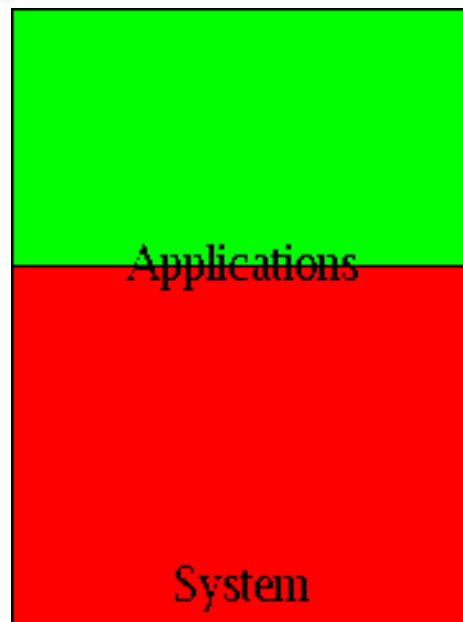
Classic, flexible  
software management:



- difficult to install  
+ easy to maintain

E.g. Solaris, Irix,  
Linux From Scratch

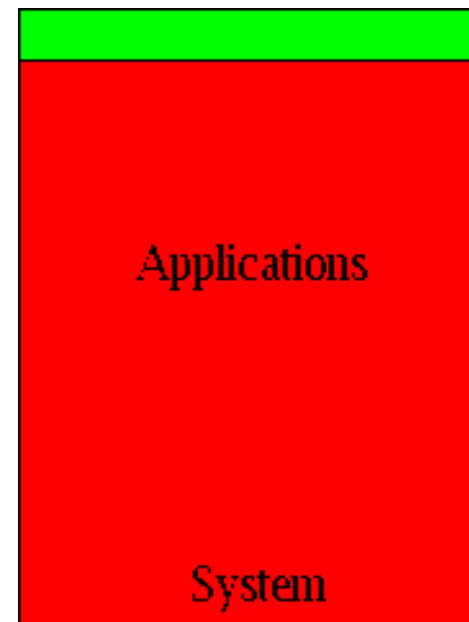
Hybrid systems:



+ easy to install  
+ easy to maintain

E.g. NetBSD, FreeBSD,  
Debian & Gentoo Linux

Complete integration  
of applications and system:



+ easy to install  
- difficult to maintain

E.g. SuSE, RedHat,  
Mandrake Linux



# Solution: It depends! (2/2)

## Where do you want to go today?

- **Easy Installation:** choose this if your software doesn't change often. Use ready-to-user binary distribution. E.g. for desktop systems install Windows or SuSE Linux from CD/DVD.
- **Easy Maintenance:** choose this if you have few packages that change a lot. Take a stable base operating system, and install important packages on your own, e.g. compile on your own on a webserver with Solaris, Apache and PHP.
- **Both:** Welcome to pkgsrc!



# A Cross-Platform Solution: pkgsrc



# Introducing pkgsrc

- System for easy installation and updating of packages
- Source-based package management system
- Uses original source code for compiling
- Creation and installation of binary packages is possible
- Components: Management tools & packages collection (pkgsrc)
- Automatic handling of dependencies (of course!?!)





# Introducing pkgsrc (cont'd)

- Originally ported from FreeBSD to NetBSD
- Primary development platform of pkgsrc: NetBSD
- Ported to: AIX, BSD/OS, Darwin, FreeBSD, Irix, Linux, OpenBSD, Solaris, Windows w/ “Interix”
- Linux Distributions: SuSE 9.0, Debian, ROOT Linux, Slackware, RedHat 8.1/9, Mandrake 9.2, Bluewall, ...



# pkgsrc in Detail



# How to get going

- `Grab pkgsrc`
- `Install bootstrap kit (binary, or compile via pkgsrc/bootstrap)`
- `cd pkgsrc/www/mozilla`
- `bmake install`



# Grabbing pkgsrc

```
$ cd $HOME/OS
$ env CVS_RSH=ssh \
  cvs -d \
    anoncvs@anoncvs.NetBSD.org: /cvsroot \
    co pkgsrc
U pkgsrc/Makefile
U pkgsrc/Packages.txt
U pkgsrc/README
...
```

Alternative: [ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar\\_files/pkgsrc.tar.gz](ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/pkgsrc.tar.gz)



# Bootstrap Kit: Binaries

- Grab a precompiled binary or compile on your own
- Precompiled binary kits are available on <http://www.pkgsrc.org/> for:

Darwin 5.5/powerpc

FreeBSD 5.1/i386

OpenBSD 3.5/i386

Darwin 6.6/powerpc

FreeBSD 5.2.1/i386

Slackware 8.1/i386

Darwin 7.6/powerpc

FreeBSD 5.3/i386

Slackware 9/i386

Darwin 7.8/powerpc

Interix 3.5

Solaris 8/sparc

Debian GNU Linux/i386

IRIX 6.5/mips

Solaris 8/i386

FreeBSD 3.5/i386

IRIX64 6.5/mips

Solaris 9/sparc

FreeBSD 4.7/i386

OpenBSD 3.2/i386

Solaris 9/i386

FreeBSD 5.0/i386

OpenBSD 3.3/i386



# Bootstrap Kit: Compiling (1/2)

```
$ cd pkgsrc/bootstrap
$ export MY_HOME=$HOME/OS/OS-`uname -s`
$ export LOCALBASE=${MY_HOME}/pkg
$ export PKG_DBDIR=${MY_HOME}/db/pkg
$ ./bootstrap \
?         --prefix=${LOCALBASE} \
?         --pkgdbdir=${PKG_DBDIR} \
?         --ignore-user-check
==> bootstrap command: ./bootstrap --prefix=/home/feyrer/OS/OS-Linux/pkg --pkgdbdi
==> bootstrap started: Wed Dec  8 14:42:23 CET 2004
Working directory is: work
==> running: /usr/bin/sed -e 's|@DEFAULT_INSTALL_MODE@|'0755'|' files/install-sh.i
==> running: /bin/chmod +x work/install-sh
==> building as unprivileged user feyrer/bedienst
==> Building libnbcompat
==> running: /bin/sh work/install-sh -d -o feyrer -g bedienst work/libnbcompat
==> running: (cd work/libnbcompat; /bin/sh ./configure -C --prefix=/home/feyrer/OS
configure: creating cache config.cache
checking build system type... i686-pc-linux-gnu
checking host system type... i686-pc-linux-gnu
checking whether make sets $(MAKE)... yes
.....
```



# Bootstrap Kit: Compiling (2/2)

```
.....  
/usr/bin/install -c -m 444 linkfarm.cat1 /home3/bedienst/feyrer/OS/OS-Linux/pkg/man  
/usr/bin/install -c -m 444 pkg_view.1 /home3/bedienst/feyrer/OS/OS-Linux/pkg/man/ma  
/usr/bin/install -c -m 444 pkg_view.cat1 /home/feyrer/OS/OS-Linux/pkg/man/cat1/pkg_  
==> Installing packages(7) man page  
==> running: /bin/sh work/install-sh -c -m 444 files/packages.cat7 /home/feyrer/OS
```

Please remember to add /home/feyrer/OS/OS-Linux/pkg/bin to your PATH environment variable and /home/feyrer/OS/OS-Linux/pkg/man to your MANPATH environment variable, if necessary.

An example mk.conf file "work/mk.conf.example" with the settings you provided to "bootstrap" has been created for you.

Please copy work/mk.conf.example to /home/feyrer/OS/OS-Linux/pkg/etc/mk.conf.

You can find extensive documentation of the NetBSD Packages Collection in /home/feyrer/OS/pkgsrc/Packages.txt and packages(7).

Hopefully everything is now complete.

Thank you

```
==> bootstrap started: Wed Dec  8 14:44:09 CET 2004
```

```
==> bootstrap ended:   Wed Dec  8 14:55:52 CET 2004
```

```
$
```



# Bootstrap Kit: Adjust \$PATH etc.

```
$ cd $HOME/OS/OS-`uname -s`/pkg
$ export PATH=`pwd`/bin:`pwd`/sbin:${PATH}
$ export PKG_DBDIR=$HOME/OS/OS-`uname -s`/db/pkg
$
$ pkg_info
digest-20021220      Message digest wrapper utility
```





# Installed Commands

The binaries installed by the bootstrap procedure provide the core functionality of the pkgsrc system:

```
$ cd OS/OS-`uname -s`/pkg/
```

```
$ ls bin sbin
```

```
bin:
```

```
bmake          cpio           digest         ftp
pax            tar
```

```
sbin:
```

```
linkfarm      pkg_add        pkg_create     pkg_info
mtree         pkg_admin     pkg_delete     pkg_view
```



# Compiling Packages - Overview

**Beware!** Make sure that instead of “make” the BSD-compatible “bmake” is being used!

```
$ export MAKECONF=`pwd`/pkgsrc_env_no-root # see below
$
$ cd $HOME/OS/pkgsrc
$ cd misc/figlet
$ bmake
...
$ bmake install
...
$
$ pkg_info
digest-20021220      Message digest wrapper utility
figlet-2.2.1nb2     Print text banners in fancy ASCII art characters
```



# Compiling Packages - Details (1/2)

```
$ bmake
```

```
==> *** No /home/feyrer/OS/OS-Linux/./distfiles/pkg-vulnerabilit  
==> *** skipping vulnerability checks. To fix, install  
==> *** the pkgsrc/security/audit-packages package and run  
==> *** '/home/feyrer/OS/OS-Linux/pkg/sbin/download-vulnerability  
=> Checksum OK for figlet221.tar.gz.
```

```
work.i386 -> /home/feyrer/OS/OS-Linux/tmp/misc/figlet/work.i386
```

```
==> Extracting for figlet-2.2.1nb2
```

```
==> Patching for figlet-2.2.1nb2
```

```
==> Applying pkgsrc patches for figlet-2.2.1nb2
```

```
==> Overriding tools for figlet-2.2.1nb2
```

```
==> Configuring for figlet-2.2.1nb2
```

```
==> Building for figlet-2.2.1nb2
```

```
gcc -O2 -DDEFAULTFONTDIR=\"/home/feyrer/OS/OS-Linux/pkg/share/figl
```

```
chmod a+x figlet
```

```
gcc -O2 -o chkfont chkfont.c
```

```
$
```



# Compiling Packages - Details (2/2)

```
$ bmake install
==> Installing for figlet-2.2.1nb2
==> Becoming root@rfhinf032 to install figlet.
Warning: not superuser, can't run mtree.
Become root and try again to ensure correct permissions.
install -d -o feyrer -g bedienst -m 755 /home/feyrer/OS/OS-Linux/pkg
mkdir -p /home/feyrer/OS/OS-Linux/pkg/share/figlet
cp figlet /home/feyrer/OS/OS-Linux/pkg/bin
cp chkfont /home/feyrer/OS/OS-Linux/pkg/bin
chmod 555 figlist showfigfonts
cp figlist /home/feyrer/OS/OS-Linux/pkg/bin
cp showfigfonts /home/feyrer/OS/OS-Linux/pkg/bin
cp fonts/*.flf /home/feyrer/OS/OS-Linux/pkg/share/figlet
cp fonts/*.flc /home/feyrer/OS/OS-Linux/pkg/share/figlet
cp figlet.6 /home/feyrer/OS/OS-Linux/pkg/man/man6
==> Registering installation for figlet-2.2.1nb2
$
```



# Compiling Packages - Running

```
$ type figlet
/home/feyrer/OS/OS-Linux/pkg/bin/figlet
$
$ figlet Hello `uname -s`
```

```
 _ _ _ _ _          _ _          _ _          _ _          _ _
|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|
|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|
|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|
|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|
|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|
```



# Compiling as Non-root

To use pkgsrc without root privileges, put the following into \$MAKECONF (shortened!):

```
MY_NAME!=          whoami
MY_GROUP!=         groups | sed 's/ .*$$//'
MY_HOME=           ${HOME}/OS
BINOWN=            ${MY_NAME}
BINGRP=            ${MY_GROUP}
WRKOBJDIR=         ${MY_HOME}/tmp
LOCALBASE=         ${MY_HOME}/pkg
VARBASE=           ${MY_HOME}/var
OBJMACHINE=        1
SU_CMD=            /bin/sh -c
CHOWN=             true
CHGRP=             true
BINMODE=           755                                # for Solaris strip(1)
```

Complete: [http://www.feyrer.de/OS/pkgsrc\\_env\\_no-root!](http://www.feyrer.de/OS/pkgsrc_env_no-root!)



## 1. `make fetch`: Download sources



# Other Interesting Targets

- `make package`: Create binary package for `pkg_add(8)`
- `make clean`: Remove working directory
- `make deinstall`: Deinstall package
- `make replace`: Replace installed package with new version
- `make update`: Rebuild package and all dependencies





# What packages are there: Categories

```
$ cd ../pkgsrc/  
$ ls  
CVS                comms              games              misc               security  
Makefile           converters         geography          mk                 shells  
Packages.txt       cross             graphics          multimedia         sysutils  
README            databases         ham                net                 templates  
archivers          devel             inputmethod       news                textproc  
audio              distfiles         lang               packages            time  
benchmarks         doc               licenses           parallel            wm  
biology            editors           mail               pkglocate           www  
bootstrap          emulators        math               pkgtools            x11  
cad                finance           mbone              print  
chat               fonts             meta-pkgs          regress
```



# Example: WWW Category

```
$ cd ../pkgsrc
$ ls www
CVS                cadaver            jakarta-servletap p5-Apache-Test
Makefile           calamaris          jakarta-tomcat     p5-Apache-ePerl
Mosaic             cgic               jakarta-tomcat4   p5-CGI
SpeedyCGI          cgicc              jsdk20             p5-CGI-Applicatio
adzap              cgilib            jssi               p5-CGI-FastTempla
amaya              checkbot           kannel             p5-CGI-FormBuilde
analog             chimera            kdwebdev3          p5-CGI-Kwiki
ap-Embperl         clearsilver        kimagemapeditor   p5-CGI-Minimal
ap-access-referer cocoon             lhs                p5-CGI-Session
ap-aolserver       communicator       libghttp           p5-CGI_Lite
ap-auth-cookie     cronolog           libgtkhtml         p5-ExtUtils-XSBui
ap-auth-ldap       curl              libwww             p5-FCGI
ap-auth-mysql      cvsweb            liferea            p5-HTML-Clean
ap-auth-pam        dillo             links              p5-HTML-FillInFor
ap-auth-pgsql      drivell           links-gui          p5-HTML-FixEntiti
ap-auth-postgresq elinks            lynx               p5-HTML-Format
ap-auth-script     elinks04          mMosaic            p5-HTML-Mason
ap-bandwidth       emacs-w3m         make_album         p5-HTML-Parser
...
```



# Number of Available Packages

```
$ date
Sat May  7 06:08:52 EDT 2005
$
$ cd .../pkgsrc/
$ ls */*/Makefile | wc -l
    5526                                <- pkgsrc
$ ls wip/*/Makefile | wc -l
    1053                                <- SourceForge's pkgsrc-wip
$ expr 5526 + 1053
6579                                    <- total
```



# Internals



# Makefile: Construction Manual

```
$ cat x11/xteddy/Makefile
# $NetBSD: Makefile,v 1.12 2005/04/11 21:48:16 tv Exp $

DISTNAME=      xteddy-1.1
CATEGORIES=    x11 games
MASTER_SITES= http://www.ITN.LiU.SE/~stegu/xteddy/

MAINTAINER=    johnam@mail.kemper.org
HOMEPAGE=     http://www.ITN.LiU.SE/~stegu/xteddy
COMMENT=      Xteddy is a cuddly teddy bear for your X Windows desktop

USE_X11=      YES
GNU_CONFIGURE= YES

pre-install:
    ${INSTALL_DATA_DIR} ${PREFIX}/share/xteddy
    ${INSTALL_DATA_DIR} ${PREFIX}/share/xteddy/pixmaps

.include "../../graphics/xpm/buildlink3.mk"

.include "../../mk/bsd.pkg.mk"
```



# Dependencies

## Various ways:

- Compile-time only: BUILD\_DEPENDS
- Compile- and runtime: DEPENDS
- Compile- and runtime: buildlink3.mk



# Dependencies: \*DEPENDS

```
$ cd .../pkgsrc/  
$ grep ^DEPEND meta-pkgs/kde3/Makefile  
DEPENDS+=      kdeaccessibility-3.3.0nb1:../../misc/kdeaccessibil  
DEPENDS+=      kdeartwork-3.3.0nb1:../../misc/kdeartwork3  
DEPENDS+=      kdeaddons-3.3.0nb1:../../misc/kdeaddons3  
DEPENDS+=      kdeadmin-3.3.0nb1:../../misc/kdeadmin3  
...
```

The variables `BUILD_DEPENDS` and `DEPENDS` are assigned pairs of “Name-Version:Directory”. “Name-Version” is name and version of the required package, “Directory” is the path relative to this pkg’s directory where the package can be found if it’s not installed and needs to be built from source.



## The buildlink3.mk files contain variables which say ...

- which header-files to use  
(`BUILDLINK_INCDIRS.$BUILDLINK_PACKAGES`)
- which libraries to use  
(`BUILDLINK_LIBDIRS.$BUILDLINK_PACKAGES`)
- which name+version of this package should be expected  
(`BUILDLINK_DEPENDS.$BUILDLINK_PACKAGES`)
- in which pkgsrc directory to look if the package needs to be installed  
(`BUILDLINK_PKGSRCDIR.$BUILDLINK_PACKAGES`)
- if there are additional CPP flags to use  
(`BUILDLINK_CPPFLAGS.$BUILDLINK_PACKAGES`)
- if this package needs further packages installed





# Example: tiff/buildlink3.mk

```
$ cat graphics/tiff/buildlink3.mk
# $NetBSD: buildlink3.mk,v 1.8 2004/10/03 00:14:58 tv Exp $

BUILDLINK_DEPTH:=      ${BUILDLINK_DEPTH}+
TIFF_BUILDLINK3_MK:=   ${TIFF_BUILDLINK3_MK}+

.if !empty(BUILDLINK_DEPTH:M+)
BUILDLINK_DEPENDS+=    tiff
.endif

BUILDLINK_PACKAGES:=   ${BUILDLINK_PACKAGES:Ntiff}
BUILDLINK_PACKAGES+=  tiff

.if !empty(TIFF_BUILDLINK3_MK:M+)
BUILDLINK_DEPENDS.tiff+=    tiff>=3.6.1
BUILDLINK_RECOMMENDED.tiff+=    tiff>=3.6.1nb3
BUILDLINK_PKGSRCDIR.tiff?=    ../../graphics/tiff
.endif  # TIFF_BUILDLINK3_MK

.include "../../devel/zlib/buildlink3.mk"
.include "../../graphics/jpeg/buildlink3.mk"

BUILDLINK_DEPTH:=      ${BUILDLINK_DEPTH:S/+$/ /}
```



<http://www.pkgsrc.org/>

<http://www.NetBSD.org/packages/>

