Home
Recent changes
NetBSD blog
Presentations
About
Developers
Gallery
Ports
Packages
Documentation
FAQ & HOWTOs
The Guide
Manual pages
Wiki
Support
Community
Mailing lists
Bug reports
Security
Developers
CVSWeb
Mercurial
Cross-reference
Release engineering
Projects list

Announcing NetBSD 9.2 (May 12, 2021)

Introduction

The NetBSD Project is pleased to announce NetBSD 9.2 "Nakatomi Socrates", the second update of the NetBSD 9 release branch.

It represents a selected subset of fixes deemed important for security or stability reasons since the release of NetBSD 9.1 in October 2020, as well some enhancements backported from the development branch. It is fully compatible with NetBSD 9.0.

Quick download links

- USB stick installation images: 64-bit x86, 32-bit x86
- SD card live images: 64-bit ARM, ARMv7 (most 32-bit boards), ARMv6 (Raspberry Pi 1 only)
- CD installation images: 64-bit x86, 32-bit x86, 64-bit SPARC
- Other images and distribution files

The CD images support booting from an actual CD/DVD or virtual machine *only*, for everything else use the other images. In order to write USB drive and SD card images, use gunzip(1) and dd(1) on Unix, or Rawrite32 on Windows. On ARM boards (not Raspberry Pi), you may also need to write U-Boot to the SD card.

Upgrade instructions

An existing installation can be upgraded by booting an installation image and selecting the Upgrade option.

Unattended upgrades can be performed using the sysupgrade tool from pkgsrc. If you are using sysupgrade from a release earlier than 9.0, update the kernel and modules *first*, reboot and make sure the NetBSD 9.2 kernel is running, then update the rest of the system.

Changes since NetBSD 9.1

Kernel

- *netinet*: avoid information disclosure, NetBSD-SA2021-001: Predictable ID disclosures in IPv4 and IPv6
- *netinet*: fixed "multicast router sends multicast packet with invalid UDP checksum" (PR 55779)
- *xen*: fixes for XSA-362 - backends treating grant mapping errors as bugs. A malicious DomU could trigger a Dom0 kernel panic.
- *xen*: removed support for `rx-flip` mode in xennet(4) and xvif(4) as part of XSA-362 fixes (driver already defaulted to the faster `rx-copy` mode).
- *zfs*: various stability fixes. Fixed "panic when creating a directory on a NFS served ZFS". (PR 55042)
- *coda*: fixed "coda client opens wrong files instead of cache containers". (PR 55775)
- *hyperv*: fixed "unable to ifconfig(8) up/down with hvn device".
- *msdosfs*: fixed "`BOOTSIG0` and `BOOTSIG1` checks prevent mounting Raspberry Pi Pico's USB mass storage" (PR 55985)
- *kern*: fixed "panic while loading multiple large firmware files before init(8)" (PR 55906)
- *fdescfs*: fixed "fdescfs creates nodes with wrong major number" (PR 56130)
- *procfs*: corrected the permissions of the `environ` node.
- *usb*: removed incorrect assertions in abort paths, fixes false assertion failures with `DIAGNOSTIC` enabled.
- *sysctl(7)*: `kern.maxfiles`'s default value now scales with system RAM. Avoids resource exhaustion in hungry applications, e.g. multiprocess Mozilla Firefox.
- *compat_netbsd32(8)*: various improvements on AArch64:
  - Added support for ARMv6 userspace. We now build ARMv6 binary packages in a sandbox on an aarch64 server.
  - Added support for ptrace(2), fixed clone(2), fixed core file format.
  - Emulate instructions that were deprecated in ARMv7.
- *compat_linux(8):* fixed bug-compatibility with programs that use a longer `namelen` than the size of a valid `struct sockaddr_in *`.
- *threadpool(9):* fixed "`threadpool_job_cancelthrash` test randomly fails" (PR 55948)

Programs and services

- *calendar(1):* updated Judaic calendar to 2021.
- *ctwm(1):* adjusted default window manager configuration to improve accessibility, based on feedback from users. Fixed problems with window focus.
- *ftp(1):* fixed "`ftp -q` does not work". (PR 55857)
- *nl(1):* improved POSIX conformance. Allow one and two character delimiters with `-d`. (PR 55891)
- *patch(1):* fixed the behaviour of *-V none*.
- *progress(1):* handle `EINTR` in writes. (PR 55914)
- *ps(1):* fixed the calculation of widths for the `lstart` column if an empty column header is specified.
- *ksh(1):* fixed "ksh unable to execute ERR traps" (PR 56007)
- *sh(1):* fixed handling of `NUL` characters in shell scripts. (PR 55979)
- *sh(1):* fixed fallout related to PR 48875: avoid invalid subshell-elimination optimization when there are pending background jobs.
- *pkg_add(1):* moved the default package database location on *new installations* from `/var/db/pkg` to `/usr/pkg/pkgdb`, for consistency with the pkgsrc bootstrap and pkgsrc on other platforms. It can be overridden in pkg_install.conf(5).
- *vmstat(1):* stopped vmstat from exiting if it can't get the addresses of time values it often doesn't

NetBSD-Announce archive

[Date Prev][Date Next][Thread Prev][Thread Next][Date Index][Thread Index][Old Index]

pkgsrc-2021Q2 branch announcement

- **To**: **netbsd-announce%netbsd.org@localhost**
- **Subject**: **pkgsrc-2021Q2 branch announcement**
- **From**: **Greg Troxel <gdt%netbsd.org@localhost>**
- Date: Thu, 01 Jul 2021 13:15:18 -0400

```
[resend; the copy I sent Monday appears to have vanished]

The pkgsrc developers are proud to announce the 71st quarterly release
of pkgsrc, the cross-platform packaging system.  pkgsrc is available
with more than 26,000 packages, running on 23 separate platforms; more
information on pkgsrc itself is available at https://www.pkgsrc.org/

In total, 210 packages were added, 103 packages were removed, and 2458
package updates (to 1900 unique packages) were processed since the
pkgsrc-2021Q1 release.  Updates include 419 Python packages, 207 Ruby
packages, 169 Haskell packages, 108 perl5 packages, and 83 R packages.

As always, many packages have been brought up to date relative to
upstream.  For the 2021Q2 release we welcome the following
notable packages additions and changes to the pkgsrc collection:

 - cmake 3.20.4
 - Firefox 78.11.0 (as an ESR), 88.0.1
 - Go 1.15.13, 1.16.6
 - LibreOffice 7.1.3.2
 - LXQt 0.17
 - MariaDB 10.4.18, 10.5.10
 - Node.js 12.22.1, 14.17.0
 - openblas 0.3.15
 - PHP 7.3.28, 7.4.20, 8.0.7
 - PostgreSQL 9.6.22, 10.17, 11.12, 12.7, 13.3
 - Python 3.8.10, 3.9.5
 - qemu 6.0.0
 - qgis 3.16.7
 - Rust 1.52.1
 - spotify-qt 3.6
 - SQLite 3.35.5
 - Syncthing 1.15.1
 - TeXLive 2021
 - Thunderbird 78.10.00
 - tor 0.4.5.9
 - Tor Browser 10.0.18
 - vlc-3.0.14
 - WebKitGTK 2.32.1

This branch we say notable goodbyes to:

 - a bunch of go packages, used only for building other departed packages
 - gcc48, gcc49, gcc5
 - postgresql95
 - ruby25

Changes to the pkgsrc infrastructure and notes:

 - Added PKGSRC_BLAS_TYPES which can be set to netlib or certain
   optimized variants, which causes many programs that need a BLAS
   implementation to use the specified one.
 - fftw (-long and -quad variants added, fftwf merged into fftw, more
   parallelization options, explicit SIMD support)
 - github submodule support was added.
 - Note that Firefox, Thunderbird and likely other packages with
   difficult dependencies do not build on NetBSD 8 and other systems
   with non-recent compilers.  Users who wish to run these programs
   are advised to update to NetBSD 9 or newer versions of other
   operating systems.

Instructions on using the binary package manager can be found at
https://pkgin.net, and pkgsrc itself can be retrieved from via CVS or
tar file, and also from a mirror at https://github.com/NetBSD/pkgsrc.
See https://www.netbsd.org/docs/pkgsrc/getting.html for instructions.
The branch name for the 2021Q2 branch is "pkgsrc-2021Q2".
```


- Prev by Date: **Public IRC chat channels moved to libera**
- Next by Date: **NetBSD Security Advisory 2021-001: Predictable ID disclosures in IPv4 and IPv6**
- Previous by Thread: **Public IRC chat channels moved to libera**
- Next by Thread: **NetBSD Security Advisory 2021-002: Incorrect permissions in kernfs**
- Indexes:
  - **reverse Date**
  - **reverse Thread**
  - **Old Index**

Home | **Main Index** | **Thread Index** | **Old Index**

3. RaspberryPIのNetBSDイメージ2021進捗どうですか

3.1. RaspberryPIのNetBSDイメージについて

今年もオープンソースカンファレンスごとにRaspberryPI用のNetBSDイメージを作って配布しています。この一年、どんなことがあったのか表にしてまとめてみました。

年月	NetBSD	mikutter	mlterm	OpenSSL	ネタ	OSC	URL
2019/8/3	8.99.51→9.99.1	3.9.2			9.0_BETA	OSC京都	http://mail-index.netbsd.org/port-arm/2019/07/31/msg005994.html
2020/08/28	9.99.71	4.0.6			RPI4+UEFI	OSC京都	http://mail-index.netbsd.org/port-arm/2020/08/27/msg006954.html
2020/09/19	9.99.72		3.9.0nb3		GCC9.3	OSC広島	http://mail-index.netbsd.org/port-arm/2020/09/17/msg006975.html
2020/10/24	9.99.74	4.1.2			NetBSD9.1	OSC東京秋	http://mail-index.netbsd.org/port-arm/2020/10/18/msg007015.html
2020/12/19	9.99.77		3.9.1		pkgdb	ODC	http://mail-index.netbsd.org/port-arm/2020/12/10/msg007120.html
2021/1/30	9.99.78	4.1.4		1.1.1i	python3.8	OSC大阪	http://mail-index.netbsd.org/port-arm/2021/01/20/msg007165.html
2021/2/27	9.99.80			1.1.1j	sudo	OSC東京春	http://mail-index.netbsd.org/port-arm/2021/02/27/msg007187.html
2021/4/2	9.99.81			1.1.1k	openssh8.5	NBUG2021/4	http://mail-index.netbsd.org/port-arm/2021/04/02/msg007213.html
2021/5/26	9.99.82	4.1.5	3.9.1nb1		NetBSD9.2	OSC名古屋	http://mail-index.netbsd.org/port-arm/2021/05/26/msg007290.html
2021/6/26	9.99.85				次はgcc10	OSC北海道	http://mail-index.netbsd.org/port-arm/2021/06/17/msg007309.html
2021/7/31	9.99.87				gcc10/ruby27	OSC京都	http://mail-index.netbsd.org/port-arm/2021/07/28/msg007381.html
2021/8/26	9.99.88				bind-9.16.20	ODC	http://mail-index.netbsd.org/port-arm/2021/08/23/msg007421.html
年月	NetBSD	mikutter	mlterm	OpenSSL	ネタ	OSC	URL

OSCはほぼ毎月のように日本各地で行われています。前に、OpenBSDのTheoさんに、自分のノートPCのアップデートをどのくらいの周期でやってるのかきいてみました。 2週間くらいごとかなと答えてくれて、ああだいたいそんなものなのかと思っていました。

NetBSDのイメージを配るとしたとき、どのくらいの周期でアップデートしていけばいいのでしょうか？イメージを配る理由は、何かソフトウェアが新しくなって新しい機能が入ったとか、ハードウェアのサポート種類が増えたとか、ソフトウェアの脆

Port-arm archive

[Date Prev][Date Next][Thread Prev][Thread Next][Date Index][Thread Index][Old Index]

2021-08-23-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)

- **To**: **port-arm%NetBSD.org@localhost**
- **Subject**: **2021-08-23-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)**
- **From**: **Jun Ebihara <jun%soum.co.jp@localhost>**
- Date: Mon, 23 Aug 2021 16:07:36 +0900 (JST)

```
I've updated 2021-08-23-netbsd-raspi-earmv6hf.img.gz for RPI.

https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2021-08-23-earmv6hf/2021-08-23-netbsd-raspi-earmv6hf.img.gz
https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2021-08-23-earmv6hf/MD5

Update:
- NetBSD 9.99.88 evbarm-earmv6hf 202108210520Z rpi.img from nyftp.
- gcc10
- ruby27
- pkgin support
  check /usr/pkg/etc/pkgin/repositories.conf.
  I add
  http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv6hf/2021-1
  for testing, with pkg_summery.gz .

- Raspberry Pi [0-3] have been supported in big-endian mode (Rin Okuyama)
XXX: should test big-endian image and pkgsrc

http://mail-index.netbsd.org/port-arm/2021/06/17/msg007310.html
 "earmv7hfeb works fine so far, pkgsrc included. aarc64eb doesn't boot
  on a Raspberry Pi 3,"

PR
#55505 RaspberryPi3A+ can't find Wi-Fi module
 http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=55505

- RPI4: testing on NetBSD/aarch64
  https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/README

Changes from 2021-06-18 version
https://github.com/ebijun/NetBSD/commit/461c616423ed15c5050bd6ae15912a5d75e93787#diff-
39e86155be9b5a0f937a5481e59c7784ee1515b36e534e2672c663d59d1ca679
https://github.com/ebijun/NetBSD/commit/5723965b1dda992a0cc9a646d8d852907026e3d4#diff-
39e86155be9b5a0f937a5481e59c7784ee1515b36e534e2672c663d59d1ca679

sysinfo:
 bind-9.16.20 NetBSD-9.99.88 sh-20181212-20210820202528Z
 userland-NetBSD-9.99.88/evbarm

pkgsrc:
 at-spi2-core-2.40.3 fltk-1.3.7 gtk3+-3.24.30 libidn-1.38
 libidn2-2.3.2 lz4-1.9.3nb1 p11-kit-0.24.0nb2 pango-1.48.7
 ruby27-addressable-2.7.0 ruby27-atk-3.4.2 ruby27-base-2.7.4
 ruby27-cairo-1.17.5 ruby27-cairo-gobject-3.4.2nb1
 ruby27-delayer-1.1.2 ruby27-delayer-deferred-2.2.0
 ruby27-diva-1.0.2 ruby27-gdk_pixbuf2-3.4.2
 ruby27-gettext-3.3.8 ruby27-gio2-3.4.2 ruby27-glib2-3.4.2
 ruby27-gobject-introspection-3.4.2 ruby27-gtk2-3.4.2nb1
 ruby27-httpclient-2.8.3 ruby27-instance_storage-1.0.0
 ruby27-locale-2.1.3 ruby27-memoist-0.16.2
 ruby27-mikutter-4.1.5 ruby27-mikutter-plugins-twitter-4.1
 ruby27-mini_portile2-2.5.0 ruby27-moneta-1.4.2
 ruby27-native-package-installer-1.1.1 ruby27-nokogiri-1.11.2nb1
 ruby27-oauth-0.5.6 ruby27-pango-3.4.2nb1
 ruby27-pkg-config-1.4.6 ruby27-pluggaloid-1.5.0
 ruby27-public_suffix-4.0.6 ruby27-racc-1.5.2nb1
 ruby27-red-colors-0.2.0 ruby27-red-datasets-0.1.2
 ruby27-simpleidn-0.2.1 ruby27-text-1.3.1
 ruby27-twitter-text-simpleidn-3.0.0.0
 ruby27-typed-array-0.1.2 ruby27-unf-0.1.4
 ruby27-unf_ext-0.0.7.7 ruby27-zip-2.3.2

Need Checking:
- bluetooth keyboard/mouse setting
- RPI camera module

Got Reports:
- HDMI output works very well however I wanted to change the resolution
  into 800x600 both tty console and X Window graphical modes. Where should I
  change it? As config.txt with hdmi_group=1, hdmi_mode=1 or hdmi_safe=1
  didn't work.
- USB input devices seem to work fine as well with my USB keyboard, mouse
  and barcode reader. However for my USB output device such as my Xprinter
  printer didn't work with device driver of ulpt(4). It is detected
  and working in OpenBSD.[9.0 release also occurs same error]

  [   195.114857] ulpt0 at uhub1 port 5 configuration 1 interface 0
  [   195.114857] ulpt0: Xprinter (0x0483) USB Printer P (0x5743), rev 2.00/1.00, addr 5, iclass 7/1
  [   195.114857] ulpt0: using bi-directional mode
```

```
  rpi# cat myfile.txt > /dev/ulpt0
  -sh: cannot create /dev/ulpt0: error 16
- with my Raspberry Pi 3B because every time I invoked this command
 "shutdown -h now" a kernel panic occurs relating to usbd_transfer.
- After shutting-down leaving the system unplugged, this will
  turn it's processor very very hot and seems harmful to the system.

pre-installed packages:
 https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/pkgsrc/pkginfo

Keyboard layout checkpoint:
 http://www.netbsd.org/docs/guide/en/chap-cons.html

/etc/wscons.conf
 #encoding sv
 #encoding us.swapctrlcaps
 encoding jp

System Update:

http://cvsweb.netbsd.org/bsdweb.cgi/src/distrib/sets/lists/base/shl.mi

mikutter support :
 I make sample API key,pre-installed.
 https://github.com/ebijun/NetBSD/tree/master/RPI/RPIimage/root/.mikutter/plugin
 https://github.com/Akkiesoft/how-to-make-mikutter-work-again
  cd /root/.mikutter
  git submodule add https://github.com/toshia/twitter_api_keys.git twitter_api_keys

RPI Wifi:
 http://mail-index.netbsd.org/port-arm/2019/08/31/msg006102.html

Overview:
  http://wiki.NetBSD.org/ports/evbarm/raspberry_pi/

QEMU,with GENERIC kernel : vexpress to GENERIC: testing
 https://github.com/ebijun/NetBSD/tree/master/vexpress/Boot
 https://github.com/ebijun/NetBSD/tree/master/vexpress/vexpress-v2p-ca15-tc1.dtb
 http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2021-08-23-earmv6hf/QEMU/netbsd-GENERIC.ub.gz
 http://mail-index.netbsd.org/port-arm/2017/06/02/msg004154.html

dmesg:
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI0
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI0W
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI2
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI2-1.2
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI3
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI3A+
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI3B+

Problem:
- CPUFLAGS=-mfpu=neon-vfpv4 breaks some programs
  http://gnats.netbsd.org/52528
- webkit24-gtk-2.4.11nb9 compile failed.
- dillo-3.0.5nb2 works with -O0
  thanx Jared McNeill.

security.pax.mprotect.enabled
  http://netbsd.org/gallery/presentations/msaitoh/2016_BSDCan/BSDCan2016-NetBSD.pdf
  man security
  man paxctl
  sysctl -a |grep pax
  If application failed, such as omxplayer.
  try to test
  sysctl -w security.pax.mprotect.enabled=0

Automatic resize partition: see /etc/rc.conf and /etc/fstab
  1. copy image to SD/MicroSD
  2. Boot
  3. Calculate and resize ld0 partition and automatic reboot
  4. after the reboot,root partition fit for your card.
    http://movapic.com/pic/20150416115108552fa22c4f225
    In this image, ld0a re-created with newfs -b 4096.

pkgsrc:
  # cd /usr
  # ftp http://cdn.netbsd.org/pub/pkgsrc/current/pkgsrc.tar.gz
  # ls /usr/pkgsrc                ... check if exists.
  # tar tzvf pkgsrc.tar.gz |head  ... check the archive
  # tar xzvf pkgsrc.tar.gz        ... extract
  # ls /usr/pkgsrc                ... check what extracted
  # pkg_chk -g                    ... List to/usr/pkgsrc/pkgchk.conf
  # (cd /usr/pkgsrc;cvs update -PAd) ... update
  # pkg_chk -un                   ... Update (listup)
  # pkg_chk -u                    ... Update

  I use /usr/pkgsrc with USB SSD disk.

Pre-compiled packages:
- Pre-compiled packages path setting: man 5 pkg_install.conf

See /etc/pkg_install.conf

PKG_PATH=http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv6hf/2021-1/

- If you use Official Package for NetBSD9:

set /etc/pkg_install.conf:
```

```
PKG_PATH=http://cdn.netbsd.org/pub/pkgsrc/packages/NetBSD/earmv6hf/9.0/

- If you update pkgsrc by yourself,comment out /etc/pkg_install.conf and check
 /etc/mk.conf.

Install application: man 1 pkg_add
  # pkg_add some_application_name

- omxplayer
  # pkg_add omxplayer
  # pkg_add youtube-dl
  # youtube-dl https://www.youtube.com/watch?v=wG8ZCC8IwvM
  # omxplayer *.mkv
  MPEG-2 license key:
   sysctl machdep.serial and convert to hex.

- Xfce4
  # pkg_add xfce4
  # startxfce4
   or edit /root/.xinitrc and comment out startxfce4

- seamonkey
  # pkg_add seamonkey
  # pkg_add seamonkey-l10n

- nodejs
  # pkg_add nodejs

  omxfinder (https://www.npmjs.com/package/omxfinder)
  # npm install -g omxfinder
  cd video archive directory,
  # omxfinder
  can start video viewing via file finder.

  SHARP MZ700 emulator
  # npm install -g mz700-js
  # cd /usr/pkg/lib/node_modules/mz700-js
  # npm start
  # mz700-js@0.0.0 start  /usr/pkg/lib/node_modules/mz700-js
  # access http://localhost:3000/MZ-700/client.html

- openjdk
  # pkg_add openjdk8

- gimp
  # pkg_add gimp

- mpv
  # pkg_add mpv

- emacs
  # pkg_add emacs
  # pkg_add anthy-elisp # for inputmethod/anthy

- evince
  # pkg_add evince

- typical apache+php environment
  # pkg_add ap22-php56

- gedit
  # pkg_add gedit

- sphinx
  # pkg_add py27-sphinx

- mcomix .. Book scanning data viewer
  # pkg_add py27-mcomix
  to avoid ImportError: cannot import name _getexif,
  use py27-Pillow package instead of py27-imaging.

- firefox52
  # pkg_add firefox52
  # pkg_add firefox52-l10n
  firefox52: fixed by Jared McNeill.
   http://mail-index.netbsd.org/pkgsrc-changes/2017/07/16/msg160171.html

- awscli
  # pkg_add py27-awscli
  # /usr/pkg/bin/aws ec2 describe-instances ....

Testing::

- midori
  # pkg_add midori
  XXX: start failed

- scribus
  # pkg_add scribus-1.4.3
  XXX: libpodofo.so.0.9.3 not found:need re-compile

- inkscape
  XXX: libboost_system.so.1.60 not found:need re-compile

- wordpress
  # pkg_add wordpress
  XXX: need php56-gd

- (shotwell)
  # pkg_add shotwell
```

```
USB mass storage boot
 https://www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes/msd.md

CrossCompiling:
  http://www.slideshare.net/junebihara18/netbsdworkshop

NetBSD GPIO DOC by Marina Brown
  https://github.com/catskillmarina/netbsd-gpio-doc/blob/master/README.md

I2C - "Raspberry Pi I2C implementation still broken?"
  http://mail-index.netbsd.org/port-arm/2015/02/10/msg002853.html
  "I can confirm the IOCTL is fixed, and can now successfully program I2C
  EEPROMs using NetBSD on the Pi."
  NetBSD RPi i2c sample code:
   https://gist.github.com/cr1901/76af0b3db9e9001a8d5b
   http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=48855
   http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=48932

XM6i - SHARP X68030 Emulator for NetBSD/x68k (Thanx isaki@)
  http://www.pastel-flower.jp/~isaki/XM6i/
  XM6i-0.55-netbsd7.0-earmv6hf-2015Q2.tar.gz
  pkg_add wxGTK30-3.0.2nb6
  XXX: use wxGTK30-3.0.2nb6 for XM6i-0.55

  https://twitter.com/isaki68k/status/625138538271502337

Todo:
- set2pkg: update via pkgsrc.
- pkg_in/pkg_summary
- Summarize /usr/tests atf result on earm/earmhf/earmv6hf.
- DTRACE http://wiki.netbsd.org/tutorials/how_to_enable_and_run_dtrace/
- yaft (yet another framebuffer terminal):
  https://github.com/uobikiemukot/yaft
- /dev/mem: http://mail-index.netbsd.org/port-arm/2015/03/12/msg002934.html
  "can't open /dev/mem" -> re-compile kernel with INSECURE.
- Sound output to the jack: port-arm/2015/03/12/msg002938.html
   $ mixerctl -v outputs.select
   outputs.select=auto  [ auto headphones hdmi ]

pkgsrc Todo:
- lang/go :
  earmv7hf: pkgsrc: go-1.9.3.tgz go14-1.4.3nb6.tgz or later.
  earmv6hf: pkgsrc: go14-1.4.3nb6.tgz

  Golang for NetBSD/arm problem summarized by @oshimyja
  http://www.yagoto-urayama.jp/~oshimaya/netbsd/netbsd_goarm.html
  http://mail-index.netbsd.org/port-arm/2015/08/02/msg003361.html
  https://twitter.com/oshimyja/status/604871730125864960
  https://twitter.com/oshimyja/status/840750347022876672
  https://github.com/golang/go/commit/30d60936d97423af0403f2d5395c604ac0ff3757
  runtime: fetch physical page size from the OS
   https://github.com/golang/go/commit/276a52de55fb48c4e56a778f1f7cac9292d8fad7
- gnuradio: g77 failed. need RTL2832U master.
  http://mail-index.netbsd.org/port-arm/2017/01/26/msg004090.html
- www/otter-browser: compiling.
- omxplayer: sometimes core dumps.
  -> add "gpu=256" to /boot/cmdline.txt, advice from Brandon Wickelhaus.


=====================

For Open Developers Conference 2021, NetBSD BOF.
I've updated raspberry-pi image.

2021 Aug.28 Sat 14:00-14:45 JST (UTC+9) ROOM C
https://event.ospn.jp/odc2021-online/session/375193

Join meeting with ZOOM/YoutubeLive
  https://ospn.connpass.com/event/216508/
YoutubeLive https://www.youtube.com/c/OSPNjp

http://www.re.soum.co.jp/~jun/ODC2021.pdf
http://www.jp.NetBSD.org/
https://www.facebook.com/NetBSD.jp
https://github.com/ebijun/NetBSD/blob/master/Guide/RPI/RPIupdate2021.rst

- NetBSD 9.99.88 earmv6hf rpi.img.gz base
- Connect HDMI,USB Keyboard,USB Mouse,Ether(dhcpd and ntpd will work)
- login root (no passwd)
- startx ,and icewm running.
- running mikutter.
  twitter client (net/mikutter) on 40inch home TV.
  http://movapic.com/ebijun/pic/5168479

  # startx
  # dillo &
  # mikutter &
  or
  # LANG=en_US.UTF-8 mikutter &
       English menu support. LANG environment setting on .xinitrc .

  ==== one or two or threee moment =====

Appear mikutter window. and mikutter-chan tell you,

"Well done on the installation!" [Next]
"Hi! This is mikutter-chan speaking to you,join the twitter with me!" [Next]
 1. "Click the link https:....." click the URL,
    copy URL into dillo.
```

```
 2. "login with the Twitter account you wish to use."
 3. "Go along until you see a 7-digit code and type it in at the top."
  -> get pin number.
  -> paste pin number to mikutter
 4. "Congratulations! You have attained achievement register_account!"

  Share your twitter timeline with your family!

Features:
  - fit size for 4GB SD Card
  - with X11
  - increase more inodes on /dev/ld0a
  - Recent current RPI kernel
  - USB/video support: as NetBSD-current
  - pre-build packages
    http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv6hf/2021-1/

Installed Packages:
  bash
  tcsh
  vlgothic-ttf
  icewm
  xli
  ruby27-mikutter
  uim
  fossil
  raspberrypi-userland
  sudo
  git-base
  zsh
  mlterm
  pkg_chk
  lintpkgsrc
  mozilla-rootcerts
  dillo
  medit
  lrzsz

To control HDMI output,add/delete "console=fb" on cmdline.txt.
If delete console=fb definition,you can get serial console instead.
 rpi$ more /boot/cmdline.txt
 root=ld0a console=fb


/root/.xinitrc
 setxkbmap -model jp106 jp

/root/.mikutter/plugin/ : http://yuzuki.hachune.net/wiki/Plugin
- display_requirements.rb [Pre installed]

Build sample script:
 https://github.com/ebijun/NetBSD/tree/master/RPI/RPIimage/Image

Guide:
 https://github.com/ebijun/NetBSD/blob/master/Guide/RPI.rst

BUGS:
1. port-arm/48855
  i2cscan on RPi NetBSD build finds device for all addresses
2. i2c problem reported from "its sead".
  /usr/sbin/i2scan on iic0 or iic1 often crashes the device (also with the -r
  (use writes) option)
3. cap_mkdb failed, on update terminfo db.
  https://twitter.com/uobikiemukot/status/487977340949893121

One more time: (we're gonna celebrate
- Everything you always wanted to know about six but were afraid to ask.

 1. login root
 2. mlterm-wscons
 3. ftp http://mlterm.sf.net/vimperator.six
 4. cat vimperator.six
 5. exit
 6. mlterm-wscons --rotate=left
 7. cat vimperator.six
    and @Chris_J_Baird only knows how to get to 1987.
 8. ftp http://kildall.apana.org.au/~cjb/mandel5.c
 9. cc mandel5.c
10. ./a.out 1024 728 -2 -1.5 4.0 |tee f
11. cat f

--
Jun Ebihara
```

- Prev by Date: **Re: serial console on Pi Zero W**
- Next by Date: **Re: serial console on Pi Zero W**
- Previous by Thread: **serial console on Pi Zero W**
- Indexes:
  - **reverse Date**
  - **reverse Thread**
  - **Old Index**

Home | **Main Index** | **Thread Index** | **Old Index**

Port-arm archive

[Date Prev][Date Next][Thread Prev][Thread Next][Date Index][Thread Index][Old Index]

2021-08-15-netbsd-armv7-earmv7hf.img

- **To**: **port-arm%NetBSD.org@localhost**
- **Subject**: **2021-08-15-netbsd-armv7-earmv7hf.img**
- **From**: **Jun Ebihara <jun%soum.co.jp@localhost>**
- Date: Sun, 15 Aug 2021 17:50:23 +0900 (JST)

```
I've updated 2021-08-15-netbsd9-armv7-earmv7hf.img.gz for RPI2/3.

http://cdn.netbsd.org/pub/NetBSD/misc/jun/armv7/2021-08-15-earmv7hf/2021-08-15-netbsd-armv7-earmv7hf.img.gz
http://cdn.netbsd.org/pub/NetBSD/misc/jun/armv7/2021-08-15-earmv7hf/MD5

Update:
- NetBSD 9.99.88 evbarm-armv7hf 202108130630Z armv7.img from nyftp.

- Recent Update from 2021-04-25
https://github.com/ebijun/NetBSD/commit/8c4509792a24add73604f24104d23770cbb3df5f

sysinfo:
 bind-9.16.15 bozohttpd-20210504 g++-10.3.0 gcc-10.3.0
 libdrm-3.5 libX11-7.1 NetBSD-9.99.88
 sh-20181212-20210812205318Z userland-NetBSD-9.99.88/evbarm

pkgsrc:
 bash-5.1.8nb3 cairo-1.16.0nb4 curl-7.78.0 dbus-1.12.20nb1
 dillo-3.0.5nb10 fltk-1.3.7 fossil-2.16 gd-2.3.2nb1
 gdk-pixbuf2-2.40.0nb2 git-base-2.32.0nb1 glib2-2.68.3nb1
 gnutls-3.7.2 gobject-introspection-1.68.0 gtk2+-2.24.33nb2
 gtk3+-3.24.30 harfbuzz-2.8.1 icu-69.1 libcups-2.3.3op2nb1
 libffi-3.3nb5 libgcrypt-1.9.3 libgpg-error-1.42 libidn-1.38
 libidn2-2.3.2 libimagequant-2.15.1 libtasn1-4.17.0
 libxkbcommon-1.3.0 libxml2-2.9.12nb1 libxslt-1.1.34nb6
 lintpkgsrc-4.96nb1 lz4-1.9.3nb1 m17n-lib-1.8.0nb5
 medit-1.2.0nb18 mlterm-3.9.1nb1 nettle-3.7.3 nghttp2-1.44.0
 p11-kit-0.24.0nb1 p5-Authen-SASL-2.16nb9 p5-Digest-HMAC-1.04nb11
 p5-Email-Valid-1.202nb5 p5-Error-0.17029nb2 p5-GSSAPI-0.28nb12
 p5-IO-CaptureOutput-1.1105nb2 p5-IO-Socket-INET6-2.72nb7
 p5-IO-Socket-SSL-2.071 p5-MailTools-2.21nb2
 p5-Mozilla-CA-20200520nb1 p5-Net-DNS-1.30nb1
 p5-Net-Domain-TLD-1.75nb5 p5-Net-IP-1.26nb9 p5-Net-LibIDN-0.12nb13
 p5-Net-SMTP-SSL-1.04nb5 p5-Net-SSLeay-1.90nb1 p5-Socket6-0.29nb3
 p5-TimeDate-2.33nb1 pango-1.48.7 pcre-8.45 pcre2-10.37
 perl-5.34.0nb2 pkg_install-20210410 pkgin-21.7.0
 png-1.6.37nb1 py38-cElementTree-3.8.11 py38-expat-3.8.11
 python38-3.8.11 qt5-qtbase-5.15.2nb6 qt5-qtx11extras-5.15.2nb2
 qt5-qtxmlpatterns-5.15.2nb2 ruby27-addressable-2.7.0
 ruby27-atk-3.4.2 ruby27-base-2.7.4 ruby27-cairo-1.17.5
 ruby27-cairo-gobject-3.4.2nb1 ruby27-delayer-1.1.2
 ruby27-delayer-deferred-2.2.0 ruby27-diva-1.0.2
 ruby27-gdk_pixbuf2-3.4.2 ruby27-gettext-3.3.8
 ruby27-gio2-3.4.2 ruby27-glib2-3.4.2
 ruby27-gobject-introspection-3.4.2 ruby27-gtk2-3.4.2nb1
 ruby27-httpclient-2.8.3 ruby27-instance_storage-1.0.0
 ruby27-locale-2.1.3 ruby27-memoist-0.16.2 ruby27-mikutter-4.1.5
 ruby27-mikutter-plugins-twitter-4.1 ruby27-mini_portile2-2.5.0
 ruby27-moneta-1.4.2 ruby27-native-package-installer-1.1.1
 ruby27-nokogiri-1.11.2nb1 ruby27-oauth-0.5.6
 ruby27-pango-3.4.2nb1 ruby27-pkg-config-1.4.6
 ruby27-pluggaloid-1.5.0 ruby27-public_suffix-4.0.6
 ruby27-racc-1.5.2nb1 ruby27-red-colors-0.2.0
 ruby27-red-datasets-0.1.2 ruby27-simpleidn-0.2.1
 ruby27-text-1.3.1 ruby27-twitter-text-simpleidn-3.0.0.0
 ruby27-typed-array-0.1.2 ruby27-unf-0.1.4
 ruby27-unf_ext-0.0.7.7 ruby27-zip-2.3.2
 shared-mime-info-1.10nb4 sudo-1.9.7p1 tcsh-6.22.02nb2
 tiff-4.3.0 uim-1.8.9pre20210104nb1 wayland-1.18.0nb3
 wget-1.21.1nb2 zstd-1.5.0

- packages:
  http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv7hf/2021-1/

- pre-installed packages:
 https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/earmv7hf/pkginfo

Keyboard layout checkpoint:
 http://www.netbsd.org/docs/guide/en/chap-cons.html

/etc/wscons.conf
 #encoding sv
 #encoding us.swapctrlcaps
 encoding jp

System Update:

http://cvsweb.netbsd.org/bsdweb.cgi/src/distrib/sets/lists/base/shl.mi

dmesg:
  https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv7hf/RPI2
```

Port-arm archive

[Date Prev][Date Next][Thread Prev][Thread Next][Date Index][Thread Index][Old Index]

2021-08-13-netbsd-raspi-aarch64.img

- **To**: **port-arm%NetBSD.org@localhost**
- **Subject**: **2021-08-13-netbsd-raspi-aarch64.img**
- **From**: **Jun Ebihara <jun%soum.co.jp@localhost>**
- Date: Fri, 13 Aug 2021 21:08:11 +0900 (JST)

```
I've updated 2021-08-13-netbsd-raspi-aarch64.img.gz for RPI4.

http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2021-08-13-aarch64/2021-08-13-netbsd-raspi-aarch64.img.gz
http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2021-08-13-aarch64/MD5

Topics:
- UEFI: v1.29
  Should save default boot device setting.
  It takes some time to boot from microsd.

- SD/MMC Configration
  Switch Default uSD/eMMC Routing: Arasan SDHCI -> EMMC2
  http://mail-index.netbsd.org/port-arm/2021/08/08/msg007393.html
  the Arasan SDHCI is used to connect bwfm(4).

bwfm0: Firmware file default:    brcmfmac43455-sdio.bin
bwfm0: Firmware file model-spec: brcmfmac43455-sdio.Raspberry Pi 4 Model B.bin
bwfm0: Found Firmware file: brcmfmac43455-sdio.bin
bwfm0: NVRAM file default:    brcmfmac43455-sdio.txt
bwfm0: NVRAM file model-spec: brcmfmac43455-sdio.Raspberry Pi 4 Model B.txt
bwfm0: autoconfiguration error: NVRAM file not available

- only for RPI4
  (fill dtb for RPI3?)
- build script
 https://github.com/ebijun/NetBSD/tree/master/RPI/RPIimage/Image/aarch64
- [Issue] Can't recognize DOS partition from Windows
  Should I Use MBR version image?
- [PR] diagnostic assertion "l->l_stat == LSONPROC" failed on RPI3
  http://gnats.netbsd.org/56264
- rpi cm4 + waveshare mini base board B seems stop after kernel load
  http://mail-index.netbsd.org/port-arm/2021/06/30/msg007343.html
- pkgin support
  edit /usr/pkg/etc/pkgin/repositories.conf
  cf.
  http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/aarch64/2021-1
  and pkgin update

UEFI: v1.29
 https://github.com/pftf/RPi4/releases
    Add Secure Boot default keys enrolment [tianocore/edk2-platforms@5ba08a9, tianocore/edk2-platforms@6196d93, tianocore/edk2@55266a9,
tianocore/edk2@1910759, etc.]
    Enable Boot Discovery Policy [tianocore/edk2-platforms@2e87ce8]
    Fix non-standard ACPI CIDs [tianocore/edk2-platforms@1942692]

sysinfo:
 NetBSD-9.99.88 sh-20181212-20210811111649Z
 userland-NetBSD-9.99.88/evbarm

pkgsrc:
 gtk3+-3.24.30 libidn-1.38 libidn2-2.3.2 lz4-1.9.3nb1
 pango-1.48.7 ruby27-zip-2.3.2

RPI4 status:
- boot via UEFI_VER=v1.29
 https://github.com/pftf/RPi4/releases

In this image,pre-setted RPI_EFI.fd
 Advanced Configration
 -> Limit RAM to 3GB    Disabled

- HDMI: works
- audio: vcaudio
  http://mail-index.netbsd.org/source-changes-d/2021/01/22/msg013133.html
- USB device: works
- bwfm0 Wireless network:?
  (works well on RPI3,not works on RPI4?)
- Pi 4 Bootloader Configuration
  https://www.raspberrypi.org/documentation/hardware/raspberrypi/bcm2711_bootloader_config.md

Update:
- NetBSD-current 9.99.88 evbarm-aarch64 202108111200Z arm64.img from nyftp.
- pkginfo
 https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/pkginfo
- sysinfo
 https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/sysinfo

Update from 2021-07-27:
https://github.com/ebijun/NetBSD/commit/ff8782a194f788aea18ac53d15f9ee519c856f2e#

- mate/firefox/inkscape/libreoffice works well
```

```
    pkg_add mate
    pkg_add gvfs
    pkg_add firefox
    pkg_add firefox-l10n
    pkg_add libreoffice
    pkg_add inkscape
    pkg_add gimp
    pkg_add scribus
    pkg_add py38-sphinx
    pkg_add wxGTK30
    pkg_add gnome-keyring

- Sphinx with latexpdf
  https://github.com/ebijun/NetBSD/blob/master/Guide/latexpdf.rst

- pkgin: upgrade failed
- raspberry-pi userland: compile failed

Pre-installed packages:
  bash
  tcsh
  vlgothic-ttf
  icewm
  xli
  ruby27-mikutter
  uim
  fossil
  sudo
  git-base
  zsh
  mlterm
  pkg_chk
  lintpkgsrc
  mozilla-rootcerts
  dillo
  medit
  lrzsz

- pre-build packages:
  See /etc/pkg_install.conf
  PKG_PATH=http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/aarch64/2021-1/

pkgsrc:
  # cd /usr
  # ftp http://cdn.netbsd.org/pub/pkgsrc/current/pkgsrc.tar.gz
  # ls /usr/pkgsrc                ... check if exists.
  # tar tzvf pkgsrc.tar.gz |head  ... check the archive
  # tar xzvf pkgsrc.tar.gz        ... extract
  # ls /usr/pkgsrc                ... check what extracted
  # pkg_chk -g                    ... List to/usr/pkgsrc/pkgchk.conf
  # (cd /usr/pkgsrc;cvs update -PAd) ... update
  # pkg_chk -un                   ... Update (listup)
  # pkg_chk -u                    ... Update

EEPROM verson:
 Check&update with Raspberry Pi OS (pi:raspberry)
 pi@raspberrypi:~ $ cat /etc/default/rpi-eeprom-update
 FIRMWARE_RELEASE_STATUS="stable"
 pi@raspberrypi:~ $ sudo rpi-eeprom-update
 BCM2711 detected
 Dedicated VL805 EEPROM detected
 BOOTLOADER: up-to-date
 CURRENT: Thu 16 Jul 15:15:46 UTC 2020 (1594912546)
 LATEST: Thu 16 Jul 15:15:46 UTC 2020 (1594912546)
 FW DIR: /lib/firmware/raspberrypi/bootloader/stable
 VL805: up-to-date
 CURRENT: 000138a1
 LATEST: 000138a1

--
Jun Ebihara
```

- **Follow-Ups**:
  - **Re: 2021-08-13-netbsd-raspi-aarch64.img**
    - *From:* Jared McNeill

- Prev by Date: **Re: 2021-07-27-netbsd-raspi-aarch64.img**
- Next by Date: **2021-08-15-netbsd-armv7-earmv7hf.img**
- Previous by Thread: **sunxigpio vs. LOCKDEBUG (PR 54664)**
- Next by Thread: **Re: 2021-08-13-netbsd-raspi-aarch64.img**
- Indexes:
  - **reverse Date**
  - **reverse Thread**
  - **Old Index**

Home | **Main Index** | **Thread Index** | **Old Index**

Port-arm archive

[Date Prev][Date Next][Thread Prev][Thread Next][Date Index][Thread Index][Old Index]

pinebook status update (20210819)

- **To**: **port-arm%netbsd.org@localhost**
- **Subject**: **pinebook status update (20210819)**
- **From**: **Jun Ebihara <jun%soum.co.jp@localhost>**
- Date: Thu, 19 Aug 2021 13:44:25 +0900 (JST)

```
Topics:
- /usr/X11R7/lib/modules/drivers/modesetting_drv.so issue

X hungs with xfce4/mate desktop.icewm works.
https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/X/Xorg.0.log

depends on /usr/X11R7/lib/modules/drivers/modesetting_drv.so
NetBSD-9.99.86-evbarm-aarch64-202107140840Z failed.
NetBSD-9.99.85-evbarm-aarch64-202106200030Z works well.

- graphical mixer for NetBSD audio.

on pinebook:
 Select a mixer device:
 [*] /dev/mixer0: sun50i-a64-audi  ausoc
 [ ] /dev/mixer1: hdmi-audio  ausoc

on pinebook pro:
XXX: [inputs] tab controls audio output level.
 https://twitter.com/ebijun/status/1395184397599469568

- Audio CD with wavpack
Encode:
 On Windows: Exact Audio Copy & foobar2000
 On pkgsrc:
 1. pkg_add abcde; pkg_add glyr
    abcde -d /dev/rcd0a -B -o wav -1 -a default,cue
    -> wav,cue,jpg file created.
 2. pkg_add wavpack
    wavpack -h .wav --write-binary-tag "Cover Art (Front)=@.jpg" -w "cuesheet=@.cue"
    -> wv file created.

Play:
  qmmp: can play wavpack file with cue. How can I show Cover Art?
   Setting->Output->OSS plugin(liboss.so)

- touchpad: sometimes cursor far away.
- audio0 input issue
  http://mail-index.netbsd.org/port-arm/2021/02/22/msg007185.html
  "Is audio input from the inbuilt microphone(s) working?  I get "read failed:
  Resource temporarily unavailable" from audiorecord"
  audio0(ausoc1): trigger_input failed: errno=5
  audio0(audoc1): device timeout

pkgsrc:
- works : inkscape,scribus,seamonkey,minitube,mikutter,xournalpp,libreoffice

- firefox 90.0 and libreoffce 7.1.4.2 on NetBSD 9.99.85/aarch64 on pinebook.

 zoom meeting with firefox on NetBSD/aarch64 on pinebook pro
  with UserAgent switcher addon as Linux.
 Send: Share Screen,camera/USB Camera
 Problem: Can"t show others screen/in-camera.

- pkgin support
 make pkg_summary. and pkgin works.
 cd /usr/pkgsrc/packages/All
 find . -name '*.tgz' -exec pkg_info -X {} \; > pkg_summary
 gzip -f pkg_summary

 pkgsrc packages:
 http://cdn.netbsd.org/pub/NetBSD/misc/jun/aarch64/aarch64/2021-1/

- nono: luna68k/luna88k emulator
 add /etc/mk.conf
  ACCEPTABLE_LICENSES= nono-license
 cd /usr/pkgsrc/emulators/nono;make ;make package-install
- search soumd mixer GUI to manage mixerctl

ToDo:
- 3D acceleration

- Pinebook Pro
 mate/xfce4/camera/WiFi
 ToDo:
 - HDMI output via USB-C
 - Audio output: can't switch speaker & headphone out

- Pinebook
 mate/xfce4/camera/HDMI
 ToDo:
```

```
 - internal WiFi

System Updates:

sysinfo:
 NetBSD-9.99.88 sh-20181212-202108152222252Z
 userland-NetBSD-9.99.88/evbarm

pkgsrc:
 ImageMagick-7.1.0.4nb1 ImageMagick6-6.9.11.7nb10
 SDL-1.2.15nb35 SDL2-2.0.16 adwaita-icon-theme-40.1.1
 alsa-lib-1.2.5.1nb1 ap24-php74-7.4.22nb6 atril-1.24.1
 audacious-plugins-4.1nb5 automake-1.16.4 bash-5.1.8nb3
 blas-3.9.1 caja-1.24.1 caja-extensions-1.24.1 cblas-3.9.1nb1
 cmake-3.21.1 cmark-0.30.1 cmus-2.9.1nb3 curl-7.78.0
 dav1d-0.9.1 dbus-python-common-1.2.18 emacs26-26.3nb15
 enchant2-2.3.1 engrampa-1.24.2 eom-1.24.2
 ffmpeg2-2.8.17nb11 ffmpeg3-3.4.8nb12 ffmpeg4-4.4nb7
 firefox-91.0 firefox-l10n-91.0 fltk-1.3.7
 fluidsynth-2.2.2nb1 go-1.16.7 go-bin-1.14.2nb4
 go115-1.15.15 go116-1.16.7 gtk3+-3.24.30 help2man-1.48.4
 imlib2-1.7.2 inkscape-1.1nb1 jasper-2.0.33
 kdelibs4-4.14.38nb22 koruri-ttf-20210720
 lapack-3.9.1nb1 lapacke-3.9.1nb2 libXft-2.3.4
 libcares-1.17.2 libidn-1.38 libidn2-2.3.2
 libmatekbd-1.24.1 libmateweather-1.24.1 libmikmod-3.3.11.1nb7
 libreoffice-7.1.5.2 librsvg-2.50.7 libuv-1.42.0
 libxml2-2.9.12nb1 lz4-1.9.3nb1 mate-1.24.1
 mate-applets-1.24.1 mate-backgrounds-1.24.2
 mate-common-1.24.2 mate-control-center-1.24.2
 mate-desktop-1.24.1 mate-media-1.24.1 mate-menus-1.24.1
 mate-notification-daemon-1.24.2 mate-power-manager-1.24.3
 mate-sensors-applet-1.24.1 mate-settings-daemon-1.24.2
 mate-themes-3.22.22 minitube-3.9 mkvtoolnix-60.0.0
 mozo-1.24.1 mpv-0.33.1nb4 mysql-client-5.7.35
 nghttp2-1.44.0 nodejs-14.17.4 nss-3.69 openal-soft-1.21.1nb1
 osabi-NetBSD-9.99.88 pango-1.48.7 phonon-4.10.3nb8
 php-7.4.22 php74-curl-7.4.22nb10 php74-fpm-7.4.22nb6
 php74-gd-7.4.22nb1 php74-iconv-7.4.22 php74-intl-7.4.22nb6
 php74-json-7.4.22 php74-mbstring-7.4.22 php74-pdo-7.4.22
 php74-pdo_sqlite-7.4.22nb6 php74-posix-7.4.22 php74-sqlite3-7.4.22nb6
 php74-zip-7.4.22nb5 php74-zlib-7.4.22nb1 pkgconf-1.8.0
 pkgin-21.7.0 poppler-21.07.0 poppler-cpp-21.07.0
 poppler-glib-21.07.0 poppler-includes-21.07.0 pulseaudio-15.0
 py27-gobject-2.28.7nb3 py38-charset-normalizer-2.0.3 py38-cython-0.29.24
 py38-dbus-1.2.18 py38-iso8601-0.1.16 py38-numpy-1.20.3nb1
 py38-requests-2.26.0 py38-scour-0.38.2 py38-serial-3.5
 py38-setuptools-57.4.0 py38-sphinxcontrib-serializinghtml-1.1.5
 qmmp-1.5.1 qr-code-generator-1.7.0 qt5-qtmultimedia-5.15.2nb5
 qt5-qttools-5.15.2nb5 rclone-1.55.0nb4 re2c-2.2 rhash-1.4.2
 seamonkey-2.53.6nb5 talloc-2.3.3 u-boot-pinebook-pro-2021.07
 upower-0.99.12 w3m-0.5.3.0.20210102nb2 xorg-cf-files-1.0.7

System Updates:
https://github.com/ebijun/NetBSD/commit/aeb9c0272203be2d79967cfdcafd17182202101b
https://github.com/ebijun/NetBSD/commit/acbed05e8c4c468143fcc49bff489e1a56de15b5

Tips:
- System Clocks
 % sysctl -a |grep freq

 Install pkgsrc/sysutils/estd from pkgsrc and start it on bootup
 for automatic up-/downscaling.

pinebook:
 machdep.cpufreq.cpu0.available = 1152 1104 1008 816 648 408
 # sysctl -w machdep.cpufreq.cpu0.target=1152

pinebook-pro:
 machdep.cpufreq.cpu0.available = 1416 1200 1008 816 600 408
 machdep.cpufreq.cpu4.available = 2000 1800 1608 1416 1200 1008 816 600 408
 # sysctl -w machdep.cpufreq.cpu0.target=1416
 # sysctl -w machdep.cpufreq.cpu4.target=2000

- Battery Status
  envstat
  xbattbar
XXX: link from dbus or mate-power-manager?

- HDMI port works on pinebook.
 Connect display to pinebook HDMI port
 # pkg_add arandr
 % arandr

- mate: pkgsrc/meta-pkgs/mate
 https://twitter.com/ebijun/status/1086814579647102976
 pkg_add mate
 echo "mate-session" >> .xinitrc
 startx

- ibus
 export XMODIFIERS=@im=ibus

- Add eDP transmitter and enable display pipeline
 http://mail-index.netbsd.org/source-changes/2019/02/03/msg103047.html
 arandr works well: display up-side down
 https://twitter.com/ebijun/status/1092885154127724544

- keymapping: [Shift][<>] on 11inch
 https://wiki.netbsd.org/ports/evbarm/allwinner/#index3h2
```

```
- replace AC Adapter for heavy load
 Anker PowerPort10 A2133+USB DC Cable with 3.5mm/1.35mm connecter
 USB-C laptop charger from HP

- USB Wifi: urtwn0/run0 works
 urtwn0: I-O DATA DEVICE, INC. (0x4bb) WN-G150UM (0x94c)
 pinebook: internal Wi-Fi not supported
 pinebook pro: internal bwfm0 works,sometimes hangs


Version:
 NetBSD 9.99.88 (GENERIC) #0: Sun Aug  8 11:11:29 UTC 2021  mkrepro%mkrepro.NetBSD.org@localhost:/usr/src/sys/arch/amd64/compile
/GENERIC

dmesg:
 https://github.com/ebijun/NetBSD/blob/master/dmesg/aarch64/pinebook
 https://github.com/ebijun/NetBSD/blob/master/dmesg/aarch64/pinebook-pro

System image: dd to eMMC.
 http://www.invisible.ca/arm/

 pinebook-pro:
 1. Insert microSD, boot (Linux from eMMC comes up),
 2. hit "Restart",
 3. It will boot from microSD.
 4. dd'ing to eMMC for installation, the eMMC is faster
- eMMC can disable with switch inside,and boot MicroSD.

 Getting Started with NetBSD on the Pinebook Pro by Benny Siegert
 https://bentsukun.ch/posts/pinebook-pro-netbsd/

sysinfo:
 https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/sysinfo

pkginfo:
 https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/pkginfo

pkgchk.conf:
 https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/pkgchk.conf
  cp pkgchk.conf /usr/pkgsrc;pkg_add pkg_chk; pkg_chk -au

--
Jun Ebihara
```

- Prev by Date: **Re: NetBSD, rPi and e-ink readers**
- Next by Date: **RPI4 (8GB) + UEFI -> bricked?**
- Previous by Thread: **NetBSD, rPi and e-ink readers**
- Next by Thread: **Re: pinebook status update (20210819)**
- Indexes:
  - **reverse Date**
  - **reverse Thread**
  - **Old Index**

Home | **Main Index** | **Thread Index** | **Old Index**

Port-arm archive

[Date Prev][Date Next][Thread Prev][Thread Next][Date Index][Thread Index][Old Index]

Re: pinebook status update (20210819)

- **To**: **mmwidup%gmail.com@localhost**
- **Subject**: **Re: pinebook status update (20210819)**
- **From**: **Jun Ebihara <jun%soum.co.jp@localhost>**
- Date: Wed, 25 Aug 2021 11:04:41 +0900 (JST)

```
From: Matt <mmwidup%gmail.com@localhost>
Subject: Re: pinebook status update (20210819)
Date: Thu, 19 Aug 2021 06:21:02 -0500

> I'm not sure if this solves the Pinebook Pro touchpad issues for everyone,
> but the latest firmware update for the touchpad (As of 28 July 2021) seems
> to solve all but UI settings related touchpad problems.  I could not get
> the updater to work in NetBSD; I had to install GNU/Linux on a micro SD
> card to install the new firmware.  But it solved all the touchpad issues I
> had.

Thanx!
With https://github.com/dragan-simic/pinebook-pro-keyboard-updater ,
I've update my pinebook-pro keyboard/touchpad firmware.

need: MicroSD card,USB keyboard/mouse.

1. Copy Manjaro ARM pinebook pro image to MicroSD

disklabel sd0   ... connect microsd with USB dongle, as sd0.
dd if=Manjaro-ARM-xfce-pbpro-21.08.img of=/dev/rsd0d bs=4m

2. Boot Manjaro ARM from MicroSD

setup wireless LAN with clicking icon which smells like WiFi spirit.

3. make updater

git clone https://github.com/dragan-simic/pinebook-pro-keyboard-updater.git
cd pinebook-pro-keyboard-updater
sudo pacman -Syy base-devel libusb vim
make

4. update firmware

My pinebook pro is "ANSI" model. [SHIFT][Z][X]..

# Execute step-1
sudo ./updater step-1
# connect USB keyboard and mouse
sudo poweroff

# Turn your Pinebook Pro on again, then run step-2
# connect USB keyboard and mouse
sudo ./updater step-2 ansi
# plug off USB keyboard and mouse,
# within 5 seconds,during firmware update count down.
# update succeeded.
# connect USB keyboard and mouse
sudo poweroff

I also install "Revised" firmware.
# Turn it on again, then update to the revised ANSI firmware
sudo ./updater flash-kb firmware/default_ansi.hex


--
Jun Ebihara
```

- **Follow-Ups**:
  - **Re: pinebook status update (20210819)**
    - *From:* Jason Thorpe
- **References**:
  - **pinebook status update (20210819)**
    - *From:* Jun Ebihara

- Prev by Date: **Re: serial console on Pi Zero W**
- Next by Date: **Re: pinebook status update (20210819)**
- Previous by Thread: **pinebook status update (20210819)**
- Next by Thread: **Re: pinebook status update (20210819)**
- Indexes:
  - **reverse Date**
  - **reverse Thread**
  - **Old Index**

NetBSD Developer Documentation

This page is the jumping-off point into the NetBSD developer documentation tree. The information here is likely to be most useful for NetBSD developers, but doesn't really have to be kept “confidential.”

General

- New NetBSD developer application procedure
- How to de-lint files
- Commit Guidelines
- NetBSD source code style guide
- Handling Problem Reports
- Role Accounts
- CVS Repository Issues
- Working with Mercurial
- Port/Feature Cross Reference
- Adding a new port
- CVS Server access
- PGP usage notes
- Other notes

Packages

- Pkgsrc Developer Information

Releng

- Building and Release Engineering
- Releng Server (Autobuild, etc)

WWW / Mirrors

- The www mailing list
- The mirrors mailing list
- News story guidelines
- Editing htdocs
- Translating htdocs

Home
Recent changes
NetBSD blog
Presentations
About
Developers
Gallery
Ports
Packages
Documentation
FAQ & HOWTOs
The Guide
Manual pages
Wiki
Support
Community
Mailing lists
Bug reports
Security
Developers
CVSWeb
Mercurial
Cross-reference
Release engineering
Projects list

New NetBSD developer application procedure

NetBSD Foundation membership application process:

- Existing members decide to sponsor a potential member.

  First, existing members of the Foundation must decide that they are willing to sponsor a potential member. (This can happen as a result of discussions amongst the members, or, for instance, because the potential member asks someone to sponsor them.)

  These members should communicate this to the potential member, and should send to her/him a membership application form, which can be found in the file membership-app in this directory.

- The potential member fills out the form, and returns it to the Membership committee membership-exec@NetBSD.org by e-mail.

  The form may be filled out in consultation with one or more developers, but is intended to be primarily filled out by the applicant.

  At this point an entry for the potential member is created in the "completed-forms" subdirectory. At all later stages, it should be kept up to date.

- The Membership committee informs the Foundation's members about the potential member, and solicits comments about the application.

  The form to be used to solicit opinions can be found in the file "rfc-template" in this directory. There's also a script to automatically fill out and send the form ("send-rfc-form").

  A period of 14 calendar days shall be allowed for comments.

  The Membership committee also informs the applicant that his application is received and when the applicant will next hear from the committee.

- The Membership committee, weighing the application and the comments, decides whether to offer membership to the applicant.

  If membership is denied, that should be communicated to the applicant along with some information about why, and this process stops.

- The Membership committee sends an invitation for membership and a membership agreement form to the applicant.

  An entry for the applicant is made in the "status" file, filling in all pertinent information from the corresponding entry in the "completed-forms" directory.

- The applicant returns the signed membership agreement, if he or she accepts its terms, to the Secretary of the NetBSD Project, whose postal address will be specified in the invitation letter.
- The Membership committee then asks admins@NetBSD.org to create accounts for the new member.
- The administrators create accounts for the new member, and update the "status" file and the corresponding entry in the "completed-forms" directory accordingly;
  - Admins inform the applicant, his/her sponsor(s), and the Membership EC when this is done
  - Sponsor(s) welcome applicant on the NetBSD developers mailing list
  - www announces applicant(s) on www.NetBSD.org on a programatic base (e.g. once per month, if any) and informs Membership EC about announcement
  - Membership EC posts new membership announcement (text from www) to netbsd-announce
  - Membership EC reviews new members's application, makes sure that all files have been updated, and closes the case

```
$NetBSD: membership-app.txt,v 1.2 2019/04/24 16:43:39 maya Exp $

NetBSD Foundation Membership Application Form

This form should be filled out by the applicant, with help from their
sponsors.

After completing this form, please send it to the NetBSD Foundation
Executive Committee for Membership, at the e-mail address
<membership-exec@NetBSD.org> and CC: your sponsors.

When filling out this application, please provide information in all
of the fields (except those in section 5).  If in any free-form text
field you need more lines than are provided by the form, add
additional lines (with the correct prefix) as needed.

If English is not your native language, then please fill the form in
to the best of your ability and pass it to someone else for proofing
and correction of spelling and grammar, particularly sections 2 and 3.


Section 1 - Contact info
---------

This is the relevant contact information for the prospective developer.

1A: Full name:
1B: E-mail address:
1C: Requested NetBSD.org login:


Section 2 - Proposed work area(s)
---------

This is a free form text field where your prospective work areas
are listed.  This field will be used as part of the posting to the
netbsd-announce@NetBSD.org mailing list.

2A:
2A:
2A:


Section 3 - Accomplishments and Qualifications
---------

This section contains free form text fields where you may list your
recent accomplishments or contributions to the NetBSD project.

Section 3A should include a summary of your qualifications.

3A:
3A:
3A:

Section 3B should include a small but detailed set of examples of your
technical contributions and qualifications.  This will be used by the
Membership committee and by other members of the Foundation when
evaluating your membership application.  Include URLs or other
citations for examples, as appropriate.

3B:
3B:
3B:


Section 4 - Sponsors
---------

In this section, please list the e-mail addresses of the NetBSD
Foundation member or members who are sponsoring your application, one
per line.  (Add more lines if necessary.)

4A:


Section 5 - Paper trail
---------

Please leave this section blank; it will be filled out when tracking
your membership application.

5A: Received by Membership committee:
5B: Sent for member comments:
5C: Decision (ACCEPT or REJECT):
5D: Decision published:
5E: Membership agreement sent:
5F: Membership agreement received:
5G: Accounts created:
5H: WWW processed:
5I: Announcement processed:
```

NetBSD Commit Guidelines

The following Commit Guidelines define the Project's standards for making commits to the source tree:

1. ## Commit only code you are familiar with.

   If you are not sure if the code you plan to commit is acceptable (e. g. when taking code that was submitted with a problem report), ask a developer who is familiar with that part of the system for review. If you're new to the Project, check with your sponsor.

2. ## Do not commit tainted code to the repository.

   If you commit code that was not written by yourself, double check that the license on that code permits import into the NetBSD source repository, and permits free distribution. Check with the author(s) of the code, make sure that they were the sole author of the code and verify with them that they did not copy any other code.

3. ## Do not commit code from foreign trees.

   Do not commit code from a tree checked out from anywhere but from cvs.NetBSD.org. Note that all developers have access to the private rsync-over-ssh service on cvs.NetBSD.org.

4. ## The more intrusive your changes are the higher is the level of required prior approval.

   - "Obvious" fixes can be committed without any prior discussion or review. (The definition of "obvious" in the GCC Project is: "could not possibly cause anyone to object." We adopt this definition here)
   - All other (i. e. "non-obvious") fixes *should* have a review.
   - Implementing (significant) new features requires a prior discussion on an appropriate technical mailing list.
   - Adding a completely new package (e.g. openldap) requires prior discussion on a mailing list *and* approval from core.

5. ## Commit only code that you have tested.

   Make sure that your code actually works as expected, by compiling and running the code that is affected by your change with your system's tools. If you changed a man page, make sure that groff/nroff creates the formatted man page you expect.

   For normal commits (to the trunk), test that your code works on -current. Prior to requesting a pull-up to a branch, test the very changes you will request from releng on the respective release branch.

   Run all relevant tests from `/usr/tests` or ideally the whole test suite. The logs of the automated test runs are currently available here. Long-term regressions (build breakage or failing tests) are not acceptable and changes causing such will be backed out if the regressions are not addressed.

6. ## Group commits together that are part of the same fix.

   Rototilling a make variable that affects 50 Makefiles should get one commit for all of them together.

7. ## Each commit should be a separate patch/fix /addition/etc.

   Don't fix 3 bugs with the same commit and roll that into "Fixed some bugs". Fix one, test, commit, rinse, repeat. This makes life infinitely easier for releng to pull up fixes since often not everything applies to a given branch.

8. ## Do not mix functionality or bug-fix patches with whitespace/layout updates.

   Do those separately (apart from the general requirement that changes be made in such a way that it is easy to discern what each fix did, as described in item 6, there will be pullup issues with largely changed files from trunk->branch when mixing whitespace with functionality fixes).

9. ## Clearly document the reasons for your changes in the commit log.

   Detail to some degree what was changed and why. This doesn't need to be a code review/walkthrough but it should be informative to someone reading the log and looking over the diff in 6 months. The focus should be on "why" since the target audience reading the logs can usually figure out "what" from looking the diffs. As an exercise, consider the difference of usefulness between "set i to 1" and "initialize iterator correctly to fix a rarely triggering bug in memory handling".

   Also, "fixed some stuff" or "cvs-1.10.0" isn't informative whereas the following example is:

   ```
   "Print useful line number on error while executing .for directive."
   ```

   If your change fixes a PR, document it with an appropriate message. Using the template “PR category/bug-id” in your commit message will also append it to the respective problem report in our bug database:

   ```
   "Closes PR bin/6666"
   ```

   If your code has been reviewed by someone else, document this:

   ```
   "Reviewed by <mrg>"
   ```

   (Please note that a good commit log doesn't relieve the need for good documentation in the program itself.)

10. ## Give proper credit.

    If you commit code that had been submitted in a PR, give proper credit, like:

    ```
    "Code submitted in PR lib/393939 by Joe Doe"
    ```

    Since the commit messages will eventually appear on the source-changes mailing list which is also available via web, specifying the e-mail addresses of the PR submitter should generally be avoided.

    If you took code from an other Open Source Project, give credit, like:

    ```
    "From FreeBSD"
    ```

11. ## Do not revert other developer's commits.

If you do not agree with another developer's commit, do not revert it on your own. Contact the other developer and explain to him or her the issues you have with the commit in question. Ask the other developer to back out the changes instead of doing it yourself.

If no agreement can be reached, contact the Core Team <core@NetBSD.org>, which will serve as mediation authority.

Resolving Problem Reports

Resolving Problem Reports

Resolving Problem Reports

Introduction

The NetBSD Project uses the GNATS “Problem Report” database to accept and track bug/problem reports from all users of NetBSD. When used properly, this facility allows us to make sure that no problem with the NetBSD software goes unfixed.

The GNATS database uses Internet E-mail as its principal submission mechanism, and keeps the problem reports (commonly abbreviated “PR”) in Internet E-mail format, with an extended header format in the body of the message. The database uses one file per PR, and each category is a directory, in a manner similar to the MH mail system, and NetNews.

Accessing the Problem Report Database

Web Interface

There is a web-based interface to the GNATS database that has both a tree of database summary pages, and a search facility. The web interface has some limitations:

- The web interface *does not* provide access to any PR marked “confidential”.
- PRs displayed by a web browser have been through sufficient transformation that “cut & paste” is not likely to do what you expect (i.e. patches will not apply properly because of white space substitution, uuencoded text will not decode properly).

The netbsd-bugs Mailing List

All PRs except confidential ones are sent to the netbsd-bugs mailing list, so that subscribers can see each PR as it is added to the database.

The GNATS commands on gnats.NetBSD.org

The GNATS database lives on the host gnats.NetBSD.org. All developers are given an account on that host so that they can directly manipulate PRs, and access confidential PRs. The commands **edit-pr** and **query-pr** live in `/usr/pkg/bin`; Make sure you have that directory in your `$PATH`.

The **query-pr** command is a proper database query interface; it has a large number of options to search the database. Once you know the PR number of the PR you wish to manipulate, you can:

- **query-pr --full** <number>

  This will dump out the full PR without any transformation to standard output.

- **edit-pr** <number>

  This command will start up a text editor (vi by default; but this can be overridden by the `$EDITOR` or `$VISUAL` environment variables), so that changes can be made to the PR.

  Please don't kill your edit-pr without checking for a left-over shell afterwards (also check if you lost your connection with edit-pr running).

For more information please consult `edit-pr(1)` or `query-pr(1)` man pages. As with most GNU software, there are also “info” pages available through the **info** command. Also, invoking **edit-pr** or **query-pr** without any arguments will cause it to give its usage message.

The Common Reasons to edit-pr

The most common changes made to a PR are:

1. Change the PR's `>State:` field to one of the values listed in state as it progresses through the process of resolving it.

   The person listed in the `>Responsible:` field of the PR should be making these state changes, as it is necessary.

2. Change the PR's `>Responsible:` field to the account name of a developer who will handle the PR. This person becomes the PR Submitter's primary contact for getting the problem resolved.

   This field can have any username from /etc/passwd on `gnats.NetBSD.org` and anyone listed in the `/home/gnats/gnatsdb/gnats-adm/responsible` file.

   All PRs get a default Responsible Person when they are initially filed, appropriate to the category in which the PR was filed (e.g. “security-officer” for PRs in the “security” category). The `/home/gnats/gnatsdb/gnats-adm/categories` file lists the default Responsible Person for each category.

   There is also a table of developers responsible for current PRs.

3. Change the PR's `>Category:` field.

   It is not unusual for a PR Submitter to have made a poor category choice. There is a list of PR categories and their definitions and a table of current PRs by category.

   The `/home/gnats/gnatsdb/gnats-adm/categories` file lists the valid categories and the default Responsible Person. It is usually necessary to change the `>Responsible:` field at the same time to a more appropriate person. Most often, the correct Responsible Person is the default Responsible Person for the new category.

4. Add themselves to the PR's `>Notify-List:` field.

   If you have an interest in the PR without being the submitter nor `>Responsible` for it, add yourself to this field. You may have to insert it.

5. Change other PR Fields according to the analysis of the responsible developer.

Edit each field you think needs modifying, then save the file and exit the editor. The **edit-pr** will then prompt for a short explanation to be typed for each key field change (mostly `>State:` and `>Responsible:`). This text is entered one line at time, ending with **^D**.

This text is then sent via e-mail to the PR Submitter, the Responsible developer, and <`gnats-admin@NetBSD.org`>. It is also appended to the PR by **edit-pr** along with the user ID of the developer making the change, a timestamp, and the entered text.

Unfortunately, no external editor can be invoked at this point; if you make a mistake, you'll have to use **edit-pr** to correct it.

Resolving Problem Reports

The Ideal Process

In an ideal world, the process for a problem report is as follows:

1. A NetBSD user has a problem with NetBSD. He invokes the **send-pr** command on his system (assuming it's still stable enough to do that), and files a Problem

Report. Hopefully, he follows all the advice found in "What goes into a Problem Report."

If the user's own system is not stable enough to use **send-pr**, there is a web interface that can be used to submit problem reports.

2. When the PR arrives at `gnats.NetBSD.org`, the GNATS database system examines it, and files it. If the PR is malformed, it will be filed in the `pending` category, marked confidential, awaiting manual intervention by the GNATS database administrator.

   If the format is OK, the PR is assigned a PR number, filed into the requested category, and E-mailed out again to the default responsible party for the category, and to the netbsd-bugs mailing list. A notice of the PR number and default Responsible Person is also E-mailed back to the PR Submitter.

3. The Responsible Person should read and analyze the PR. Any other person who has insight into the problem should also add whatever information they can to the problem report (this is why the report is mailed out to a mailing list; a wide audience increases the probability that a key insight needed to solve the problem will be discovered).

   If the default Responsible Person determines that another developer is a more appropriate Responsible Person, the PR should be reassigned with **edit-pr**. The new Responsible Person should read and analyze the PR.

4. Once a cause and potential fix has been identified, a description should be added to the PR, and the state should be changed to `analyzed`. At this point, implementation of the fix for the problem begins.

   This part of the process should begin as quickly as possible, since a user with a current problem is suffering, but also has his attention engaged, and his hardware available for testing potential fixes. If a PR is allowed to languish, the opportunity to reproduce the problem and test potential fixes may be lost.

5. Once the implementation of the fix is completed and communicated to the PR submitter, the PR state should be changed to `feedback`, awaiting response from the PR submitter that the fix really works.

   A PR should also be put into `feedback` state when input is required from the submitter to complete the analysis of the PR (i.e. when you ask them a question), or when you need information from some other source (essentially, `feedback` is a wait state).

6. Once it is confirmed that the problem has been fixed, it should be pulled up to the appropriate active branches. After sending the pullup request(s), you should set the PR state to `pending-pullups`, and list the pullup number(s) in the reason.

   When the pullup request(s) have not yet been sent (perhaps because they are complicated) the state should be set to `needs-pullups` to draw this fact to the attention of passersby.

7. Once the PR submitter confirms that the fix works, and all pullup requests have been issued and completed, the PR can be `closed`.

At each step of the PR handling process, make sure that feedback and other analysis and commentary is appended to the PR by using a proper E-mail subject line and making sure that the messages are copied to <`gnats-bugs@NetBSD.org`>. Having a complete record of information about the PR is valuable both while hunting down the bug and for future system maintenance.

If at all possible, it is important to get any fix committed to the CVS trunk pulled up by NetBSD Release Engineering to the "-release" branch, so that people who are tracking that branch can get the fix right away rather than waiting for the next major release of NetBSD. This also makes it possible for the next point release of NetBSD to have the fix.

The NetBSD Community is a whole lot of very smart, and very experienced people. If you're having trouble analyzing a problem report, ask questions in the appropriate

mailing list; more than likely, someone will be able to help.

Other Ways Things Get Done

The Ideal Process was described above. That's not the only way that problem reports get handled. All of the principal people involved in NetBSD are pretty busy, and can't devote full attention to this project. As such, if you see a PR that you can solve that hasn't been attended to yet, go claim it by setting yourself as the Responsible Person with **edit-pr**, and solve it.

Even if you don't feel qualified to hack the code yourself, if you can offer a test case or other information, send it along to GNATS to be appended to the PR. “Many hands make light work.”

Some problem reports are so trivial that the fix is obvious (or perhaps the fix was provided by the submitter), that they go directly from open to `closed` immediately after the fix is committed.

If for some reason you find that you're unable to finish handling the PR, reset the `>Responsible:` field to whoever had responsibility before you took the PR over. Don't prevent others from making progress on the PR because they think you're taking care of it.

As long as you're marked as the Responsible Person for a PR, you'll receive a monthly E-mail reminder about it. Use those reminders to drive you to review PRs and put them into their correct states as time passes.

When a PR is in `feedback` state, the PR submitter gets an E-mail monthly reminder at the same time as the Responsible Persons do, to prompt or prod them into responding. Generally, if there has been no response for more than three months (three reminder cycles), it's pretty safe to assume that the submitter is gone or no longer cares. At that point, whether to close the PR becomes a judgement call for the Responsible Person - how serious is it? Should it be solved without further input from the Submitter?

The other way we use the GNATS PR database is to keep track of problems which are waiting for larger issues to be solved. The oldest PR in the database at this writing, lib/13 (yes, of course it would be 13!) begs the entire internationalization of the NetBSD system. I18N is a hard problem that requires a wholesale overhaul of the system, which is why that PR is still open after seven years. This doesn't mean we'll never solve it; just that it isn't as critical as some other problems reported in the database.

In effect, this usage of the GNATS PR database is as a long-term project tracking system.

Priorities, Severities, and Releases

In an ideal world, the GNATS PR database would be empty, we'd release perfect software, and everyone would get along.

The `>Priority:` field in the PR reflects this ideal, in that `high` priority is supposed to be fixed immediately; `medium` is supposed to be resolved before the next release of NetBSD (major or minor?), and `low` priority gets solved “eventually”.

In practice, PR resolution is dependent on the right mix of submitter interest, developer interest, problem reproduceability, hardware availability, and good timing. If any of the required elements is missing from the mix, the PR will sit.

If we were really diligent about PRs, we would adjust the priority of each PR to reflect its actual importance, and probability of getting fixed according to the definitions. Unfortunately, that requires an overall evaluation of Release Engineering goals and targets and all PRs relative to each other, which is difficult for a dispersed group to do in an organized fashion.

In contrast, the `>Severity:` field is really an expression of the amount of pain the user is going through with the problem being reported, and it's something we really shouldn't adjust without careful consideration.

The proper procedure would be to review all PRs in the database at each release point,

to decide on a per-PR basis whether to “fix now”, “fix later”, “suspend” and adjust priorities. Perhaps one day we'll have the resources and manpower for that.

Remote GNATS Operations

For people who find logging into a remote host tedious, the following csh aliases might be useful:

```
alias query-pr  'ssh gnats.NetBSD.org query-pr --full \!* | tee
pr-\!*'
alias edit-pr   ssh -t gnats.NetBSD.org edit-pr
```

Back to *NetBSD Developer Documentation*

NetBSD CVS HOWTO

Introduction and Purpose

Getting Started

Using CVS

Other Useful CVS Operations

Other Sources of Information

Introduction and Purpose

What is CVS?

CVS stands for *Concurrent Versions System*. CVS is a *Source Control* or *Revision Control* tool designed to keep track of source changes made by groups of developers working on the same files, allowing them to stay in sync with each other as each individual chooses. In particular, CVS is the tool used by The NetBSD Project to manage its source tree.

It is important to note that thanks to the *concurrent* part of CVS, several people can work on things at the same time. There is no locking of files as with RCS.

Getting Started

This section shows the basic steps for installing CVS on your system.

Install CVS

You need to install CVS on your local machine. It is your machine that you will do all your work on, not the CVS server. If you don't already have CVS installed on your machine, then install CVS from the package collection .

Configure your environment for CVS

On your local machine that you plan to use CVS on, add the following to your `.cshrc` file if you use **csh** or **tcsh** as your shell.

```
setenv CVSEDITOR vi
setenv CVSROOT yourusername@cvs.NetBSD.org:/cvsroot
setenv CVS_RSH ssh
```

Add the following to your `.profile` file if you use **sh** or **bash**.

```
CVSEDITOR=vi
CVSROOT=yourusername@cvs.NetBSD.org:/cvsroot
CVS_RSH=ssh
export CVSEDITOR
export CVSROOT
export CVS_RSH
```

Make a directory for CVS work

I keep all my CVS related work in `/usr/cvs` on my machine, but this is just personal preference. In the rest of this document I will assume that you have created a `/usr/cvs` directory on your machine. Once again, this is your local machine, not the CVS server.

Using CVS

This section shows the basic steps involved with using CVS.

An Example

This section shows how to start with an empty `/usr/cvs` directory, add the `htdocs/people/` module, make a change and then commit the change. The basic steps involved are:

```
cd /usr/cvs
cvs checkout htdocs/people
cd htdocs/people
vi developers.xml
make
cvs ci developers.xml developers.html
```

A more detailed explanation of these steps follows.

```
cvs checkout htdocs/people
```

This line told CVS to check out the directory `htdocs/people` to our local machine. This means that the most recent version of the files in `htdocs/people` are copied from the CVS server to our local machine. In addition, CVS directories were created in `/usr/cvs`, `/usr/cvs/htdocs`, and `/usr/cvs/htdocs/people`. It is important to keep in mind that unlike a checkout in RCS, a CVS checkout doesn't lock the file. Ie, other developers can still access the checked out file.

Once we have checked out a directory to our local machine, we don't need to do a **cvs co**, or equivalently **cvs checkout**, again for this directory. If we want to update a previously checked out directory to the most recent version, we would simply do, for example, **`cvs update htdocs/people`**.

The next step after checking out the directory (or updating if we had already checked out the desired module before) was to edit a file. After all our edits were complete, we did:

```
  make
```

This regenerated the `developers.html` file from the `developers.xml`. We then did:

```
cvs ci developers.xml developers.html
```

This command uploaded our changes to the CVS server and updated the RCS header in the files to reflect the new version. When you checkin (commit) a change, you will be prompted to enter a short description of your changes.

It should be noted that within the source tree adding new entries and regenerating files is normally done as two commits.

CVS Modules

In CVS a module can refer to a single file, a directory with several files, or even an entire directory tree. A list of NetBSD top level CVS modules details the top level CVS modules in the NetBSD tree. A couple of other top level modules of particular interest are:

htdocs
: The NetBSD WWW pages.

othersrc
: Related code that is not part of NetBSD itself.

For each of the top level modules, you can check out (and later update) the entire top level, or you can check out (and update) lower level modules. For example, to check out the entire `htdocs` module, we would do:

```
cd /usr/cvs
cvs checkout htdocs
```

We could also only get the `htdocs/people` subdirectory with

```
cd /usr/cvs
cvs checkout htdocs/people
```

In these examples, don't forget that you only need to check out a module once. For example, if you have already checked out the `htdocs` module but want to bring it all up to date, you would do:

```
cd /usr/cvs
cvs update htdocs
```

Importing a new package

This section has a few supplements to the Importing a New Package instructions.

Here is an example of adding a newly created package. This example assumes that your CVS stuff is kept in `/usr/cvs` and `pkgsrc` is a subdirectory from there. The new package, `foo/bar` in this case, is in the directory `/usr/cvs/pkgsrc/foo/bar.work`. The steps to import this package into the source tree are:

```
cd /usr/cvs/pkgsrc/foo/bar.work

# do the usual testing, pkglint, etc.
# don't forget to do 'make makepatchsum' to generate the
# files/patch-sum file.

grep TNF /usr/pkgsrc/doc/pkgsrc.txt

# this command tells me the correct syntax for "cvs import".
# (/usr/pkgsrc is a symlink to /usr/cvs/pkgsrc).

cvs import pkgsrc/foo/bar TNF pkgsrc-base

# note it's "bar", not "bar.work"

cd ../../..
cvs checkout pkgsrc/foo/bar

# this checks out the newly imported package to
# see if everything went ok

cd pkgsrc/foo
cvs update Makefile
vi Makefile

# add bar to the pkgsrc/foo/Makefile

cvs ci -m 'add & enable bar' Makefile

# check in the changes to pkgsrc/foo/Makefile
```

```
cd /usr/cvs/pkgsrc/doc
cvs update CHANGES-YYYY

# this makes sure you have the most up to date CHANGES-YYYY

vi CHANGES-YYYY

# make a note of your changes to the foo/bar package

cvs ci -m 'Added foo/bar version 17.42' CHANGES-YYYY

# check in CHANGES-YYYY
```

Other Useful CVS Operations

CVS output

When you perform various CVS tasks, you will see a list of each file which is involved. The filename will be preceeded with a single letter code. A quick summary of the letters is here:

- **U**pdated
- **P**atched
- **C**onflict
- **M**odified (locally)
- **A**dded
- **R**emoved
- **?** (not under CVS control)

Other Sources of Information

NetBSD specific

- pkgsrc home page.
- Useful NetBSD developers information.
- Complete pkgsrc documentation.

CVS specific

- The CVS Manual
- The CVS FAQ
- Cyclic Software, the company that maintains CVS
- Open Source Development with CVS, 3rd edition, a book that's made available on the web.

NetBSD Developer Documentation: Adding a new port

The following is intended to be a brief list of items that need to be addressed when adding a new port to the NetBSD tree.

Note the new files should be added with `cvs add`, not `cvs import`. Importing is for vendor branches, whereas this code is to be integrated as part of NetBSD, and it's also one more needless tag on each file.

Steps for those just starting out

- Things to try at first

Items that need to be addressed when adding new port

- CVS setting
- htdocs changes
- GNATS configration
- Mail changes (requires special privileges)
- Create FTP directory for snapshots of new port
- Other changes

Important tasks once port is imported into the tree

- You mean there's more?

Other related documents

- Valeriy E. Ushakov's papers
- Frank van der Linden's papers

Steps for those just starting out

Things to try at first

- Try to find as much documentation as you can. Then try some more. Contact the original vendor, and the vendors of major chips used on the boards. Frequently the source for another vendor's OS (even Linux) can prove a useful resource.
- Copy the sys tree of a similar port.
- Setup a cross compiler on another NetBSD machine (something with the same CPU would be ideal, but usually an i386 should be fine).
- Work out how to generate a binary understood by the boot loader. If there is an existing OS on the target, you might be better off taking advantage of it.
- Try to build a standalone binary which can output to the console, possibly using PROM drivers, or serial console. No need for VM.
- Build a kernel with memorydisk root.
- Add disk and network device support. It may be easier to start with external cards for which MI drivers already exist in the NetBSD tree.

Items that need to be addressed when adding new port

CVS setting

Setting up cvs checkout-update-commit environment.

- Read NetBSD CVS HOWTO and setup CVS commitable environment.
- `cvs checkout {htdocs,htutils,doc}`

htdocs changes

Edit and commit information about your new port on www.NetBSD.org. Obviously, you need to create a port page under `htdocs/ports/${MACHINE}/` first. Please do not commit modifications to other htdocs files until this page has been created and is committed. The DocBook/XML format is useful for FAQ-style information (see the htdocs documentation).

- Create `htdocs/ports/${MACHINE}/`, add link from `htdocs/index.html`, `htdocs/ports/index.xml`. You may want to include a copy of this checklist and mark off items as completed. If you wish to include a logo or a graphic for your port page, please make sure that you get permission from the copyright owner (if applicable) before adding a new image to the htdocs tree.
- Add details to `htdocs/changes/index.xml` and regenerate the HTML file with make(1). See htdocs documentation for details.
- Add to tables in
  - `htdocs/about/query-pr.xml` and regenerate the HTML file with make(1)
  - `htutils/cgi-src/gnats/netbsd.def`
  - `htutils/changes/changes2html`
  - `htutils/changes/code-changes2rss`
  - `htutils/changes/cvschanges2html`
- Add to `htdocs/developers/features/table` and regenerate.
- Add your entry to `htdocs/people/port-maintainers.xml`.

The list of modified files should be committed in one go. Please contact <www@NetBSD.org> if you have any questions and remember to announce the new port on other websites once the new html files are online.

GNATS configration

- Add new category to `src/gnu/usr.bin/send-pr/categories`
- Send mail to <admins@NetBSD.org> about creating the new category on the GNATS server.

Mail changes (requires special privileges)

- Create `port-${MACHINE}` mailing list. Send a request to <admins@NetBSD.org> over it.
- Add new `port-${MACHINE}` list to `htdocs/mailinglists/index.xml`. Regenerate with make(1).
- Create `port-${MACHINE}-maintainer` alias for the portmaster.
- Add portmaster to the port-masters mailing list.
- Add `port-${MACHINE}-maintainer` to bug-managers mailing list. e.g. add port-hpcmips-maintainer to bug-managers.
- If the port uses a shared `MACHINE_ARCH`, add `port-${MACHINE}-maintainer` to the `port-${MACHINE_ARCH}-maintainer` mailing list, e.g. add port-hpcmips-maintainer to port-mips-maintainer.

Create FTP directory for snapshots of new port

- Update port machinery in `distrib/` to build snapshots and releases. Did you create `src/distrib/${MACHINE}` ?
- Make snapshot. See NetBSD Developer Documentation: Release Engineering
- Make source tarballs. The `src/distrib/sets/makesrctars` script will help you.
- Create `ftp.NetBSD.org:/pub/NetBSD/arch/${MACHINE}/snapshot/YYYYMMDD/` and put the snapshot there.

Other changes

- Add an entry to `src/doc/CHANGES`.
- Add an entry to `src/build.sh`.
- Regenerate `src/tools/headerlist` by running `src/tools/mkheaderlist.sh`.
- Add an entry to `src/gnu/usr.bin/groff/tmac/mdoc.local`.
- Add an entry to `src/distrib/notes/common/main`.
- Add an entry to `src/share/man/man7/hier.7`.
- Add an entry to `src/sys/arch/README`.
- Add an entry to `src/sys/arch/Makefile`.
- Add an entry to `src/usr.bin/calendar/calendars/calendar.netbsd`.
- Create sup(1) set for new port (current and release collection for `src/sys/arch/${MACHINE}/`, and mirror collection for `ftp.NetBSD.org:/pub/NetBSD/arch/${MACHINE}/`). Consult other developers.
- Update tarball-building scripts to build tarball for new `${MACHINE}`. Consult other developers.
- Send email to <announce@NetBSD.org> and note the announcement on the WWW

news page.

Important tasks once port is imported into the tree

You mean there's more?

- Keep adding hardware support.
- Get sysinst running.
- Produce formal releases.
- Write a `compat_<original_os>`.
- Ongoing maintenance. Make sure your port continues to build and run as the NetBSD source tree evolves.

Other related documents

Valeriy E. Ushakov's papers

Valeriy a related paper on how to get started on a new port.

Frank van der Linden's papers

Frank van der Linden's paper Porting NetBSD to the AMD x86-64: a case study in OS portability from BSDCon 2002.

Back to *NetBSD Developer Documentation*

NetBSD Port/Feature Cross Reference

	Latest Release	shlibs	SSP	gdb	sysinst	X server	wscons	bus space	bus dma	evcnt	cpu_in_cksum	SMP	RAS	topd vi
aarch64	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	Y	N	Y	Y	Y
acorn32	8.1	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	N	Y	spinup	Y	Y
algor	9.2	Y	N	8.0.1	N/A	N/A	Y	Y	Y	Y	N	N/A	Y	Y
alpha	9.2	Y	N	8.0.1	Y	X.Org	Y	Y	Y	Y	N	Y	Y	Y
amd64	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	Y	Y	Y	Y
amiga	9.2	Y	Y	8.0.1	Y	X.Org	N	Partial	N	N	Y	N/A	Y	N
amigappc	9.2	Y	Y	8.0.1	N/A	N	N	N	N	N	N	N/A	Y	Y
arc	9.2	Y	N	8.0.1	Y	N	Y	Y	Partial	Y	N	N	Y	Y
atari	9.2	Y	Y	8.0.1	Y	N	N	Y	Y	N	Y	N/A	Y	N
bebox	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	N	N	Y	Y
cats	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	Y	Y	N/A	Y	Y
cesfic	9.2	Y	Y	8.0.1	N/A	N/A	N/A	N	N	N	Y	NFW	Y	N
cobalt	9.2	Y	N	8.0.1	Y	N/A	Y	Partial	Y	Y	N	N/A	Y	Y
dreamcast	9.2	Y	Y	8.0.1	N/A	X.Org	Y	Partial	Partial	Y	Y	N/A	Y	N
evbarm	9.2	Y	Y	8.0.1	Y	N/A	Y	Partial	Y	Y	Y	Y	Y	Y
evbmips	9.2	Y	N	8.0.1	Y	N/A	N	Y	Y	Y	N	N/A	Y	Y
evbppc	9.2	Y	Y	8.0.1	Y	N	Y	Y	Y	Y	N	N/A	Y	Y
evbsh3	9.2	Y	Y	8.0.1	N/A	N/A	N	Partial	N	Y	Y	N/A	Y	N
ews4800mips	9.2	Y	N	8.0.1	Y	X.Org	Y	Partial	Y	Y	N	N	Y	Y
hp300	9.2	Y	Y	8.0.1	Y	N	Y	Partial	N	Y	Y	N/A	Y	N
hppa	9.2	Y	N	8.0.1	Y	N	N	Partial	Y	Y	N	N	Y	Y
hpcarm	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	N	Y	N/A	Y	Y
hpcmips	9.2	Y	N	8.0.1	Y	X.Org	Y	Partial	Y	Y	N	N/A	Y	Y
hpcsh	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	Y	Y	N/A	Y	N
i386	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	Y	Y	Y	Y
ia64	9.2	Y	N	N	N	N	N	Y	N	Y	N	N/A	N	N
ibmnws	9.2	Y	Y	8.0.1	N/A	N	Partial	Y	Y	Y	N	N/A	Y	Y
iyonix	9.2	Y	Y	8.0.1	N/A	N	Y	Partial	Y	Y	Y	N/A	Y	Y
landisk	9.2	Y	Y	8.0.1	Y	N/A	Y	Y	Y	N	Y	N/A	Y	N
luna68k	9.2	Y	Y	8.0.1	N/A	N	Y	Partial	N	N	Y	N/A	Y	N
mac68k	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	N	Y	N/A	Y	N
macppc	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	N	Y	Y	Y
mipsco	9.2	Y	N	8.0.1	Y	N	N	Y	Y	Y	N	N/A	Y	Y
mmeye	9.2	Y	Y	8.0.1	N/A	N/A	N	Partial	N	Y	Y	N/A	Y	N
mvme68k	9.2	Y	Y	8.0.1	Y	N/A	N/A	Partial	Y	N	Y	N/A	Y	N
mvmeppc	9.2	Y	Y	8.0.1	N/A	N/A	N/A	Y	Y	Y	N	N	Y	Y
netwinder	9.2	Y	Y	8.0.1	N/A	X.Org	Y	Partial	Y	Y	Y	N/A	Y	Y
news68k	9.2	Y	Y	8.0.1	Y	N	N	Partial	N	N	Y	N/A	Y	N
newsmips	9.2	Y	N	8.0.1	Y	X.Org	Y	Partial	Y	N	N	N/A	Y	Y
next68k	9.2	Y	Y	8.0.1	N/A	N	Y	Y	Y	N	Y	N/A	Y	N
ofppc	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	N	N	Y	Y
pmax	9.2	Y	N	8.0.1	Y	X.Org	Y	Partial	Y	Y	N	NFW	Y	Y
prep	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	N	N	Y	Y
rs6000	9.2	Y	Y	8.0.1	N/A	N	N	Y	N	Y	N	N	Y	Y
sandpoint	9.2	Y	Y	8.0.1	Y	N/A	N	Y	Y	Y	N	N/A	Y	Y
sbmips	9.2	Y	N	8.0.1	N/A	N/A	N	N	N	Y	N	N	Y	Y
sgimips	9.2	Y	N	8.0.1	Y	X.Org	Y	Partial	Y	Y	N	N	Y	Y
shark	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	N	Y	N/A	Y	Y
sparc	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	N	Y	Y	Y	Y
sparc64	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	N	Y	Y	Y
sun2	9.2	N	Y	8.0.1	N	N	N	Y	Y	N	Y	N/A	Y	N
sun3	9.2	Y	Y	8.0.1	N	X.Org	N	Y	Y	N	Y	N/A	Y	N
	Latest Release	shlibs	SSP	gdb	sysinst	X server	wscons	bus space	bus dma	evcnt	cpu_in_cksum	SMP	RAS	topd vi

	Latest Release	shlibs	SSP	gdb	sysinst	X server	wscons	bus space	bus dma	evcnt	cpu_in_cksum	SMP	RAS	topd vi
vax	9.2	Y	Y	8.0.1	Y	N	Y	Y	Y	Y	Y	Y	Y	Y
x68k	9.2	Y	Y	8.0.1	Y	X.Org	N	Partial	Y	N	Y	N/A	Y	N
xen	9.2	Y	Y	8.0.1	Y	X.Org	Y	Y	Y	Y	N	N	Y	Y
zaurus	9.2	Y	Y	8.0.1	Y	X.Org	Y	Partial	Y	Y	Y	N/A	Y	Y
	Latest Release	shlibs	SSP	gdb	sysinst	X server	wscons	bus space	bus dma	evcnt	cpu_in_cksum	SMP	RAS	topd vi

Description of features

Latest Release
What is the latest formal release for which this port was built?

shlibs
Does the port support shared libraries and dynamic linking?

SSP
Does the port have SSP (Stack Smashing Protection) support?

gdb
Which gdb version is used for the port, if any (gdb-6.x, gdb-7.x or none)?

sysinst
Does the port use sysinst for installation? "no" means that install or diagnostic boot media exists, but sysinst is not on the media or is not the default install method.

X server
Does the port provide an X11 server? "X.Org", "XF4.x", "no", or "N/A".

wscons
Does the port use the MI WSCONS console driver system, rather than a homegrown console interface? (Y, partial, or N)

bus space
Does the port provide the bus_space interfaces in machine/bus.h? Support is considered complete ("yes") even if some word size operations (e.g. the 8-byte operations) are not provided. If some support is provided but some sets of operations (e.g. bus_space_read_region_N) have been omitted, support is considered "Partial".

bus dma
Does the port provide the bus_dma interfaces in machine/bus.h?

evcnt
Does the port use the evcnt(9) generic event counter framework?

cpu_in_cksum
Does the port provide an optimized cpu_in_cksum?

SMP
Does the port support multiple processors? "yes" means yes, "Y-dev" means yes, but not production stable, i.e. Yes for developers. "spinup" means initial support (the additional processors are started up), "no" stands for not yet, "NFW" means that there exist MP-machines, but at this time there are no plans to support them, and "N/A" means that there are no MP-machines of that type.

RAS
Does the port provide kernel support for user-level restartable atomic sequences (RAS)? See ras(9) for implementation details.

topdown vm
Does the port offer or use topdown mmap ordering? "yes" means yes, "opt" means it is offered as an option (presumably to be upgraded to yes at some point), "no" means no.

rf boot
Can the port boot from a RAIDframe mirror?

ffsv2 boot
Can the port boot from an FFSv2 root partition?

fast softint
Does the port provide the fast software interrupt mechanism (and therefore define __HAVE_FAST_SOFTINTS)?

kernel preempt
Does the port provide kernel preemption? (and therefore define __HAVE_PREEMPTION)?

anita
Is the port supported by Anita (Automated NetBSD Installation and Test Application)? In other words, is it possible to do automated install/test runs in a virtual machine.

modules
Does the port work with new style kernel modules. "Y" means kernel modules work and bootloader also has support for loading bootstrap modules to mount the root file system. "kernel" means kernel modules work but no bootloader support. "N" means no.

Features supported by all ports

ELF
Can the port use ELF as its native object file format? "yes" if ELF is the default, or "no" if ELF is not supported as the native object file format.
All: Y

gcc
Which compiler does the port use (gcc-4.5.x, gcc-4.8.x, or other)?

All: 6.4

binutils

Which gas and ld does the port use? "2.19.1" means 2.19.1, and "other" means a version that's not in the CVS tree.
All: 2.27

 Home Page

 Developer Documentation

---

tsarna@NetBSD.org
(Contact us) *$NetBSD: index.html,v 1.795 2021/05/17 12:47:46 martin Exp $*

nono 0.2.1 (2021/08/18)

nono は NetBSD とかで動作する OMRON LUNA-I や LUNA88K とかのエミュレータです。 [nono is OMRON LUNA-I and LUNA88K emulator runs on NetBSD and etc.]

Index of this page:

ビルド方法 [How to build]

ビルドには以下が必要です。 NetBSD(pkgsrc) なら wxWidgets は pkgsrc/x11/wxGTK30 です (OPTIONS:gtk3 で動作確認しています)。 [The followings are required for build. If you use NetBSD(pkgsrc), wxWidgets is pkgsrc/x11/wxGTK30 (OPTIONS:gtk3 is tested).]

- make (BSD make, not GNU make)
- C/C++ compiler which supports -std=c++14.
  (For gcc, 7.4 works but 5.5 doesn't work, at least. For clang, 7.0 and 8.0 works at least.)
- wxWidgets >= 3.0
- gettext

(NetBSD 以外でのビルドはサポートしていませんが) Ubuntu 20.04 ではたぶん以下のパッケージが必要です。 [You may need the following packages on Ubuntu 18.04 (though we won't support non-NetBSD platform).]

- bmake build-essential gettext libbsd-dev libedit-dev libkqueue-dev libwxgtk3.0-gtk-dev zlib1g-dev

nono のソースアーカイブを展開したら以下のようにビルドします。 configure には環境変数 CC、CXX でコンパイラを指定することが出来ます。また wx-config が標準的な名前で提供されていないために見付けられない場合には環境変数 WX_CONFIG にパスを指定することが出来ます。 [Extract the nono's source archive and build as following. You can specify C/C++ compiler using environment variable CC and CXX if configure cannot find standard name suitable compiler. Also, you can specify wx-config path using environment variable WX_CONFIG if configure cannot find wx-config.]

```
% ./configure
% make depend
% make
% su
# make install
```

2つの実行ファイルがインストールされます。 `nono` が GUI 版実行ファイル、 `nono-cli` がコマンドライン版です。 [Two executables will be installed. `nono` is the GUI executable and `nono-cli` is the command line executable.]

コマンドラインオプション [Command Line Option]

`-c` *vmpath*
: VM ディレクトリ/設定ファイルを指定します。 *vmpath* がディレクトリならそのディレクトリの中の nono.cfg を設定ファイルとします。 *vmpath* がファイルならそれを設定ファイルとします。そしていずれの場合も設定ファイルがあるディレクトリを VM ディレクトリとします。 -c オプションを省略すると *vmpath* をカレントディレクトリとします。 [Specifies the VM directory/configuration file. If *vmpath* is a directory, make nono.cfg in that directory a configuration file. Or if *vmpath* is a file, make the specified file a configuration file. And in both cases, make the directory where that file is located a VM directory. If `-c` option is omitted, *vmpath* is considered as the current directory.]

`-f`
: 高速モードで起動します。 GUI なら起動後にもメニューから変更できますが、その初期値を変えるだけです。 [Boot as the fast mode. You can change this mode on GUI menu after boot, and the option only changes its initial state.]

`--fontsize` *height*
: GUI 版のみ。全サブウインドウの起動時のフォントサイズを指定します。 *height* には 12, 16 のいずれかを指定します。デフォルトは `12` です。起動後にメニューから変更できます。 [GUI Only. Specifies the initial fontsize on all sub windows. *height* is one of 12 or 16. The default value is `12`. You can change this value on GUI menu after boot.]

`-s` *scale*
`--scale` *scale*
: GUI 版のみ。メインウィンドウの起動時のスケールを実数で指定します。起動後にもメニューからプリセットされた倍率には変更可能ですが、任意倍率は起動時のみ指定可能です (そのうちなんとかしたい)。デフォルトは 1.0 です。 [GUI Only. Specifies the initial main window scale in real number. You can change this scale on GUI menu after boot, but unlike this option, there are only a few preset choices (Should be improved in someday). The default value is 1.0]

`--show-config`
: 設定ファイルと `-v` オプションを読み込んだ結果を表示します。 [Shows the result of reading configuration file and

parsing -v options.]

-v
バージョンを表示します。 [Shows the version.]

-V *name=value*
設定ファイルで指定した *name=configvalue* の代わりにこのオプションの *name=value* を適用します。 [Use this *name=value* instead of *name=configvalue* specified in configuration file.]

-X *file*
ホストの *file* をロードして実行します。 *file* が相対パスの場合カレントディレクトリからのパスになります。ファイルが gzip 圧縮されていれば自動的に展開します。 (展開後の) ファイル形式は a.out (OMAGIC) か ELF で、実際にはブートローダとカーネル程度しか想定していません。設定ファイルの prom-image とともに指定されると -X のほうが優先します。 [Loads and executes host's *file*. If *file* is relative path, it is path from the current directory. If the file is gzip'd, it is automatically extracted. The supported file format (after extracting) is a.out (OMAGIC) or ELF. Actually, it only assumes bootloaders or kernels. If this option is specified at the same time as prom-image in configuration file, this option preceeds.]

以下開発用。[For developers:]

-b *hexaddr*[,*skipcount*]
デバッガのブレークポイントを 16進数で指定します。
-C
ログをコンソールにも出力します。通常はログウィンドウにだけ出力されます。
-d
起動時にデバッガプロンプトで停止します。
-D
コンソールをデバッガとして使用します。 -d を指定しなくても起動時にプロンプトで停止します (そのうちなんとかしたい)。
-L *name1=level1*[,*name2=level2*,...]
ログレベルを指定します。カンマで区切って複数指定することも出来ます。 -Lhelp で name の一覧を表示します。
-M *name*[,*name2*,...]
起動時に表示するモニタウィンドウを指定します。カンマで区切って複数指定することも出来ます。 -Mhelp で name の一覧を表示します。

設定 [Configuration]

VM の設定ファイルはその VM ディレクトリ内の nono.cfg (または -c で指定したファイル) です。書式は key = value 形式で1行1項目ずつです。 key と value の前後の空白は取り除かれます。また空行と "#" で始まる行は無視します。知らないキーの行も無視します。 [VM configuration file is nono.cfg (or the file specified by -c option) in the VM directory. Its syntax is key = value format, one per line. White spaces before and after key and value are ignored. And, blank lines, lines beginning with "#", and lines with unrecognized key are also ignored.]

設定項目は次の通りです。 [The configuration items are:]

vmtype = *string*
VM 種別を以下のいずれかから指定します。省略不可です。 [Specifies the VM type from the following. This field is mandatory.]
luna ... LUNA-I
luna88k... LUNA88K

clock-sync = *value*
仮想マシン内の時刻の同期方法を指定します。 real なら実時間に同期、virtual なら仮想時間に同期します。デフォルトは real です。この機能は実験中のため将来予告なく仕様が変更になる可能性があります。 [Specify how to synchronize the time in virtual machine. If real, synchronize with the real time; if virtual, synchronize with the virtual time. The default is real. This feature is under experimentation and may be changed in the future without notice.]

debugger-port = *integer*
デバッガの TCP 待ち受けポート番号を指定します。 0 なら待ち受けを行いません。デフォルトは 0 です。 [Specifies the TCP port number that debugger listens. If 0, it will not listen. The default is 0.]

ethernet-macaddr = *string*
イーサネットデバイスの仮想マシン側の MAC アドレスを指定します。 *xx*:*xx*:*xx*:*xx*:*xx*:*xx* 形式で指定します。 auto なら自動的に決定します。デフォルトは auto です。

hostnet-driver = *string*
イーサネットデバイスのホスト側ドライバを指定します。 afpacket、bpf、tap のうちホスト OS がサポートしているものと auto、none が選択可能です。 afpacket は Linux の AF_PACKET ソケットを使用します。 bpf は bpf(4) デバイスを使用します。 tap は tap(4) インタフェースを使用します。 none ならホスト側とは一切通信を行いません。 auto ならホスト OS がサポートしているもののうち tap → afpacket → bpf を順に試します。デフォルトは auto です。 [Specifies the host driver of ethernet device. afpacket, bpf, and tap can be specified only if the host OS supports them. auto and none can always be specified. afpacket uses Linux's AF_PACKET socket, bpf uses bpf(4) device, and tap uses tap(4) interface. none doesn't make any communication with the host. If auto is specified, it tries tap, afpacket, and bpf in that order (if the host OS supports them). The default is auto.]

hostdriver-afpacket-ifname = *ifname*
ホストドライバが afpacket の時にバインドするインタフェースを1つ指定します。 *ifname* が auto なら使用可能なインタフェースを1つ自動的に選択します。デフォルトは auto です。なお、このホストドライバではローカルホストとの通信は出来ま

せん。 [Specify an interface name to bind to, when the host driver is afpacket. If auto is specified as *ifname*, it selects an usable interface automatically. The default is auto. Note that this host driver cannot communicate with the localhost.]

hostdriver-bpf-ifname = *ifname*
ホストドライバが bpf の時にバインドするインタフェースを1つ指定します。 *ifname* が auto なら使用可能なインタフェースを1つ自動的に選択します。デフォルトは auto です。なお、このホストドライバではローカルホストとの通信は出来ません。 [Specify an interface name to bind to, when the host driver is bpf. If auto is specified as *ifname*, it selects an usable interface automatically. The default is auto. Note that this host driver cannot communicate with the localhost.]

hostdriver-tap-devpath = *path*
ホストドライバが tap の時に使用するデバイスを1つフルパスで指定します。デフォルトは auto で、この場合は自動的にデバイスを選択します。この時の探し方はホスト OS によって異なります。 Linux なら /dev/net/tun を指定したのと同じです。 OpenBSD なら /dev/tap0 から /dev/tap9 まで順番にオープンできるまで試します。 NetBSD (と FreeBSD) ならまず /dev/tap でクローニングを試み、それが失敗すれば /dev/tap0 から /dev/tap9 までを順番にオープンできるまで試します。いずれの場合も VM ディレクトリかその親ディレクトリに nono-ifup, nono-ifdown という名前のシェルスクリプトが必要です。何もすることがない場合でも正常終了する空のシェルスクリプトを用意してください。 [Specify a device pathname, when the host driver is tap. The default is auto. The behavior in this case depends on the host OS. On Linux, it's the same as /dev/net/tun. On OpenBSD, it will try from /dev/tap0 to /dev/tap9 until successful. On NetBSD (and FreeBSD), it will try cloning by /dev/tap first. If that fails, then try from /dev/tap0 to /dev/tap9 until successful. In all cases, you need to prepare two shell script files which names are nono-ifup and nono-ifdown in the VM directory or its parent directory. Even if you don't have anything to do in these scripts, you need to prepare empty scripts that will terminate successfully.]

hostnet-fallback = *integer*
hostnet-driver で指定したドライバが使用可能でなかった時、 0 ならプロセスを終了します。 1 なら none を選択して実行を継続します。デフォルトは 0 です。 [Specify the behavior when the driver which is selected by hostnet-driver is unusable; terminate the process if 0, or continue to run using the none driver if 1. The default is 0.]

luna-adjust-misused-epoch = *integer*
LUNA で誤った RTC epoch を採用している OS 向けに RTC エミュレーションを補正するかどうかを指定します。0 なら補正をしません(実機と同じ動作)、 1 なら補正します(現実世界と同じ動作)。デフォルトは 1 で、通常 1 のままで使用して問題ありません。 [Specifies whether nono corrects RTC emulation for OSes that adopts wrong RTC epoch on LUNA. 0 means making no correction (this is the same behavior as the actual machine). 1 means making correction (this is the same behavior as the real world). The default is 1. Normally, leave it 1.]

LUNA で採用している RTC (MK48T02) は2桁で保持している年の値が 4 で割り切れる年をうるう年とする仕様です。ところが NetBSD/luna68k、OpenBSD/luna88k など現存してソースが確認できる OS はいずれもこの RTC の年の値を 1970年からの経過年として使用しています (LUNA が本来どういう仕様だったのかは分かりませんが、異なる OS を起動し直すたびに RTC epoch をずらす運用は考えにくいのでおそらくすべての OS で同じだと思います)。例えば1970年はうるう年ではないため 2月28日の翌日は 3月1日ですが、 MK48T02 的には 00年であるためうるう年と認識し 2月28日の翌日が2月29日になります。このように、実機の RTC は4年のうち約2年間、1日ずれた日付を指しているようです。しかしながら、RTC の時刻は OS 起動時に一度読んだ後は基本的に参照しない上、今時必要なら NTP で時間を合わせるため、実機でも問題が顕在化することはまずないと思います。 nono の場合は実機と異なり、アプリケーション実行中しか RTC が進まないため、補正がない場合の動作が問題になるのは nono を起動したまま偶数年の2月末日から日付をまたいで、かつ nono を起動したまま OS を再起動して NTP などで時刻修正を行わなかった時だけだと思います。このオプションはほぼ開発者向けの動作確認用です。

luna-dipsw1 = *string*
本体前面 DIPSW#1-1..#1-8 の内容を指定します。 "0" を DOWN、"1" を UP として、これを8つ並べた形式で、前から順に #1..#8 に対応します。 [Specifies status of the front panel DIPSW#1-1..#1-8 using 8 digit. "0" means DOWN and "1" means UP. The first character corresponds to #1 and the eighth character corresponds to #8.]

LUNA-I でのデフォルトは 11110111 です。各スイッチの内容は以下のリンクを参照してください。 [On LUNA-I, the default value is 11110111. See the following link about DIPSW.]
→ NetBSD/luna68k: Information
LUNA88K でのデフォルトは 11111111 です。各スイッチの内容は以下のリンクを参照してください。 [On LUNA88K, the default value is 11111111. See the following link about DIPSW.]
→ OpenBSD manual pages: boot_luna88k(8)

luna-dipsw2 = *string*
本体前面 DIPSW#2-1..#2-8 の内容を指定します。書式は luna-dipsw1 と同じです。デフォルトは 11111111 です。 [Specifies status of the front panel DIPSW#2-1..#2-8. The same syntax as luna-dipsw1 is used. The default value is 11111111.]

NetBSD/luna68k のブートローダは、 DIPSW#2 が "11111111" なら自動的にカーネルをロードして実行し、どれかでも "0" にするとプロンプトで停止するようです。 (本当は #8 だけで制御するつもりだったんじゃないかという気がします)
[NetBSD/luna68k bootloader will automatically load and execute the kernel, if the DIPSW#2 is "11111111". Otherwise, the bootloader will enter interactive mode. (I doubt that they actually wanted to switch with only #8)
]

monitor-rate = *integer*
テキスト系モニタウィンドウの更新頻度を Hz 単位で指定します。 1 から 60 までの間で指定でき、デフォルトは 20Hz です。起動後にメニューからプリセットされた頻度には変更可能です。 [Specifies refresh rate of all text monitor windows in Hz. It ranges from 1 to 60. The default is 20Hz. You can change this value on GUI menu after boot, but unlike this configuration value, there are only a few preset choices.]

mpu-clock = *value*

MPU のクロック数を MHz 単位で指定します。デフォルトは LUNA-I なら 20MHz、LUNA88K なら 25MHz です。 [Specifies the MPU clock in MHz. The default value is 20MHz on LUNA-I, or 25MHz on LUNA88K.]

`mpu-pseudo-stop = integer`

m88100 にて疑似 STOP 状態を有効にするかどうかを指定します。 0 なら無効(実機と同じ動作)、1 なら有効で、デフォルトは 1 です。 m88100 には、m68k の STOP 命令 (割り込みが上がるまで何もせず待つ) に相当する命令がなく、大抵ビジーウェイトループで割り込みが上がるのを待つことになります。これは実機では (消費電力を減らす手段がないという些細な問題以外には) 何のデメリットもないのですが、エミュレータで特に高速動作させている時には割り込みが上がるまで (例えば人間がキーを入力するまで) ホスト CPU パワーを使い潰してビジーウェイトループを実行し続けることになり、ホスト CPU があっつあつになります。それを防ぐための機能です。特徴的な命令列を検出して実現しているので、すべての状況で動作するわけではありません。

`prom-image = path`

LUNA-I/LUNA88K の外部 ROM イメージファイルのパスを指定します。 *path* がファイル名のみなら VM ディレクトリとその親ディレクトリからこのファイル名を検索します。 *path* が相対パスなら VM ディレクトリからの相対パスになります (現在のディレクトリからではありません)。空にすると内蔵 ROM を使用します。デフォルトは空です。 [Specifies the LUNA-I/LUNA88K's external ROM image file path. If the *path* does not have any path delimiters, the VM directory and then its parent directory will be searched. If the *path* is a relative path, it will be path from the VM directory, not from the current directory. If the *path* is empty, internal emulated ROM will be used. The default value is empty.]

実機を持っていない場合はこの値を空に (= デフォルトのままに) しておくと、 nono 内蔵のなんちゃって下位互換 ROM で起動します。 [If you does not have the real LUNA machines, you can boot with nono's internal downward compatible emulated ROM if you set this field empty (or leave it as the default).]

LUNA-I 実機を持っている場合は ROM ファイルを指定することで実機 ROM で起動できます。 ROM ファイルは実機の 0x41000000-0x4101ffff (128KB) を保存したものです。今のところ ROM は V4.22 (Thu Jul 27 11:45:42 1989) のみサポートしています。それ以外については何も分かりません。 [If you have the real LUNA-I machine, you can boot with the real ROM spcifying the ROM file path. The ROM file is extracted from 0x41000000-0x4101ffff (128KB) of the real LUNA-I machine. For now, only V4.22 (Thu Jul 27 11:45:42 1989) is supported. I have no idea about other ROMs.]

LUNA88K 実機の場合は 0x41000000-0x4103ffff (256KB) を保存したものです。今のところ ROM は version 1.20 のみサポートしています。 [For LUNA88K, the ROM file is extracted from 0x41000000-0x4103ffff (256KB). For now, only version 1.20 is supported.]

`ram-size = integer`

搭載する RAM サイズを MB 単位で指定します。 LUNA-I のデフォルトは 16MB です。 16MB 未満は 4MB 単位で、 16MB 以上は 255MB まで 1MB 単位で指定できます (ちなみに NetBSD/luna68k の起動には最低でも 8MB 必要です)。 LUNA88K のデフォルトは 64MB です。 64MB 未満は 16MB 単位で、 64MB 以上は暫定で 255MB まで 1MB 単位で指定できます。 [Specifies the RAM size in MB. On LUNA-I, the default is 16MB. If the size is less than 16MB, you can specify in 4MB unit. If larger, you can specify up to 255MB in 1MB unit. By the way, NetBSD/luna68k needs at least 8MB to boot. On LUNA88K, the default is 64MB. If the size is less than 64MB, you can specify in 16MB unit. If larger, you can specify up to tentative 255MB in 1MB unit.]

`show-statuspanel = integer`

ステータスパネルを表示するかどうかを指定します。 0 なら非表示、1 なら表示です。起動後はメニューから変更可能です。 [Specifies whether to display the status panel or not. If 0, it is hidden; if 1, it is shown. You can change it on GUI menu after boot.]

`spc0-id`*N*`-image = devtype[,path]`

SCSI デバイスとイメージを指定します。キーの *N* には 0 から 7 が入ります。ただし ID 7 は本体が使用しますので指定しないでください。値はデバイス種別 *devtype* とディスクイメージパス *path* を ","(カンマ) で区切って並べた形式です。デバイス種別 *devtype* は以下のいずれかです。 [Specifies SCSI device and image. *N* in the key is 0 to 7. But don't specify ID 7 because the host uses it. The value is in a form of device type *devtype* and the disk image path *path* separated by ","(comma). *devtype* can be one of the following:]

- `hd` ... HD drive
- `cd` ... CD-ROM drive
- `mo` ... MO drive

devtype が `hd` なら *path* は省略できません。 *devtype* が `cd` か `mo` なら *path* は省略可能です。イメージパスが相対パスなら VM ディレクトリからの相対パスになります。 [If *devtype* is `hd`, *path* cannot be ommitted. If *devtype* is `cd` or `mo`, *path* can be ommitted. If the *path* is relative path, it is from the VM directory.]

例えば、nono.cfg と同じディレクトリに置いた sd0.img を起動 HDD ディスクイメージとして使い (LUNA では通常 ID 6 をプライマリ HDD に割り当てます)、 ID 5 に同じディレクトリの install.iso をセットした CD ドライブを、 ID 4 に起動時メディアなしの MO ドライブを接続する場合は次のようになります。 [For example, if you use a harddisk image sd0.img placed in the same directory as nono.cfg (LUNA usually assigns ID 6 to the primary HDD), ID 5 for CD-ROM drive that loads install.iso in the same directory, and ID 4 for MO drive without media on boot, write as following:]

```
spc0-id6-image = hd,sd0.img
spc0-id5-image = cd,install.iso
spc0-id4-image = mo
```

`spc0-id`*N*`-seektime = integer`

指定の SCSI HDD の平均シークタイムを msec 単位で指定します。現在のデフォルトは `0` です (S・S・D!! S・S・D!!)。 16 程度を指定すると幾分往時に思いを馳せることが出来るかもしれませんが、今の所あまり安定していません。 [Specifies the average seek time of specified SCSI HDD in msec. Currently, the default value is `0` (This may be something like

SSD :-). If you specify about 16 or so, you can feel nostalgic, but this feature is still unstable.]

`spc0-id`*N*`-writeignore = integer`
: 指定の SCSI HD デバイスへの書き込みを無視するかどうか指定します。 `0` なら通常動作(書き込みを行う)です。 `1` なら書き込みコマンドは成功したように振る舞いますが実際にはディスクイメージに一切書き戻しません。 fsck を気にせずカーネルのデバッグとかを行いたい場合にはどうぞ。何が起きるか意味が分からない人は指定しないでください。デフォルトは `0` です。 [Specifies whether nono ignores writing to SCSI HD devices. `0` means normal operation (writes to the devices). If `1` is specified, nono will not actually write back to the disk image even though the write command is succeeded. nono's SCSI devices acts as write command is successfully done but it never writes back to the actual disk image. This is useful for kernel debugging because it does not require fsck after the kernel hangs. But don't use this flag if you don't understand this paragraph. The default value is `0`.]

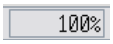
  ちなみに、メディアを書き込み禁止にしたい場合はこれではなく、イメージファイルの書き込み権を落としてください。 [By the way, if you want to make the media write-protected, clear the write permission from the image file (instead of this setting).]

`spc0-id`*N*`-writeprotect = integer`
: 古いオプションです。代わりに `spc0-id`*N*`-writeignore` を使ってください。このオプションは開発用です。 [Obsolete. Use `spc0-id`*N*`-writeignore` instead. This option is for developers.]

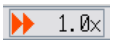
VM について [About VM]

ステータスパネル [Status Panel]

ステータスパネル中央にあるパフォーマンスメータのアイコンは高速モードの状態を表示しています。ダブルクリックすると高速/等速モードの指定を切り替えることができます。 [The performance meter's icon at the center of the status panel shows the VM speed status. You can switch full speed / syncronized mode by double-clicking on this icon.]



... マークなしの場合、ユーザが等速モードを指定していて、等速モードで実行中です。 [When no icons are displayed, the user has specified syncronized mode and the VM is running in synchronized mode.]



... 三角2つの場合、ユーザが高速モードを指定していて、等速モードで実行中です。キー入力中(後述)または VM がアイドル状態(m68k の STOP 命令)などで起きます。高速モード中はパーセントではなく何倍速で動作しているかを表します。 [When an icon (two triangles) is displayed, the user has specified full speed mode and the VM is running in synchronized mode. This will happen during keystrokes (see below) or when the VM is idle (STOP instruction on m68k).]

... 三角3つの場合、ユーザが高速モードを指定していて、高速モードで実行中です。 [When an icon (three triangles) is displayed, the user has specified full speed mode and the VM is running in full speed mode.]

キー入力 [Key input]

LUNA では、キーボードがハードウェア側でキーリピートを行わず、ソフトウェア(OS など)がキーリピートの処理を行います。そのため、キーリピートを実装していない LUNA の PROM や NetBSD/luna68k のブートローダではキーリピートは起きず、キーリピートを実装している NetBSD/luna68k カーネルではキーリピートが起こせます。キーリピートを起こす間隔をゲストのソフトウェアが測定しているということは、 VM が高速動作するとキーリピートもそれに合わせて発生することになり、これをホスト側から防ぐ手段はありません。そこで nono ではキー入力が発生している間 (キーが一つでも押されてからキーが全て離されるまでの間)、高速モードが指示されていても VM を一時的に等速モードに落として実行します。上記のアイコンがそれを区別しているのはこのためです。そのため、何らかの理由でキーが入りっぱなしになった場合 (ALT + TAB やアクセラレータキーでメニューを開くと起きがちです) 高速モードが抑制されたままということが起きえます。その場合はソフトウェアキーボードからキー入力を解除するなどしてください。 [On LUNA, key repeat is done by software(OS), not by the keyboard hardware. For this reason, key repeat doesn't occur on LUNA's PROM or NetBSD/luna68k's bootloader that don't implement it, and key repeat occurs on NetBSD/luna68k kernel that implements it. Since the timing of key repeat is measured by the guest software, if the VM is running faster than the real, the key repeat will occur faster, too. The host application doesn't have the way to avoid it. Therefore, nono will temporarily suppress the full speed mode while any keys are pressed. That is why the above-mentioned icon distinguishes them. If keys continue to be pressed for some reasons, the VM also continues to run syncronized mode. In this case, you can resolve it by using the software keyboard window.]

実行してみる [Try it]

NetBSD/luna68k を実行してみる [Try NetBSD/luna68k]

つついさんが NetBSD/luna68k 9.2 の liveimage を用意されています。 [Tsutsui-san has provided a liveimage of NetBSD/luna68k 9.2.]

https://twitter.com/tsutsuii/status/1262429647364427783
https://twitter.com/tsutsuii/status/1405206240913805313

ここではこれを起動してみます。[Let's try it.]

1. どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します (例えば ~/nono/luna/)。 [Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/luna/).]
2. 以下のリンクからイメージファイルをダウンロードして展開し、 VM ディレクトリ ~/nono/luna/ に置きます。 [Download imagefile from the following link, extract it and place it in the VM directory, ~/nono/luna/.]

   http://teokurebsd.org/netbsd/liveimage/20210614-luna68k/

3. 以下の内容の設定ファイル nono.cfg を VM ディレクトリ ~/nono/luna/ に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。 [Create a configuration file nono.cfg in the VM directory, ~/nono/luna/, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = luna
spc0-id6-image = hd,liveimage-luna68k-raw-20210614.img
hostnet-driver = none
```

4. `nono -c ~/nono/luna` で起動します (VM ディレクトリに自動的に NVRAM.DAT が作られます)。 [Run as `nono -c ~/nono/luna`. (NVRAM.DAT will be created automatically in the VM directory)]
5. Emulated ROM Monitor が起動するので、初回は以下のように入力すると NetBSD が起動します。 [The emulated ROM Monitor will be executed. Then, only for the first time, entering the following can boot NetBSD.]

```
k↲
↲
↲
d↲
boot↲
g↲
x↲
```

画面はこんな感じのはずです (太字が入力部分)。 [You will see a screen like this. The bold text indicates the characters you need to enter.]

```
NONO 0.2.0 Emulated ROM Monitor for LUNA-I

** NVRAM Initialized.

>k↲
controller: dk  ?↲
drive unit: 0  ?↲
partition : c  ?d↲
filename  : vmunix  ?boot↲
>g↲
Loaded.  Entry point = $00700000
>x↲
```

この内容は NVRAM.DAT に記録されているので次回以降は直接 NetBSD が起動します。 [The information you have just entered is recorded in the NVRAM, so next time it boots NetBSD automatically.]

6. 初回起動時、 Updating fontconfig cache はあほみたいに時間がかかりますが、 nono がハングアップしてるわけではありません(>_<)。また初回ログイン時めちゃくちゃ重たいですが、これはバックグラウンドで makemandb が動くためで nono のせいではありません(>_<)。 [At the first boot, you will see the console stops after printing "Updating fontconfig cache". This is because the infamous fontconfig takes very looooong time. nono would not have hang-up. In addition, after the first login, you will feel it's too heavy. This is because the infamous makemandb(8) runs heavily in the background for a looooong time. It's very sad to me that these two accidents which are far from the ideal are the first experiences of a newcomer.]
7. 終了する時は root ユーザで “`shutdown -p now`” を実行してください。 LUNA はソフトウェアから電源オフでき、 VM の電源オフで nono も終了します。 [To quit, type “`shutdown -p now`” as the root user. LUNA can be powered off by software, and nono will terminate when the VM is powered off.]

OpenBSD/luna88k を実行してみる [Try OpenBSD/luna88k]

あおやまさんが OpenBSD/luna88k 6.9 の liveimage を用意されています。 [Aoyama-san has provided a liveimage of OpenBSD/luna88k 6.9.]

https://twitter.com/ao_kenji/status/1404784588015112192

ここではこれを起動してみます。[Let's try it.]

1. どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します (例えば ~/nono/luna88k/)。 [Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/luna88k/).]
2. 以下のリンクから liveimage-luna88k-raw-YYYYMMDD.img.gz をダウンロードして展開し、 VM ディレクトリ ~/nono/luna88k/ に置きます。 [Download liveimage-luna88k-raw-YYYYMMDD.img.gz from the following link, extract it and place it in the VM directory, ~/nono/luna88k/.]

   http://www.nk-home.net/~aoyama/liveimage/

3. 以下の内容の設定ファイル nono.cfg を VM ディレクトリ ~/nono/luna88k/ に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。 [Create a configuration file nono.cfg in the VM directory, ~/nono/luna88k/, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = luna88k
spc0-id6-image = hd,liveimage-luna88k-raw-20210614.img
hostnet-driver = none
```

4. `nono -c ~/nono/luna88k` で起動します (VM ディレクトリに自動的に NVRAM.DAT が作られます)。 [Run as `nono -c ~/nono/luna88k`. (NVRAM.DAT will be created automatically in the VM directory)]
5. Emulated ROM Monitor が起動するので、初回は以下のように入力すると OpenBSD が起動します。 [The emulated ROM Monitor will be executed. Then, only for the first time, entering the following can boot OpenBSD.]

```
nvram boot_filename boot↲
```

```
y
b↵
```

画面はこんな感じのはずです (太字が入力部分)。 [You will see a screen like this. The bold text indicates the characters you need to enter.]

```
NONO 0.2.0 Emulated ROM Monitor for LUNA88K

** NVRAM Initialized.

N>nvram boot_filename boot↵
Update boot_filename : "vmunix" -> "boot" (Y/[N]):y
Updated
N>b↵
```

この内容は NVRAM.DAT に記録されているので次回以降は直接 OpenBSD が起動します。 [The information you have just entered is recorded in the NVRAM, so next time it boots OpenBSD automatically.]

6. 終了する時は root ユーザで “shutdown -p now” を実行してください。 LUNA88K はソフトウェアから電源オフでき、 VM の電源オフで nono も終了します。 [To quit, type “shutdown -p now” as the root user. LUNA88K can be powered off by software, and nono will terminate when the VM is powered off.]

ネットワーク設定例 [Example of network setup]

wm0 を持つ NetBSD ホストに tap(4) デバイスを用いて nono のゲスト OS を接続する場合の設定例です。

1. 設定ファイル nono.cfg に以下の行を追加します (と言いつつ NetBSD では書かなくてもデフォルトでこの動作になりますが) [Add the following line to configuration file, nono.cfg. (Although you don't need to write it since these are default behavior on NetBSD)]

```
hostnet-driver = tap
hostnet-tap-devpath = auto
```

2. デフォルトでは /dev/tap は一般ユーザからアクセスできないので、 chmod で適当にパーミッションを与えます。番号の付いていないほうの /dev/tap だけでいいです。 sysinst 等で OS をアップグレードするとパーミッションが 600 に戻るのがハマりポイントです。 [By default, /dev/tap is only accessible to privileged user. You need to chmod /dev/tap (without unit number) appropriately. Note that upgrading using sysinst always reset the permission to 600.]
3. bridge(4) インタフェースを作成し、ホストの外部(物理)インタフェースをブリッジに追加しておきます。 [Create a bridge(4) interface, and add your physical interface to the bridge.]

```
# ifconfig bridge0 create
# brconfig bridge0 add wm0
```

常用するなら /etc の設定ファイルに書いておきましょう。 [If you use it regularly, you can put configuration file into /etc.]

/etc/ifconfig.bridge0

```
create
up
!/sbin/brconfig $int add wm0

# /etc/rc.d/network restart
```

4. 一般ユーザに戻って、 VM ディレクトリかその親ディレクトリに次のような 2つのスクリプトを用意します。 nono は tap(4) をオープンし、そのデバイス名を引数にこれらのスクリプトを呼びます。 sudo の設定は別途行ってください。 [Return to non-privileged user, and create following two scripts in the VM directory or its parent directory. nono will open tap(4) and invoke these scripts with the name of the device as an argument. In addition, you need to set up sudo separately.]

nono-ifup

```
#!/bin/sh
sudo /sbin/ifconfig $1 up
sudo /sbin/brconfig bridge0 add $1
```

nono-ifdown

```
#!/bin/sh
sudo /sbin/brconfig bridge0 delete $1
sudo /sbin/ifconfig $1 down

% chmod +x nono-ifup nono-ifdown
```

5. nono を起動し、メニューの「モニタ > ホスト > ホストネットワーク」を開いて HostNet Driver: tap になっていれば動いてるはずです。 [Run nono, and open "Monitor > Host > Host Network" window from menu. It's OK if you can see "HostNet Driver: tap".]

変更履歴 [Changes]

See changes.html.

過去のバージョンからの移行方法 [How to migrate from old versions]

バージョンアップに伴い設定ファイル等に非互換が発生する場合があります。その場合は以下の移行方法を参照して設定ファイル等を更新してください。 [Some versions may have incompatibilities in the configuration files, etc. In such case, you may

need to upgrade it by referring the following link.]

- From ver 0.1.x to ver 0.2.0

ライセンス [License]

See nono-license.txt.

連絡先 [Contact us]

バグ報告などは以下にお願いします。日本語でおk。 [If you find any problems, please let me know. You may write in English.]
https://github.com/isaki68k/nono-issue/issues

パッチの提供について [About contributes]

パッチを提供してくださる場合は以下に同意したものとします。 [If you provide a patch to nono, you must agree to the following conditions:]

- 成果物が nono のライセンスに従って運用あるいは配布されること。 [All your work are operated or distributed under the nono license.]
- ライセンスが将来変わる可能性があること。 [The license may be changed in the future.]
- 著作部分に関して著作者人格権を行使しないこと。 [Do not exercise your author's rights.]

Acknowledgements

nono は以下の広告条項を含むソースコードを利用しています。 [nono uses source code with the following advertising clause.]

> This product includes software developed by Gordon Ross
> This product includes software developed by the University of California, Lawrence Berkeley Laboratory.

---

nono project

nono 0.2.1 (2021/08/18)

nono は NetBSD とかで動作する OMRON LUNA-I や LUNA88K とかのエミュレータです。 [nono is OMRON LUNA-I and LUNA88K emulator runs on NetBSD and etc.]

Index of this page:

ビルド方法 [How to build]

ビルドには以下が必要です。 NetBSD(pkgsrc) なら wxWidgets は pkgsrc/x11/wxGTK30 です (OPTIONS:gtk3 で動作確認しています)。 [The followings are required for build. If you use NetBSD(pkgsrc), wxWidgets is pkgsrc/x11/wxGTK30 (OPTIONS:gtk3 is tested).]

- make (BSD make, not GNU make)
- C/C++ compiler which supports -std=c++14.
  (For gcc, 7.4 works but 5.5 doesn't work, at least. For clang, 7.0 and 8.0 works at least.)
- wxWidgets >= 3.0
- gettext

(NetBSD 以外でのビルドはサポートしていませんが) Ubuntu 20.04 ではたぶん以下のパッケージが必要です。 [You may need the following packages on Ubuntu 18.04 (though we won't support non-NetBSD platform).]

- bmake build-essential gettext libbsd-dev libedit-dev libkqueue-dev libwxgtk3.0-gtk-dev zlib1g-dev

nono のソースアーカイブを展開したら以下のようにビルドします。 configure には環境変数 CC、CXX でコンパイラを指定することが出来ます。また wx-config が標準的な名前で提供されていないために見付けられない場合には環境変数 WX_CONFIG にパスを指定することが出来ます。 [Extract the nono's source archive and build as following. You can specify C/C++ compiler using environment variable CC and CXX if configure cannot find standard name suitable compiler. Also, you can specify wx-config path using environment variable WX_CONFIG if configure cannot find wx-config.]

```
% ./configure
% make depend
% make
% su
# make install
```

2つの実行ファイルがインストールされます。 `nono` が GUI 版実行ファイル、 `nono-cli` がコマンドライン版です。 [Two executables will be installed. `nono` is the GUI executable and `nono-cli` is the command line executable.]

コマンドラインオプション [Command Line Option]

`-c` *vmpath*
: VM ディレクトリ/設定ファイルを指定します。 *vmpath* がディレクトリならそのディレクトリの中の nono.cfg を設定ファイルとします。 *vmpath* がファイルならそれを設定ファイルとします。そしていずれの場合も設定ファイルがあるディレクトリを VM ディレクトリとします。 -c オプションを省略すると *vmpath* をカレントディレクトリとします。 [Specifies the VM directory/configuration file. If *vmpath* is a directory, make nono.cfg in that directory a configuration file. Or if *vmpath* is a file, make the specified file a configuration file. And in both cases, make the directory where that file is located a VM directory. If `-c` option is omitted, *vmpath* is considered as the current directory.]

`-f`
: 高速モードで起動します。 GUI なら起動後にもメニューから変更できますが、その初期値を変えるだけです。 [Boot as the fast mode. You can change this mode on GUI menu after boot, and the option only changes its initial state.]

`--fontsize` *height*
: GUI 版のみ。全サブウインドウの起動時のフォントサイズを指定します。 *height* には 12, 16 のいずれかを指定します。デフォルトは `12` です。起動後にメニューから変更できます。 [GUI Only. Specifies the initial fontsize on all sub windows. *height* is one of 12 or 16. The default value is `12`. You can change this value on GUI menu after boot.]

`-s` *scale*
`--scale` *scale*
: GUI 版のみ。メインウィンドウの起動時のスケールを実数で指定します。起動後にもメニューからプリセットされた倍率には変更可能ですが、任意倍率は起動時のみ指定可能です (そのうちなんとかしたい)。デフォルトは 1.0 です。 [GUI Only. Specifies the initial main window scale in real number. You can change this scale on GUI menu after boot, but unlike this option, there are only a few preset choices (Should be improved in someday). The default value is 1.0]

`--show-config`
: 設定ファイルと `-v` オプションを読み込んだ結果を表示します。 [Shows the result of reading configuration file and

parsing -v options.]

-v
バージョンを表示します。 [Shows the version.]

-V *name=value*
設定ファイルで指定した *name=configvalue* の代わりにこのオプションの *name=value* を適用します。 [Use this *name=value* instead of *name=configvalue* specified in configuration file.]

-X *file*
ホストの *file* をロードして実行します。 *file* が相対パスの場合カレントディレクトリからのパスになります。ファイルが gzip 圧縮されていれば自動的に展開します。 (展開後の) ファイル形式は a.out (OMAGIC) か ELF で、実際にはブートローダとカーネル程度しか想定していません。設定ファイルの prom-image とともに指定されると -X のほうが優先します。 [Loads and executes host's *file*. If *file* is relative path, it is path from the current directory. If the file is gzip'd, it is automatically extracted. The supported file format (after extracting) is a.out (OMAGIC) or ELF. Actually, it only assumes bootloaders or kernels. If this option is specified at the same time as prom-image in configuration file, this option preceeds.]

以下開発用。[For developers:]

-b *hexaddr*[,*skipcount*]
デバッガのブレークポイントを 16進数で指定します。
-C
ログをコンソールにも出力します。通常はログウィンドウにだけ出力されます。
-d
起動時にデバッガプロンプトで停止します。
-D
コンソールをデバッガとして使用します。 -d を指定しなくても起動時にプロンプトで停止します (そのうちなんとかしたい)。
-L *name1=level1*[,*name2=level2*,...]
ログレベルを指定します。カンマで区切って複数指定することも出来ます。 -Lhelp で name の一覧を表示します。
-M *name*[,*name2*,...]
起動時に表示するモニタウィンドウを指定します。カンマで区切って複数指定することも出来ます。 -Mhelp で name の一覧を表示します。

設定 [Configuration]

VM の設定ファイルはその VM ディレクトリ内の nono.cfg (または -c で指定したファイル) です。書式は key = value 形式で1行1項目ずつです。 key と value の前後の空白は取り除かれます。また空行と "#" で始まる行は無視します。知らないキーの行も無視します。 [VM configuration file is nono.cfg (or the file specified by -c option) in the VM directory. Its syntax is key = value format, one per line. White spaces before and after key and value are ignored. And, blank lines, lines beginning with "#", and lines with unrecognized key are also ignored.]

設定項目は次の通りです。 [The configuration items are:]

vmtype = *string*
VM 種別を以下のいずれかから指定します。省略不可です。 [Specifies the VM type from the following. This field is mandatory.]
luna ... LUNA-I
luna88k... LUNA88K

clock-sync = *value*
仮想マシン内の時刻の同期方法を指定します。 real なら実時間に同期、virtual なら仮想時間に同期します。デフォルトは real です。この機能は実験中のため将来予告なく仕様が変更になる可能性があります。 [Specify how to synchronize the time in virtual machine. If real, synchronize with the real time; if virtual, synchronize with the virtual time. The default is real. This feature is under experimentation and may be changed in the future without notice.]

debugger-port = *integer*
デバッガの TCP 待ち受けポート番号を指定します。 0 なら待ち受けを行いません。デフォルトは 0 です。 [Specifies the TCP port number that debugger listens. If 0, it will not listen. The default is 0.]

ethernet-macaddr = *string*
イーサネットデバイスの仮想マシン側の MAC アドレスを指定します。 *xx*:*xx*:*xx*:*xx*:*xx*:*xx* 形式で指定します。 auto なら自動的に決定します。デフォルトは auto です。

hostnet-driver = *string*
イーサネットデバイスのホスト側ドライバを指定します。 afpacket、bpf、tap のうちホスト OS がサポートしているものと auto、none が選択可能です。 afpacket は Linux の AF_PACKET ソケットを使用します。 bpf は bpf(4) デバイスを使用します。 tap は tap(4) インタフェースを使用します。 none ならホスト側とは一切通信を行いません。 auto ならホスト OS がサポートしているもののうち tap → afpacket → bpf を順に試します。デフォルトは auto です。 [Specifies the host driver of ethernet device. afpacket, bpf, and tap can be specified only if the host OS supports them. auto and none can always be specified. afpacket uses Linux's AF_PACKET socket, bpf uses bpf(4) device, and tap uses tap(4) interface. none doesn't make any communication with the host. If auto is specified, it tries tap, afpacket, and bpf in that order (if the host OS supports them). The default is auto.]

hostdriver-afpacket-ifname = *ifname*
ホストドライバが afpacket の時にバインドするインタフェースを1つ指定します。 *ifname* が auto なら使用可能なインタフェースを1つ自動的に選択します。デフォルトは auto です。なお、このホストドライバではローカルホストとの通信は出来ま

せん。 [Specify an interface name to bind to, when the host driver is afpacket. If auto is specified as *ifname*, it selects an usable interface automatically. The default is auto. Note that this host driver cannot communicate with the localhost.]

hostdriver-bpf-ifname = *ifname*
ホストドライバが bpf の時にバインドするインタフェースを1つ指定します。 *ifname* が auto なら使用可能なインタフェースを1つ自動的に選択します。デフォルトは auto です。なお、このホストドライバではローカルホストとの通信は出来ません。 [Specify an interface name to bind to, when the host driver is bpf. If auto is specified as *ifname*, it selects an usable interface automatically. The default is auto. Note that this host driver cannot communicate with the localhost.]

hostdriver-tap-devpath = *path*
ホストドライバが tap の時に使用するデバイスを1つフルパスで指定します。デフォルトは auto で、この場合は自動的にデバイスを選択します。この時の探し方はホスト OS によって異なります。 Linux なら /dev/net/tun を指定したのと同じです。 OpenBSD なら /dev/tap0 から /dev/tap9 まで順番にオープンできるまで試します。 NetBSD (と FreeBSD) ならまず /dev/tap でクローニングを試み、それが失敗すれば /dev/tap0 から /dev/tap9 までを順番にオープンできるまで試します。いずれの場合も VM ディレクトリかその親ディレクトリに nono-ifup, nono-ifdown という名前のシェルスクリプトが必要です。何もすることがない場合でも正常終了する空のシェルスクリプトを用意してください。 [Specify a device pathname, when the host driver is tap. The default is auto. The behavior in this case depends on the host OS. On Linux, it's the same as /dev/net/tun. On OpenBSD, it will try from /dev/tap0 to /dev/tap9 until successful. On NetBSD (and FreeBSD), it will try cloning by /dev/tap first. If that fails, then try from /dev/tap0 to /dev/tap9 until successful. In all cases, you need to prepare two shell script files which names are nono-ifup and nono-ifdown in the VM directory or its parent directory. Even if you don't have anything to do in these scripts, you need to prepare empty scripts that will terminate successfully.]

hostnet-fallback = *integer*
hostnet-driver で指定したドライバが使用可能でなかった時、 0 ならプロセスを終了します。 1 なら none を選択して実行を継続します。デフォルトは 0 です。 [Specify the behavior when the driver which is selected by hostnet-driver is unusable; terminate the process if 0, or continue to run using the none driver if 1. The default is 0.]

luna-adjust-misused-epoch = *integer*
LUNA で誤った RTC epoch を採用している OS 向けに RTC エミュレーションを補正するかどうかを指定します。0 なら補正をしません(実機と同じ動作)、 1 なら補正します(現実世界と同じ動作)。デフォルトは 1 で、通常 1 のままで使用して問題ありません。 [Specifies whether nono corrects RTC emulation for OSes that adopts wrong RTC epoch on LUNA. 0 means making no correction (this is the same behavior as the actual machine). 1 means making correction (this is the same behavior as the real world). The default is 1. Normally, leave it 1.]

LUNA で採用している RTC (MK48T02) は2桁で保持している年の値が 4 で割り切れる年をうるう年とする仕様です。ところが NetBSD/luna68k、OpenBSD/luna88k など現存してソースが確認できる OS はいずれもこの RTC の年の値を 1970年からの経過年として使用しています (LUNA が本来どういう仕様だったのかは分かりませんが、異なる OS を起動し直すたびに RTC epoch をずらす運用は考えにくいのでおそらくすべての OS で同じだと思います)。例えば1970年はうるう年ではないため 2月28日の翌日は 3月1日ですが、 MK48T02 的には 00年であるためうるう年と認識し 2月28日の翌日が2月29日になります。このように、実機の RTC は4年のうち約2年間、1日ずれた日付を指しているようです。しかしながら、RTC の時刻は OS 起動時に一度読んだ後は基本的に参照しない上、今時必要なら NTP で時間を合わせるため、実機でも問題が顕在化することはまずないと思います。 nono の場合は実機と異なり、アプリケーション実行中しか RTC が進まないため、補正がない場合の動作が問題になるのは nono を起動したまま偶数年の2月末日から日付をまたいで、かつ nono を起動したまま OS を再起動して NTP などで時刻修正を行わなかった時だけだと思います。このオプションはほぼ開発者向けの動作確認用です。

luna-dipsw1 = *string*
本体前面 DIPSW#1-1..#1-8 の内容を指定します。 "0" を DOWN、"1" を UP として、これを8つ並べた形式で、前から順に #1..#8 に対応します。 [Specifies status of the front panel DIPSW#1-1..#1-8 using 8 digit. "0" means DOWN and "1" means UP. The first character corresponds to #1 and the eighth character corresponds to #8.]

LUNA-I でのデフォルトは 11110111 です。各スイッチの内容は以下のリンクを参照してください。 [On LUNA-I, the default value is 11110111. See the following link about DIPSW.]
→ NetBSD/luna68k: Information
LUNA88K でのデフォルトは 11111111 です。各スイッチの内容は以下のリンクを参照してください。 [On LUNA88K, the default value is 11111111. See the following link about DIPSW.]
→ OpenBSD manual pages: boot_luna88k(8)

luna-dipsw2 = *string*
本体前面 DIPSW#2-1..#2-8 の内容を指定します。書式は luna-dipsw1 と同じです。デフォルトは 11111111 です。 [Specifies status of the front panel DIPSW#2-1..#2-8. The same syntax as luna-dipsw1 is used. The default value is 11111111.]

NetBSD/luna68k のブートローダは、 DIPSW#2 が "11111111" なら自動的にカーネルをロードして実行し、どれかでも "0" にするとプロンプトで停止するようです。 (本当は #8 だけで制御するつもりだったんじゃないかという気がします)
[NetBSD/luna68k bootloader will automatically load and execute the kernel, if the DIPSW#2 is "11111111". Otherwise, the bootloader will enter interactive mode. (I doubt that they actually wanted to switch with only #8)
]

monitor-rate = *integer*
テキスト系モニタウィンドウの更新頻度を Hz 単位で指定します。 1 から 60 までの間で指定でき、デフォルトは 20Hz です。起動後にメニューからプリセットされた頻度には変更可能です。 [Specifies refresh rate of all text monitor windows in Hz. It ranges from 1 to 60. The default is 20Hz. You can change this value on GUI menu after boot, but unlike this configuration value, there are only a few preset choices.]

mpu-clock = *value*

MPU のクロック数を MHz 単位で指定します。デフォルトは LUNA-I なら 20MHz、LUNA88K なら 25MHz です。 [Specifies the MPU clock in MHz. The default value is 20MHz on LUNA-I, or 25MHz on LUNA88K.]

`mpu-pseudo-stop = integer`

m88100 にて疑似 STOP 状態を有効にするかどうかを指定します。 0 なら無効(実機と同じ動作)、1 なら有効で、デフォルトは 1 です。 m88100 には、m68k の STOP 命令 (割り込みが上がるまで何もせず待つ) に相当する命令がなく、大抵ビジーウェイトループで割り込みが上がるのを待つことになります。これは実機では (消費電力を減らす手段がないという些細な問題以外には) 何のデメリットもないのですが、エミュレータで特に高速動作させている時には割り込みが上がるまで (例えば人間がキーを入力するまで) ホスト CPU パワーを使い潰してビジーウェイトループを実行し続けることになり、ホスト CPU があっつあつになります。それを防ぐための機能です。特徴的な命令列を検出して実現しているので、すべての状況で動作するわけではありません。

`prom-image = path`

LUNA-I/LUNA88K の外部 ROM イメージファイルのパスを指定します。 *path* がファイル名のみなら VM ディレクトリとその親ディレクトリからこのファイル名を検索します。 *path* が相対パスなら VM ディレクトリからの相対パスになります (現在のディレクトリからではありません)。空にすると内蔵 ROM を使用します。デフォルトは空です。 [Specifies the LUNA-I/LUNA88K's external ROM image file path. If the *path* does not have any path delimiters, the VM directory and then its parent directory will be searched. If the *path* is a relative path, it will be path from the VM directory, not from the current directory. If the *path* is empty, internal emulated ROM will be used. The default value is empty.]

実機を持っていない場合はこの値を空に (= デフォルトのままに) しておくと、 nono 内蔵のなんちゃって下位互換 ROM で起動します。 [If you does not have the real LUNA machines, you can boot with nono's internal downward compatible emulated ROM if you set this field empty (or leave it as the default).]

LUNA-I 実機を持っている場合は ROM ファイルを指定することで実機 ROM で起動できます。 ROM ファイルは実機の 0x41000000-0x4101ffff (128KB) を保存したものです。今のところ ROM は V4.22 (Thu Jul 27 11:45:42 1989) のみサポートしています。それ以外については何も分かりません。 [If you have the real LUNA-I machine, you can boot with the real ROM spcifying the ROM file path. The ROM file is extracted from 0x41000000-0x4101ffff (128KB) of the real LUNA-I machine. For now, only V4.22 (Thu Jul 27 11:45:42 1989) is supported. I have no idea about other ROMs.]

LUNA88K 実機の場合は 0x41000000-0x4103ffff (256KB) を保存したものです。今のところ ROM は version 1.20 のみサポートしています。 [For LUNA88K, the ROM file is extracted from 0x41000000-0x4103ffff (256KB). For now, only version 1.20 is supported.]

`ram-size = integer`

搭載する RAM サイズを MB 単位で指定します。 LUNA-I のデフォルトは 16MB です。 16MB 未満は 4MB 単位で、 16MB 以上は 255MB まで 1MB 単位で指定できます (ちなみに NetBSD/luna68k の起動には最低でも 8MB 必要です)。 LUNA88K のデフォルトは 64MB です。 64MB 未満は 16MB 単位で、 64MB 以上は暫定で 255MB まで 1MB 単位で指定できます。 [Specifies the RAM size in MB. On LUNA-I, the default is 16MB. If the size is less than 16MB, you can specify in 4MB unit. If larger, you can specify up to 255MB in 1MB unit. By the way, NetBSD/luna68k needs at least 8MB to boot. On LUNA88K, the default is 64MB. If the size is less than 64MB, you can specify in 16MB unit. If larger, you can specify up to tentative 255MB in 1MB unit.]

`show-statuspanel = integer`

ステータスパネルを表示するかどうかを指定します。 0 なら非表示、1 なら表示です。起動後はメニューから変更可能です。 [Specifies whether to display the status panel or not. If 0, it is hidden; if 1, it is shown. You can change it on GUI menu after boot.]

`spc0-id`*N*`-image = devtype[,path]`

SCSI デバイスとイメージを指定します。キーの *N* には 0 から 7 が入ります。ただし ID 7 は本体が使用しますので指定しないでください。値はデバイス種別 *devtype* とディスクイメージパス *path* を ","(カンマ) で区切って並べた形式です。デバイス種別 *devtype* は以下のいずれかです。 [Specifies SCSI device and image. *N* in the key is 0 to 7. But don't specify ID 7 because the host uses it. The value is in a form of device type *devtype* and the disk image path *path* separated by ","(comma). *devtype* can be one of the following:]

- `hd` ... HD drive
- `cd` ... CD-ROM drive
- `mo` ... MO drive

devtype が `hd` なら *path* は省略できません。 *devtype* が `cd` か `mo` なら *path* は省略可能です。イメージパスが相対パスなら VM ディレクトリからの相対パスになります。 [If *devtype* is `hd`, *path* cannot be ommitted. If *devtype* is `cd` or `mo`, *path* can be ommitted. If the *path* is relative path, it is from the VM directory.]

例えば、nono.cfg と同じディレクトリに置いた sd0.img を起動 HDD ディスクイメージとして使い (LUNA では通常 ID 6 をプライマリ HDD に割り当てます)、 ID 5 に同じディレクトリの install.iso をセットした CD ドライブを、 ID 4 に起動時メディアなしの MO ドライブを接続する場合は次のようになります。 [For example, if you use a harddisk image sd0.img placed in the same directory as nono.cfg (LUNA usually assigns ID 6 to the primary HDD), ID 5 for CD-ROM drive that loads install.iso in the same directory, and ID 4 for MO drive without media on boot, write as following:]

```
spc0-id6-image = hd,sd0.img
spc0-id5-image = cd,install.iso
spc0-id4-image = mo
```

`spc0-id`*N*`-seektime = integer`

指定の SCSI HDD の平均シークタイムを msec 単位で指定します。現在のデフォルトは `0` です (S・S・D!! S・S・D!!)。 16 程度を指定すると幾分往時に思いを馳せることが出来るかもしれませんが、今の所あまり安定していません。 [Specifies the average seek time of specified SCSI HDD in msec. Currently, the default value is `0` (This may be something like

SSD :-). If you specify about 16 or so, you can feel nostalgic, but this feature is still unstable.]

`spc0-id`*N*`-writeignore = integer`
: 指定の SCSI HD デバイスへの書き込みを無視するかどうか指定します。 `0` なら通常動作(書き込みを行う)です。 `1` なら書き込みコマンドは成功したように振る舞いますが実際にはディスクイメージに一切書き戻しません。 fsck を気にせずカーネルのデバッグとかを行いたい場合にはどうぞ。何が起きるか意味が分からない人は指定しないでください。デフォルトは `0` です。 [Specifies whether nono ignores writing to SCSI HD devices. `0` means normal operation (writes to the devices). If `1` is specified, nono will not actually write back to the disk image even though the write command is succeeded. nono's SCSI devices acts as write command is successfully done but it never writes back to the actual disk image. This is useful for kernel debugging because it does not require fsck after the kernel hangs. But don't use this flag if you don't understand this paragraph. The default value is `0`.]

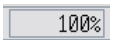
  ちなみに、メディアを書き込み禁止にしたい場合はこれではなく、イメージファイルの書き込み権を落としてください。 [By the way, if you want to make the media write-protected, clear the write permission from the image file (instead of this setting).]

`spc0-id`*N*`-writeprotect = integer`
: 古いオプションです。代わりに `spc0-id`*N*`-writeignore` を使ってください。このオプションは開発用です。 [Obsolete. Use `spc0-id`*N*`-writeignore` instead. This option is for developers.]

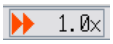
VM について [About VM]

ステータスパネル [Status Panel]

ステータスパネル中央にあるパフォーマンスメータのアイコンは高速モードの状態を表示しています。ダブルクリックすると高速/等速モードの指定を切り替えることができます。 [The performance meter's icon at the center of the status panel shows the VM speed status. You can switch full speed / syncronized mode by double-clicking on this icon.]



... マークなしの場合、ユーザが等速モードを指定していて、等速モードで実行中です。 [When no icons are displayed, the user has specified syncronized mode and the VM is running in synchronized mode.]



... 三角2つの場合、ユーザが高速モードを指定していて、等速モードで実行中です。キー入力中(後述)または VM がアイドル状態(m68k の STOP 命令)などで起きます。高速モード中はパーセントではなく何倍速で動作しているかを表します。 [When an icon (two triangles) is displayed, the user has specified full speed mode and the VM is running in synchronized mode. This will happen during keystrokes (see below) or when the VM is idle (STOP instruction on m68k).]

... 三角3つの場合、ユーザが高速モードを指定していて、高速モードで実行中です。 [When an icon (three triangles) is displayed, the user has specified full speed mode and the VM is running in full speed mode.]

キー入力 [Key input]

LUNA では、キーボードがハードウェア側でキーリピートを行わず、ソフトウェア(OS など)がキーリピートの処理を行います。そのため、キーリピートを実装していない LUNA の PROM や NetBSD/luna68k のブートローダではキーリピートは起きず、キーリピートを実装している NetBSD/luna68k カーネルではキーリピートが起こせます。キーリピートを起こす間隔をゲストのソフトウェアが測定しているということは、 VM が高速動作するとキーリピートもそれに合わせて発生することになり、これをホスト側から防ぐ手段はありません。そこで nono ではキー入力が発生している間 (キーが一つでも押されてからキーが全て離されるまでの間)、高速モードが指示されていても VM を一時的に等速モードに落として実行します。上記のアイコンがそれを区別しているのはこのためです。そのため、何らかの理由でキーが入りっぱなしになった場合 (ALT + TAB やアクセラレータキーでメニューを開くと起きがちです) 高速モードが抑制されたままということが起きえます。その場合はソフトウェアキーボードからキー入力を解除するなどしてください。 [On LUNA, key repeat is done by software(OS), not by the keyboard hardware. For this reason, key repeat doesn't occur on LUNA's PROM or NetBSD/luna68k's bootloader that don't implement it, and key repeat occurs on NetBSD/luna68k kernel that implements it. Since the timing of key repeat is measured by the guest software, if the VM is running faster than the real, the key repeat will occur faster, too. The host application doesn't have the way to avoid it. Therefore, nono will temporarily suppress the full speed mode while any keys are pressed. That is why the above-mentioned icon distinguishes them. If keys continue to be pressed for some reasons, the VM also continues to run syncronized mode. In this case, you can resolve it by using the software keyboard window.]

実行してみる [Try it]

NetBSD/luna68k を実行してみる [Try NetBSD/luna68k]

つついさんが NetBSD/luna68k 9.2 の liveimage を用意されています。 [Tsutsui-san has provided a liveimage of NetBSD/luna68k 9.2.]

https://twitter.com/tsutsuii/status/1262429647364427783
https://twitter.com/tsutsuii/status/1405206240913805313

ここではこれを起動してみます。[Let's try it.]

1. どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します (例えば ~/nono/luna/)。 [Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/luna/).]
2. 以下のリンクからイメージファイルをダウンロードして展開し、 VM ディレクトリ ~/nono/luna/ に置きます。 [Download imagefile from the following link, extract it and place it in the VM directory, ~/nono/luna/.]

   http://teokurebsd.org/netbsd/liveimage/20210614-luna68k/

3. 以下の内容の設定ファイル nono.cfg を VM ディレクトリ ~/nono/luna/ に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。 [Create a configuration file nono.cfg in the VM directory, ~/nono/luna/, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = luna
spc0-id6-image = hd,liveimage-luna68k-raw-20210614.img
hostnet-driver = none
```

4. `nono -c ~/nono/luna` で起動します (VM ディレクトリに自動的に NVRAM.DAT が作られます)。 [Run as `nono -c ~/nono/luna`. (NVRAM.DAT will be created automatically in the VM directory)]
5. Emulated ROM Monitor が起動するので、初回は以下のように入力すると NetBSD が起動します。 [The emulated ROM Monitor will be executed. Then, only for the first time, entering the following can boot NetBSD.]

```
k↲
↲
↲
d↲
boot↲
g↲
x↲
```

画面はこんな感じのはずです (太字が入力部分)。 [You will see a screen like this. The bold text indicates the characters you need to enter.]

```
NONO 0.2.0 Emulated ROM Monitor for LUNA-I

** NVRAM Initialized.

>k↲
controller: dk  ?↲
drive unit: 0  ?↲
partition : c  ?d↲
filename  : vmunix  ?boot↲
>g↲
Loaded.  Entry point = $00700000
>x↲
```

この内容は NVRAM.DAT に記録されているので次回以降は直接 NetBSD が起動します。 [The information you have just entered is recorded in the NVRAM, so next time it boots NetBSD automatically.]

6. 初回起動時、 Updating fontconfig cache はあほみたいに時間がかかりますが、 nono がハングアップしてるわけではありません(>_<)。また初回ログイン時めちゃくちゃ重たいですが、これはバックグラウンドで makemandb が動くためで nono のせいではありません(>_<)。 [At the first boot, you will see the console stops after printing "Updating fontconfig cache". This is because the infamous fontconfig takes very looooong time. nono would not have hang-up. In addition, after the first login, you will feel it's too heavy. This is because the infamous makemandb(8) runs heavily in the background for a looooong time. It's very sad to me that these two accidents which are far from the ideal are the first experiences of a newcomer.]
7. 終了する時は root ユーザで “`shutdown -p now`” を実行してください。 LUNA はソフトウェアから電源オフでき、 VM の電源オフで nono も終了します。 [To quit, type “`shutdown -p now`” as the root user. LUNA can be powered off by software, and nono will terminate when the VM is powered off.]

OpenBSD/luna88k を実行してみる [Try OpenBSD/luna88k]

あおやまさんが OpenBSD/luna88k 6.9 の liveimage を用意されています。 [Aoyama-san has provided a liveimage of OpenBSD/luna88k 6.9.]

https://twitter.com/ao_kenji/status/1404784588015112192

ここではこれを起動してみます。[Let's try it.]

1. どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します (例えば ~/nono/luna88k/)。 [Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/luna88k/).]
2. 以下のリンクから liveimage-luna88k-raw-YYYYMMDD.img.gz をダウンロードして展開し、 VM ディレクトリ ~/nono/luna88k/ に置きます。 [Download liveimage-luna88k-raw-YYYYMMDD.img.gz from the following link, extract it and place it in the VM directory, ~/nono/luna88k/.]

   http://www.nk-home.net/~aoyama/liveimage/

3. 以下の内容の設定ファイル nono.cfg を VM ディレクトリ ~/nono/luna88k/ に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。 [Create a configuration file nono.cfg in the VM directory, ~/nono/luna88k/, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = luna88k
spc0-id6-image = hd,liveimage-luna88k-raw-20210614.img
hostnet-driver = none
```

4. `nono -c ~/nono/luna88k` で起動します (VM ディレクトリに自動的に NVRAM.DAT が作られます)。 [Run as `nono -c ~/nono/luna88k`. (NVRAM.DAT will be created automatically in the VM directory)]
5. Emulated ROM Monitor が起動するので、初回は以下のように入力すると OpenBSD が起動します。 [The emulated ROM Monitor will be executed. Then, only for the first time, entering the following can boot OpenBSD.]

```
nvram boot_filename boot↲
```

```
y
b↵
```

画面はこんな感じのはずです (太字が入力部分)。 [You will see a screen like this. The bold text indicates the characters you need to enter.]

```
NONO 0.2.0 Emulated ROM Monitor for LUNA88K

** NVRAM Initialized.

N>nvram boot_filename boot↵
Update boot_filename : "vmunix" -> "boot" (Y/[N]):y
Updated
N>b↵
```

この内容は NVRAM.DAT に記録されているので次回以降は直接 OpenBSD が起動します。 [The information you have just entered is recorded in the NVRAM, so next time it boots OpenBSD automatically.]

6. 終了する時は root ユーザで “shutdown -p now” を実行してください。 LUNA88K はソフトウェアから電源オフでき、 VM の電源オフで nono も終了します。 [To quit, type “shutdown -p now” as the root user. LUNA88K can be powered off by software, and nono will terminate when the VM is powered off.]

ネットワーク設定例 [Example of network setup]

wm0 を持つ NetBSD ホストに tap(4) デバイスを用いて nono のゲスト OS を接続する場合の設定例です。

1. 設定ファイル nono.cfg に以下の行を追加します (と言いつつ NetBSD では書かなくてもデフォルトでこの動作になりますが) [Add the following line to configuration file, nono.cfg. (Although you don't need to write it since these are default behavior on NetBSD)]

```
hostnet-driver = tap
hostnet-tap-devpath = auto
```

2. デフォルトでは /dev/tap は一般ユーザからアクセスできないので、 chmod で適当にパーミッションを与えます。番号の付いていないほうの /dev/tap だけでいいです。 sysinst 等で OS をアップグレードするとパーミッションが 600 に戻るのがハマりポイントです。 [By default, /dev/tap is only accessible to privileged user. You need to chmod /dev/tap (without unit number) appropriately. Note that upgrading using sysinst always reset the permission to 600.]
3. bridge(4) インタフェースを作成し、ホストの外部(物理)インタフェースをブリッジに追加しておきます。 [Create a bridge(4) interface, and add your physical interface to the bridge.]

```
# ifconfig bridge0 create
# brconfig bridge0 add wm0
```

常用するなら /etc の設定ファイルに書いておきましょう。 [If you use it regularly, you can put configuration file into /etc.]

/etc/ifconfig.bridge0

```
create
up
!/sbin/brconfig $int add wm0

# /etc/rc.d/network restart
```

4. 一般ユーザに戻って、 VM ディレクトリかその親ディレクトリに次のような 2つのスクリプトを用意します。 nono は tap(4) をオープンし、そのデバイス名を引数にこれらのスクリプトを呼びます。 sudo の設定は別途行ってください。 [Return to non-privileged user, and create following two scripts in the VM directory or its parent directory. nono will open tap(4) and invoke these scripts with the name of the device as an argument. In addition, you need to set up sudo separately.]

nono-ifup

```
#!/bin/sh
sudo /sbin/ifconfig $1 up
sudo /sbin/brconfig bridge0 add $1
```

nono-ifdown

```
#!/bin/sh
sudo /sbin/brconfig bridge0 delete $1
sudo /sbin/ifconfig $1 down

% chmod +x nono-ifup nono-ifdown
```

5. nono を起動し、メニューの「モニタ > ホスト > ホストネットワーク」を開いて HostNet Driver: tap になっていれば動いてるはずです。 [Run nono, and open "Monitor > Host > Host Network" window from menu. It's OK if you can see "HostNet Driver: tap".]

変更履歴 [Changes]

See changes.html.

過去のバージョンからの移行方法 [How to migrate from old versions]

バージョンアップに伴い設定ファイル等に非互換が発生する場合があります。その場合は以下の移行方法を参照して設定ファイル等を更新してください。 [Some versions may have incompatibilities in the configuration files, etc. In such case, you may

need to upgrade it by referring the following link.]

- From ver 0.1.x to ver 0.2.0

ライセンス [License]

See nono-license.txt.

連絡先 [Contact us]

バグ報告などは以下にお願いします。日本語でおk。 [If you find any problems, please let me know. You may write in English.]
https://github.com/isaki68k/nono-issue/issues

パッチの提供について [About contributes]

パッチを提供してくださる場合は以下に同意したものとします。 [If you provide a patch to nono, you must agree to the following conditions:]

- 成果物が nono のライセンスに従って運用あるいは配布されること。 [All your work are operated or distributed under the nono license.]
- ライセンスが将来変わる可能性があること。 [The license may be changed in the future.]
- 著作部分に関して著作者人格権を行使しないこと。 [Do not exercise your author's rights.]

Acknowledgements

nono は以下の広告条項を含むソースコードを利用しています。 [nono uses source code with the following advertising clause.]

> This product includes software developed by Gordon Ross
> This product includes software developed by the University of California, Lawrence Berkeley Laboratory.

---

nono project