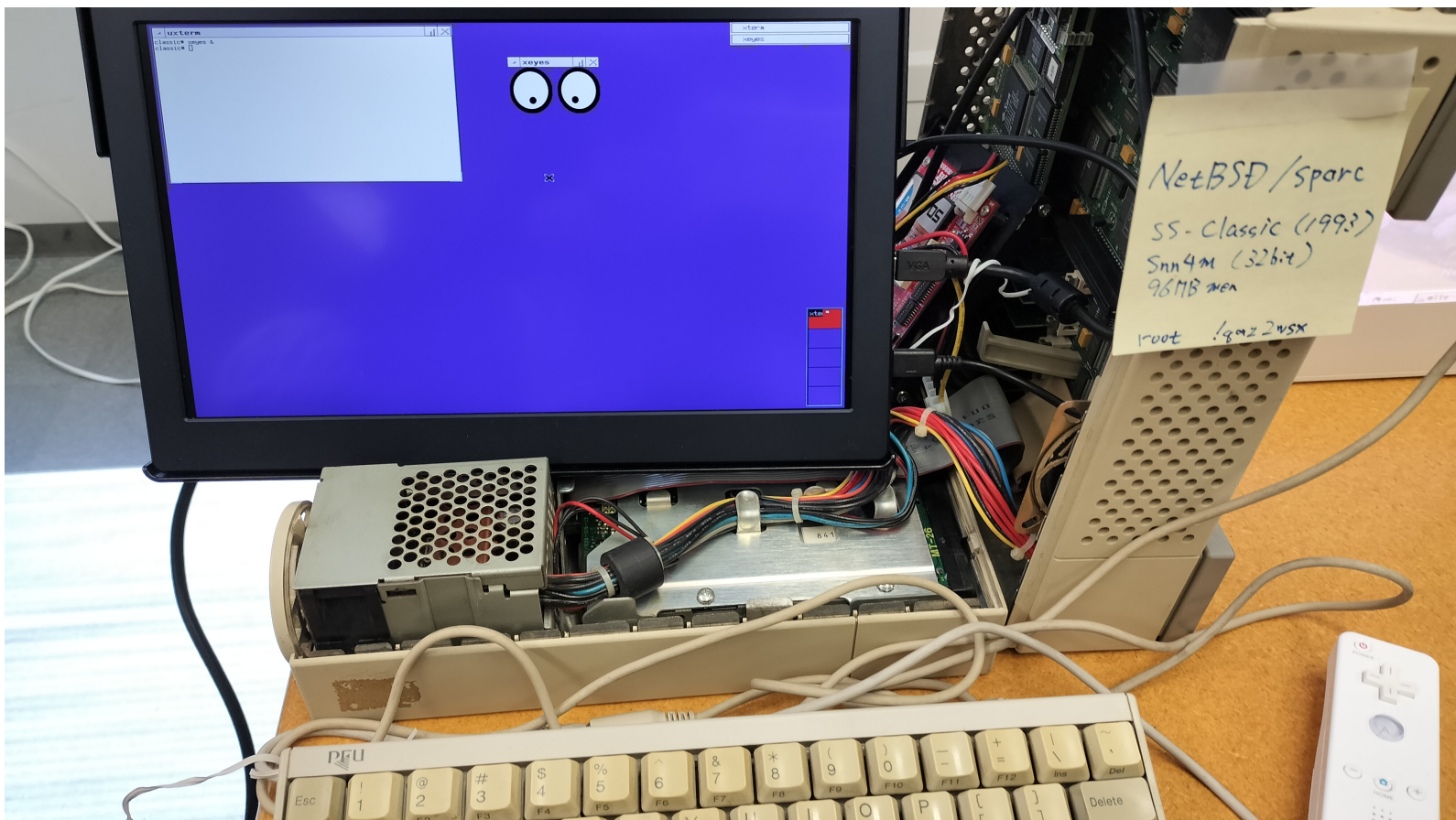


Open Source Conference 観光ガイド

オープンソースカンファレンス 北海道

NetBSD



日本 NetBSD ユーザーグループ

Japan NetBSD Users' Group

2026

1. NetBSD 11.0 RC5 リリースのお知らせ (2026年6月16日)

目次

- NetBSD 11.0 RC5 リリースのお知らせ (2026年6月16日)
 - [せっかちな方へのクイックリンク](#):
 - [はじめに](#)
 - [アップグレード手順](#)
 - NetBSD 10.1 からの変更点
 - [ハイライト](#)
 - [ポート](#)
 - [カーネル](#)
 - [機械独立デバイスドライバ](#)
 - [ユーザ空間](#)
 - [非互換の変更](#)
 - [サードパーティコンポーネント](#)
 - [NetBSD 11.0 RC5 の入手方法](#)
 - [NetBSD 11.0 がサポートするシステムファミリ](#)
 - [謝辞](#)
 - [NetBSD について](#)
 - [NetBSD Foundation について](#)

1.1. せっかちな方へのクイックリンク:

ダウンロード (CDN): * amd64 USB * amd64 DVD * i386 USB * i386 CD-ROM * sparc64 DVD * sparc64 CD-ROM * macppc CD-ROM * RISC-V デバイス * Cavium Octeon MIPS デバイス * Nintendo Wii * 各種 Arm デバイス * その他全てのリリースディレクトリ

配布物内の全ファイルの ハッシュ値 は、NetBSDセキュリティオフィサーのPGP鍵 で署名されています。

1.2. はじめに

NetBSDプロジェクトは、NetBSDオペレーティングシステムの第19メジャーリリースの第5リリース候補である、NetBSD 11.0 RC5 をアナウンスします。

1.3. アップグレード手順

既存のインストール環境は、インストールイメージから起動し「Upgrade」オプションを選択することでアップグレード可能です。他の更新方法を使用している場合は、まずカーネルとモジュールを更新し、再起動してからユーザ空間を更新してください。パッケージリポジトリのURLを調整し、すべてのサードパーティ製パッケージを更新する必要があります。また、新しいセットが追加されている点にも注意してください。これらは `sysinst(8)` を使用して個別にインストールする必要がある場合があります。以前のリリースからアップグレードする場合は、「非互換の変更」の項目に特に注意してください。

1.4. NetBSD 10.1 からの変更点

1.4.1. ハイライト

- RISC-Vプロセッサアーキテクチャへの新規ポート。NetBSD 11.0は、VisionFive 2、PINE64 STAR64などのStarFive JH71XXベースのデバイスやQEMUを含む、64ビットRISC-Vプラットフォームをサポートする初の安定版リリースです。
- POSIX.1-2024およびC23プログラミングインターフェース標準への準拠強化。
- `compat_linux(8)` におけるLinuxシステムコールのサポート強化 (`epoll` (kqueue上で実装)、POSIXメッセージキュー、`statx`、`readahead`、`close_range`、`waitid`、`renameat2`、`clone3`、`sync_file_range`、`syncfs`、`inotify`を含む)。
- Qualcomm Snapdragon X Eliteプラットフォームへの初期サポート。
- `npf(7)` ファイアウォールの改善 (レイヤ2フィルタリングおよびユーザ/グループフィルタリングを含む)。
- x86向けの新MICROVMカーネル。i386とamd64の両方をサポート。NetBSD 11.0は、極めて高速な仮想マシンブートのために設計された専用のMICROVMカーネルを導入しました。PVHブート、VirtIO MMIO、および複数のカーネル最適化を活用し、2020年頃のx86 CPU上で約10msでブート可能です。
- 新規virt68kポート。QEMUで準仮想化デバイスを使用してMotorola 68000ポートを動作させます。

1.4.2. ポート

- aarch64 - Qualcomm Oryon CPUへの初期サポート。
- aarch64 - Qualcomm Snapdragon X Elite バッテリー&充電器センサ、GPIO、I2Cコントローラのサポート追加。

- aarch64 - Advanced SIMDでアクセラレートされたX.Org最適化を有効化。
- alpha - Secure PLT ELFバイナリのサポート追加。
- amd64 - 一部のHPノートPCで見られるAMD GPIO用の `amdgp(4)` デバイスドライバを追加。
- amd64 - Siena, Turin Classic, Turin Dense, Strix Point, Zen 5 CPUの温度センササポートを追加。
- evbarm - ACPIベースシステム向けにNUMA認識を追加。
- evbarm - Arm SCMIパフォーマンスプロトコルサポートを追加。
- evbarm - Amlogic Meson I2Cコントローラ用ドライバを追加。
- evbarm - Libre Computer AML-S905X-CCのUSBコントローラをサポート。
- evbarm - Rockchip USB-C PHY用ドライバを追加。
- evbarm - NanoPi R4Sのサポートを改善。
- evbarm - UEFIでブート時のRaspberry Pi 5周辺機器のサポートを改善。
- evbarm - Allwinnerデバイス向けにDRM/KMSアトミックモードセッティングを実装。
- evbppc - Nintendo Wii DVDドライブ、`bwi(4)` WLAN、USB Geckoシリアルコンソール、ハードウェアAESエンジン（ディスク暗号化とWi-Fiの高速化に使用）、およびスタンドアロンブートローダーのサポートを追加。
- evbppc - WiiでのX.Orgパフォーマンスと色再現性を向上（GX RGBからYUY2へのビデオ変換高速化）。
- hp300 - 362および382向けに `sti(4)` フレームバッファサポートを追加。これらのマシンでX.Orgサーバが動作可能に。
- hppa - HP Visualize FXビデオカード用の新しい `summitfb(4)` ドライバ。
- hppa - HP HCRXビデオカード用の新しい `hyperfb(4)` ドライバ。
- hppa - HPのNGLEファミリグラフィックスデバイス専用X.Orgドライバを追加。Visualize EGでのアクセラレーションをサポート。
- m68k - 浮動小数点エミュレーションを改善。
- mac68k - PowerBook 1xxシリーズのサポートを改善（バックライト制御を含む）。新しい `pbbat(4)` バッテリドライバを追加。
- macppc - G5/G4 I2C/OpenPICサポートを改善。
- macppc - PowerMac7,2向けファン制御を有効化。
- macppc - デフォルトでPaX MPROTECTおよびASLRセキュリティ緩和機能を有効化。
- mips - デフォルトでスタック破壊保護を有効化。mips64でRELRO緩和を有効化。
- riscv - QEMUへの初期サポート（Google Goldfish RTC、VirtIO（ネットワーク、ディスク等）デバイス、仮想化オーディオを含む）。
- riscv - Allwinner D1 SoC (MangoPi MQ ProやAllwinner Nezhaなどで使用) への初期基本サポート。GPIOおよびUARTシリアルコンソールサポートを含む。
- riscv - StarFive JH7110 SoC (VisionFive 2、STAR64などで使用) へのサポート。PCI/PCIe、TRNG、ピン、温度センサなどを含む。
- riscv - `crash(8)` カーネルデバッグへのサポート。
- x86 - `pv(4)` 擬似バス経由のカーネルコマンドラインパラメータで処理される、MMIO上のVirtIOデバイス検出をサポート。
- x86 - `pci(4)` のような既知のバスに接続されない全準仮想化デバイス向けに、`pv(4)` 擬似バスのサポートを追加。
- x86 - 非Xen PVHブート（例：QEMU `-kernel` オプション）をサポート。
- x86 - `tco(4)` において新しいIntel TCOウォッチドッグタイマーをサポート。
- x86 - サポートシステム上で充電動作の一部を制御するための `thinkpad(4)` `sysctl` インターフェイスを公開。
- x68k - フレームバッファコンソールへのSixelグラフィックスシーケンスサポートを追加。

1.4.3. カーネル

- カーネル - 最適化：各種ポートでソフト割り込みをトリガする際の不要なメモリバリアを省略。
- カーネル - 最適化：`vdrain`カーネルスレッドを2つのスレッドプールジョブに置き換え。
- カーネル - 最適化：SPARC、PowerPC、MIPS上でのバイトスワップ（特にAdiantumディスク暗号化向け）のパフォーマンスを改善。
- カーネル - `0_CLOFORK`実装を追加。
- `procfs` - Linux互換性のために`sysvipc`および`self/limits`のサポートを追加。
- `zfs` - `rm/unlink`のパフォーマンスを改善。
- `zfs` - Xenで使用される`zvol`のDIOCCACHESYNCを実装。
- `clock_getres(2)` - `CLOCK_PROCESS_CPUTIME_ID`および`CLOCK_THREAD_CPUTIME_ID`をサポート。
- `memfd_create(2)` - 新しいシステムコール。同名のLinuxシステムコールと互換性あり。
- `pipe(2)` - パフォーマンス向上、ロック競合の削減。
- `ddb(4)` - カーネル内デバッグに単純な`show`コマンド（`condvar`, `selinfo`, `sleepq`）を追加。
- `dk(4)` - Atari TOSパーティションをディスクウェッジとして検出するサポートを追加。
- `drum(4)` - サブシステムをMPSAFE化。
- `nvmm(4)` - CPUIDにおけるVMware互換のTSCおよびLAPIC周波数検出を実装。
- `wscons(4)` - VT100エミュレーションに輝度および`xterm`スタイルの256色エスケープコードのサポートを追加。
- `wscons(4)` - 現在のフォントに関する情報を取得するための`WSDISPLAYIO_GFONT ioctl`を追加。
- `npf(7)` - レイヤ2フィルタリングサポートを追加。
- `npf(7)` - ユーザ/グループベースのフィルタリングサポートを追加。
- `heartbeat(9)` - カーネルの進捗を確認するための新しいメカニズム。ハード割り込みを使用して低優先度のソフト割り込みの進捗を、あるCPUが別のCPUの進捗をチェックするために使用されます。

1.4.4. 機械独立デバイスドライバ

- `aac(4)` - Adaptec RAID 5445, 5805, 5085のサポートを追加。
- `apei(4)` - ACPI Platform Error Interfaces用ドライバ。
- `awge(4)` - NET_MPSAFEカーネルオプションによる詳細なロックを改善。
- `axen(4)` - ASIX AX88179A USB Ethernetのサポートを追加。
- `bwi(4)` - 不要な遅延を削除。
- `ds2482ow(4)` - Maxim DS2482-100およびDS2482-800 I2C-1-Wireブリッジチップ用ドライバ。
- `ds28e17iic(4)` - DS28E17 1-Wire-I2Cブリッジチップ用ドライバ。
- `dse(4)` - DaynaPORT SCSI/Link Ethernetデバイス用ドライバ。これらのレガシーデバイスは、PiSCSIソフトウェアを動かすRaSCSIボードを搭載したRaspberry Piで現在エミュレーション可能です。
- `emcfan(4)` - Microchip Technology / SMC EMC210XおよびEMC230Xファンコントローラ用ドライバ。
- `gftty(4)` / `gfpic(4)` - "Goldfish" 仮想ハードウェアプラットフォーム用ドライバ。
- `gscan(4)` - USB-CANバスアダプタ用ドライバ。
- `igc(4)` - TCP Segmentation Offload (TSO) のサポートを追加。
- `ikbd(4)` - I2C HIDキーボードドライバ。
- `lm(4)` - NCT6794D向けに5つの補助ファンセンサをサポート。
- `mcx(4)` - NVIDIA Mellanox ConnectX-6仮想関数のサポートを追加。
- `ncm(4)` - USB Network Control Model用ドライバ (新しいAndroidのUSBテザリングで使用)。
- `pms(4)` - PS/2マウスのホットプラグサポートを追加。
- `puc(4)` - BrainboxesおよびOxford Semiconductor PCIシリアルカードのサポートを追加。
- `pvscsi(4)` - VMware準仮想化SCSIコントローラ用ドライバ。
- `rge(4)` - Realtek RTL8126 5Gbps Ethernetのサポートを追加。
- `u3g(4)` - Sierra Wireless MC7304 LTEモデムをサポート。
- `uaudio(4)` - USB Audio Class 2.0互換デバイスをサポート。
- `udl(4)` - SANWA SUPPLY 500-KC002N USB-VGAアダプタのサポートを追加。
- `uftdi(4)` - ドライバが個別のインターフェースを選択的に拒否し、`ugen(4)` でマッチさせてlibusb経由でアクセス可能にする「マッチクォーク」メカニズムを追加。
- `umcpmio(4)` - MCP-2221 / 2221AマルチIOチップ用ドライバ。
- `urndis(4)` - 追加のデバイスタイプをマッチ対象に。
- `urtnw(4)` - Asus USB-N10 Nano B1 Wi-Fiアダプタのサポートを追加。
- `viaide(4)` - 各種VIA IDE/PATA/SATAコントローラのサポートを改善 (特にRAIDモード)。
- `viogpu(4)` - QEMU仮想GPU用ドライバ。
- `virtio(4)` - VirtIOドライバをMPSAFEとマークし、巨大なカーネルロックを保持しないように変更。

1.4.5. ユーザ空間

- `build.sh` - 新ターゲット `pkg=CATEGORY/PACKAGE`。 `pkgsrc` から `CATEGORY/PACKAGE` をクロスビルドし、`pkgsrc` をブートストラップします。
- `libc` - `stdio FILE` ハンドル経由の書き込みパフォーマンスを向上。
- `libc` - `ffsl(3)`, `ffsll(3)`, `mempcpy(3)`, `wmempcpy(3)`, `heapsort_r(3)`, `mergesort_r(3)`, `qsort_r(3)`, `c8rtomb(3)`, `mbrtoc8(3)`, `timespec_getres(3)` 関数を追加。
- `libm` - より多くの `long double` 関数を追加。テストを拡充。
- `libpthread` - 不足していたPOSIX.1-2024キャンセルポイントを監査。
- `aiomixer(1)` - 環境変数 `NO_COLOR` を設定して色を無効にするという非公式標準をサポート。
- `c17(1)` - POSIX.1-2024で要求されるISO 2017 Cコンパイラ用の新しいラッパースクリプト。
- `calendar(1)` - 2026年の移動祝日データを更新。
- `crunchgen(1)` - ライブラリディレクトリを設定するための `-L` オプションを尊重。
- `cut(1)` - `-b` オプション使用時にマルチバイト文字を分割しない `-n` オプションを追加。
- `date(1)` - GNU `date` 同様、RFC 5322形式で時刻を表示する `-R` オプションを追加。
- `df(1)` - `-M` (非マウント引数を無視) および `-q` (警告を抑制) オプションを追加。
- `ftp(1)` - ソケットバッファ処理を簡素化しパフォーマンスを向上。
- `gzip(1)` - GNU互換のための `--ascii` および `--license` オプションを追加。
- `install(1)` - `-v` (verbose) オプションを追加。
- `lint(1)` - C23コードのチェックをサポート。
- `lint(1)` - コード内のより多くの整数オーバーフロー型を検出。
- `make(1)` - インクルードガードの認識、`:M`および`:N`によるパターンマッチングの改善など、各種パフォーマンス向上。
- `man(1)` - ローカルパスをロードする `-l` フラグを追加。
- `patch(1)` - GNU `patch`互換のための `--backup-if-mismatch` および `--no-backup-if-mismatch` を追加。
- `patch(1)` - `INT16_MAX` を超える長さの行を処理。
- `pkill(1)` - `pidfile` を使用する `-F` オプション、およびロックを確実にする `-L` を追加。
- `printf(1)` - 変換用無効値の検出と診断を改善。
- `printf(1)` - `%C`形式変換と、`long double` を使用する `-L` オプションを追加。
- `sh(1)` - POSIX Issue 8の要件に準拠するようチルダ展開を調整。
- `sh(1)` - シェル入力内のNUL文字を拒否するように変更。
- `sh(1)` - バージョン情報を表示する `-r` オプションを追加。

- `sh(1)` - `read`組み込みコマンドに `-b` および `-nMAX` オプションを追加。
- `sh(1)` - `csh(1)` 組み込みコマンド同様、`suspend`を組み込みみとして実装。
- `sh(1)` - `HISTFILE`および`HISTAPPEND`変数を実装。
- `stat(1)` - シンボリックフラグの出力を追加。
- `touch(1)` - `-d`, `-R`, `-D` オプションを追加。
- `videoctl(1)` - フレームサイズの列挙を改善。
- `curses(3)` - `ncurses`拡張 `wgetscrreg(3)` を実装。
- `propLib(3)` - JSONシリアライズ形式のサポートを追加。
- `cribbage(6)` - 尋ねずにプレイし続ける `-y` オプションを追加。
- `tetris(6)` - 環境変数 `NO_COLOR` を設定して色を無効にするという非公式標準をサポート。
- `worms(6)` - ワームをカラー表示する `-C` オプション、ワームの頭のバリエーションを増やす `-H` オプションを追加。
- `blocklistd(8)` - `/etc/blocklistd.conf.d` のような設定ディレクトリ内の複数の設定ファイルをサポート。
- `chown(8)` - ファイルの所有者/グループを現在値に変更しないための `-d` フラグを追加。不必要な操作を回避。
- `cpuctl(8)` - Intel Meteor LakeおよびEmerald Rapids CPUを認識。
- `dkctl(8)` - 新しい`getgeometry`コマンドを追加。
- `envstat(8)` - JSON出力をサポート。
- `gpt(8)` - GUID管理の追加、開始/サイズを16進数で表示するオプション。
- `iostat(8)` - 限定的な代替統計である `-X` オプションを追加。"`-D`" オプションに似ているが、`xfers/s`をスキップし、`KB/s`ではなく`MB/s`を報告。デフォルトで1行あたり約50%多くのデバイスを表示可能に。
- `makefs(8)` - `-m` で`cd9660`ファイルシステムのサイズ制限を可能にし、互換ブートROM用に`version=0 4.3BSD FFS`ファイルシステムの作成を許可。
- `mount_cd9660(8)` - `mask`, `dirmask`, `uid`, `gid` マウントオプションのサポートを追加。
- `scsictl(8)` - デバイスを識別するVital Product Dataを報告する`identify vpd`を追加。
- `syslogd(8)` - リモートメッセージの "`kern`" から "`user`" への変換を無効にする `-k` オプションを追加。

1.4.6. 非互換の変更

- 本リリースの新しいOpenSSHは、DSA鍵をサポートしなくなりました。これらを有効にしているカスタム設定がある場合、`sshd`の起動が失敗する可能性があります。設定を適宜調整してください。
- 互換ライブラリ (64ビットプラットフォーム上の32ビットバイナリ、MIPS上の64ビットバイナリ用) は、新しい`base32/base64`、`debug32/debug64`セットに分割されました。
- HTMLマニュアルページは「`man`」セットから新しい「`manhtml`」セットに分割されました。
- `aarch64`でのCPU周波数を制御する`sysctl`インターフェースは、パフォーマンス単位ではなくMHzを使用するようになりました。
- `hdaudio(4)` ドライバは、ミキサーコントロールのソートと命名が変更されました。スクリプトを更新する際は注意してください (特に、入力モニター (以前は "`record`" と呼ばれていたもの) やビープ音量を調整していた場合)。
- `libc - ctype(3)` 関数にガードページを追加し、実行時におけるこのAPIの一般的な誤用を検出。以前誤った結果を返していたコードは、現在セグメンテーションフォールトを起こします。以前の動作に戻すには、環境変数 `LIBC_ALLOWCTYPEABUSE` を設定してください。
- `i386` - `dosboot`からXMS固有のコードを削除。
- `heimdall` - `sqlite3`資格情報キャッシュ (SCC) を無効化し、`sqlite3`依存関係を削除。
- `Xorg(1)` - `libXxf86misc`ライブラリを削除。この拡張のサポートは2008年にX.Orgから削除されており、どのXサーバもサポートしていません。
- `cp(1)` - ゼロサイズに見える場合でも、常に通常ファイルをコピーします。
- `expr(1)` - `":"` および `"length"` においてバイト単位ではなくマルチバイトコードポイントを使用 (POSIXで要求)。
- `xfwp(1)` - 削除。
- `mkstr(1)`, `xstr(1)` - 削除 (PDP-11時代の廃れたツール)。
- `curses(3)` - `ncurses`互換のため `define_key(3)` の引数をconst化。
- `curses(3)` - `newterm(3)`、`setterm(3)` 関数の引数をconst化。
- `lagg(4)` - `lagg`に追加されたインターフェースへ`lagg`のMTUをコピー。
- `ppp(4)` - 機能していない複数の`ioctl`を削除。
- `moused(8)` - 未公開のCオプションを削除。
- `mount(8)` - `mount -v spec fs` で出力される情報を削減。以前の出力には `-vv` を使用してください。

1.4.7. サードパーティコンポーネント

- `binutils` - 2.42に更新。
- `expat` - 2.8.1に更新。
- `openresolv` - 3.16.3に更新。
- `pam-u2f` - 1.3.1に更新。
- `tzdata`, `tzcode` - 2026b (2026bgtzを使用) および `tzcode2025b`に更新。
- `userspace-rsu` - 0.15.0に更新。
- `Xorg(1)` - 21.1.22に更新。
- `awk(1)` - one true `awk` 20240817に更新。
- `ctwm(1)` - 4.1.0に更新。
- `gcc(1)` - 12.5.0に更新。

- gdb(1) - 15.1に更新。
- less(1) - 643に更新。
- mandoc(1) - 1.14.6に更新。
- openssl(1) - 3.5.7に更新。
- pkg_add(1) - pkg_install-20250417に更新。
- postfix(1) - 3.11.2に更新。
- ssh(1) / sshd(8) - 10.3に更新。
- sqlite3(1) - 3.45.1に更新。
- tmux(1) - 3.6bに更新。
- yacc(1) - byacc-20240109に更新。
- zstd(1) - 1.5.6に更新。
- xz(1) - 5.8.3に更新。
- jemalloc(3) - 5.3.0に更新 (sun2およびvaxを除く)。
- libarchive(3) - 3.7.7に更新。
- pcap(3) - 1.10.5に更新。
- openpam(3) - 20230627に更新。
- terminfo(3) - 20231209に更新。
- zlib(3) - 1.3.1に更新。
- acpi(4) - ACPICAを20241212に更新。
- dhcpcd(8) - 10.2.3に更新。
- mdnsd(8) - 2559.1.1に更新。
- named(8) - 9.20.22に更新。
- ntpd(8) - 4.2.8p18に更新。
- pppd(8) - 2.5.2に更新。
- tcpdump(8) - 4.99.5に更新。
- unbound(8) - 1.25.1に更新。
- wpa_supplicant(8) / hostapd(8) - 2.11に更新。

完全な変更リストは、NetBSD 11.0リリースツリーのトップレベルディレクトリにある [CHANGES](#) および [CHANGES-11.0](#) ファイルで確認できます。

1.5. NetBSD 11.0 RC5 の入手方法

NetBSD 11.0 RC5の完全なソースとバイナリは、世界中の多くのサイトからダウンロード可能です。[メインのCDN](#) からダウンロードするか、お近くの [ミラーサイト](#) を使用してください。NetBSD 11.0 RC5配布物のハッシュリストは、NetBSDセキュリティオフィサーのPGP鍵で署名されており、[このファイル](#) で入手できます。NetBSDは無料です。すべてのコードは制限のないライセンスの下にあり、誰にロイヤリティを支払うことなく使用可能です。無料のサポートサービスはメーリングリストやウェブサイト経由で利用可能です。商用サポートは様々なソースから提供されています。NetBSDに関するより広範な情報は、当社のウェブサイト www.NetBSD.org を参照してください。

1.6. NetBSD 11.0 がサポートするシステムファミリ

NetBSD 11.0リリースは、以下のシステムに対してサポートされたバイナリ配布物を提供します：

- NetBSD/aarch64 Arm 64-bit
- NetBSD/acorn32 Acorn RiscPC/A7000, VLSI RC7500
- NetBSD/algor Algorithmics, Ltd. MIPS評価ボード
- NetBSD/alpha Digital/Compaq Alpha (64-bit)
- NetBSD/amd64 Opteron, Athlon64などのAMDファミリプロセッサ、およびEM64T拡張を備えたIntel CPU
- NetBSD/amiga Commodore AmigaおよびMacroSystem DraCo
- NetBSD/amigappc PowerPCベースAmigaボード
- NetBSD/arc Advanced RISC Computing仕様に従うMIPSベースマシン
- NetBSD/atari Atari TT030, Falcon, Hades
- NetBSD/bebox Be Inc. BeBox
- NetBSD/cats Chalice Technology CATSおよびIntel EBSA-285評価ボード
- NetBSD/cesfic CES FIC8234 VMEプロセッサボード
- NetBSD/cobalt Cobalt Networks MIPSベースマイクロサーバ
- NetBSD/dreamcast セガ ドリームキャスト
- NetBSD/emips Microsoft Research Extensible MIPSアーキテクチャ
- NetBSD/epoc32 Psion EPOC PDA
- NetBSD/evbarm 各種Armベース評価ボードおよびアプライアンス
- NetBSD/evbmips 各種MIPSベース評価ボードおよびアプライアンス
- NetBSD/evbppc 各種PowerPCベース評価ボードおよびアプライアンス
- NetBSD/evbsh3 各種Hitachi Super-H SH3およびSH4ベース評価ボードおよびアプライアンス
- NetBSD/ews4800mips NEC MIPSベースEWS4800ワークステーション
- NetBSD/hp300 Hewlett-Packard 9000/300および400シリーズ
- NetBSD/hpcarm StrongArmベースWindows CE PDAマシン

- NetBSD/hpcmips MIPSベースWindows CE PDAマシン
- NetBSD/hpcsh Hitachi Super-HベースWindows CE PDAマシン
- NetBSD/hppa Hewlett-Packard 9000 Series 700ワークステーション
- NetBSD/i386 i486ファミリプロセッサ以降を搭載したIBM PCおよびPCクローン
- NetBSD/ibmws IBM Network Station 1000
- NetBSD/iyonix Castle Technology Iyonix ArmベースPC
- NetBSD/landisk SH4プロセッサベースNASアライアンス
- NetBSD/luna68k OMRON Tateisi Electric LUNAシリーズ
- NetBSD/mac68k Motorola 68k CPU搭載Apple Macintosh
- NetBSD/macppc Apple PowerPCベースMacintoshおよびクローン
- NetBSD/mipsco MIPS Computer Systems Inc.ファミリのワークステーションおよびサーバ
- NetBSD/mmeye Brains mmEyeマルチメディアサーバ
- NetBSD/mvme68k Motorola MVME 68kシングルボードコンピュータ
- NetBSD/mvmeppc Motorola PowerPC VMEシングルボードコンピュータ
- NetBSD/netwinder StrongArmベースNetWinderマシン
- NetBSD/news68k Sony 68kベース「NET WORK STATION」シリーズ
- NetBSD/newsmips Sony MIPSベース「NET WORK STATION」シリーズ
- NetBSD/next68k NeXT 68k「ブラック」ハードウェア
- NetBSD/ofppc OpenFirmware PowerPCマシン
- NetBSD/pmax Digital MIPSベースDECstationsおよびDECsystems
- NetBSD/prep PReP (PowerPC Reference Platform) およびCHRPマシン
- NetBSD/riscv RISC-Vベースデバイス
- NetBSD/rs6000 IBM RS/6000 MCAベースPowerPCマシン
- NetBSD/sandpoint Motorola Sandpointリファレンスプラットフォーム (多くのPPCベースNASボックスを含む)
- NetBSD/sgimips Silicon Graphics MIPSベースワークステーション
- NetBSD/shark Digital DNARD ("shark")
- NetBSD/sparc Sun SPARC (32-bit) およびUltraSPARC (32-bitモード)
- NetBSD/sparc64 Sun UltraSPARC (ネイティブ64-bitモード)
- NetBSD/sun2 Motorola 68010 CPU搭載Sun Microsystems Sun 2マシン
- NetBSD/sun3 Motorola 68020および030ベースSun 3および3xマシン
- NetBSD/vax Digital VAX
- NetBSD/x68k Sharp X680x0シリーズ
- NetBSD/xen Xen仮想マシンモクタ
- NetBSD/zaurus Sharp Arm PDA

リリースに含まれるが、完全にはサポートされていない、または機能しないポート:

- NetBSD/ia64 Itaniumファミリプロセッサ

1.7. 謝辞

NetBSD Foundationは、長年にわたりコード、ハードウェア、ドキュメント、資金、サーバのコロケーション、ウェブページやその他のドキュメント、リリースエンジニアリング、その他のリソースに貢献してくださったすべての方々へ感謝いたします。NetBSDを実現する人々に関する詳細情報は以下で入手可能です: www.NetBSD.org/people/

1.8. NetBSD について

NetBSDは、自由で高速、安全で移植性の高いUnixライクなオープンソースオペレーティングシステムです。大規模なサーバや強力なデスクトップシステムから、ハンドヘルドデバイスや組み込みデバイスまで、幅広いプラットフォームで利用可能です。そのクリーンな設計と高度な機能により、本番環境と研究環境の両方で優れた能力を発揮し、ソースコードはビジネスフレンドリーなライセンスの下で自由に利用可能です。NetBSDは、大規模で活発な国際的コミュニティによって開発・サポートされています。多くのアプリケーションは [pkgsrc](#), [the NetBSD Packages Collection](#) を通じて容易に入手可能です。

1.9. NetBSD Foundation について

[NetBSD Foundation](#) は1995年に設立され、NetBSDプロジェクトのコアサービスの監督、業界およびオープンソースコミュニティ内でのプロジェクトの推進、NetBSDコードベースの大部分に対する知的財産権の保持を任務としています。プロジェクトの日常的な運営はボランティアによって行われています。商業的後ろ盾のない非営利団体として、NetBSD Foundationはユーザからの寄付に依存しています。私たちの優れたオペレーティングシステムの継続的な制作を支援するために、[寄付](#) を検討していただければ幸いです。寛大なご寄付は、継続的なアップグレードやメンテナンス、NetBSD Foundationの運営費を支援するために特に歓迎されます。寄付はPayPalから paypal@NetBSD.org へ、またはGoogle Checkout経由で行うことができ、米国では全額所得税控除の対象となります。詳細については www.NetBSD.org/donations/ を参照するか、finance-exec@NetBSD.org へ直接ご連絡ください。

出典: [Announcing NetBSD 11.0 RC5 (Jun 16, 2026)](<https://www.netbsd.org/releases/formal-11/NetBSD-11.0.html>)

2. pkgsrc-2026Q1のリリース

クロスプラットフォーム・パッケージングシステムである pkgsrc の90回目の四半期リリースとなる pkgsrc-2026Q1 ブランチのリリースに関するアナウンスです。このアップデート時点で、pkgsrcには様々なオペレーティングシステムで利用可能な 29,000以上のパッケージが含まれています。

2.1. ハイライトと主な変更点

pkgsrc-2025Q4のリリース以降、開発者は以下の変更を行いました：

- パッケージ統計：
 - 141の新規パッケージが追加されました。
 - 2,793のパッケージが更新されました。
 - PHP 8.1やFirefox 115など、154のパッケージが削除されました。
- 主要な新規パッケージ：
 - Wine5
 - Qt6 QtWebEngine
 - 複数のKDE Plasma 6パッケージ
- 言語サポートの更新：
 - Ruby 4.0
 - PHP 8.5
 - Lua 5.5
- その他の注目すべきツール：
 - reovimテキストエディタ
 - xtopシステムモニター
 - sddmディスプレイマネージャ

2.2. GnuPGに関する変更

- GnuPGに関する特定の変更として、*gnupg2* は *gpg* バイナリをインストールし、*gnupg1* は *gpg1* バイナリをインストールするようになりました。

2.3. pkgsrc-2026Q1の取得

pkgsrc-2026Q1ブランチは、以下の方法で取得できます：

- CVS
- tarファイル
- 様々なミラー

バイナリパッケージの管理に興味があるユーザーは、pkginウェブサイトですらなる指示を参照してください。

出典：[pkgsrc-2026Q1 released](<https://mail-index.netbsd.org/netbsd-announce/2026/03/27/msg000392.html>)

3. 北海道

マップ：

<https://www.google.co.jp/maps/ms?msa=0&msid=208676479199435389545.0004bf66743aff158ce82>

このドキュメント：

<https://github.com/ebijun/NetBSD/blob/master/Guide/Place/hokkaido.rst>

3.1. よく行く店

枯淡珈琲	南1西13	11:00-18:30L0
めいらあ堂	桑園	札幌の自家焙煎ネルドリップ珈琲の店
まだ名前のない喫茶店	石山通	11:00-19:00 札幌市南区南3 1条西 1 0 丁目 1-1 3
スカイショップ小笠原	新千歳空港	ティンカーベルのオホーツクパーク
俊カフェ	南3西7	11:00-18:00 谷川俊太郎さん公認
MEN-EIJI	平岸2-11	いつもストーンズが流れている

TRANS BREWING BEER STAND	南2条西5丁目6-1	15:00-23:00
Streetlight Brewing	北10西19	12:00-20:00/15:00-21:00
BEER COTAN	南2西3	忽布古丹醸造直営店 12:00/17:00-23:00
BOSSA	狸小路	ジャズ喫茶。M9500+336L。
そらとしば	エスコン	よなよな直営
ハリネズミ珈琲店	南3西8	青りんごソーダとプリンとタンザニアキリマンジャロ。タンノイ。
滝川クラフトビール工房	滝川	泉町1丁目9区。空知ピルスナー。温泉。
おやじ中華ナリケン	狸小路	マコモダケの春巻。
ミンガス	大通32番出口	9:00-24:00 A7。
モエレ沼公園	東区	香川のイサムノグチ庭園美術館とセットで回ろう。
ピアパブひらら	澄川4/南3西5	さいとさんの店。澄川醸造。土曜酒屋営業。Bnewsに詳しい。
ノースアイランド	南2西4	元すみだ川近くに移転。ノースアイランドビール。
モルトヘッズ	南3西8	ビール+モルトウィスキー。オリジナルラブMLとは何か。
Beer cellar sapporo	南1西12	西11の森彦の向かい。原宿/大丸札幌。
アダノンキ	西18丁目	札幌市中央区南1条西19丁目ドレイジャータワー5F
The Craft	南4西4	33タップ 17:00-26:00
梅沢無線	南2西7	部品関連は仙台店に移りました。9:30-17:30 土日祝休
フレッシュエアー	梅沢の隣	Weird MeddleもRecords Recordsもない俺たちは。
MK	札幌市中央区南3西2N・ヴィコロビル2F	PC98/X68000ソフトの聖地だった。ヤフオク pcsshop_mk
Do-夢	東札幌	札幌のアイドル。展示物はここで買いたい。コンベンションセンター近傍
サッポロジンギスカン	南6西6	移転しました。ジンギスカン鍋自動掃除ロボットは必見。
ホールステアーズ	南3西3	いつも一番最後に行く宮越屋。
ブラウンブックスカフェ	南3西7	Cafe と Vintage
イエロー	南1西4	ラムカレーとかピリカワッカのスタウト豚ロースのすき煮風スープカレー
丸大ビル	南2西1	ニャル子さん本拠地のアニメイトとメロンブックス5社連動。
麦酒停	中島公園	壁一面にビール缶が。Rogueとかmikkelerとかある。北海道生活参照。
米風亭	ブラウンブックスカフェ近く	ヒューガルデンと油そばおみやげ。
森彦	円山公園	2階席から眺める 10:00-17:30 L0
ハードオフ札幌伏古店	伏古2条3	テレフンケンのスピーカーがある
ハードオフ札幌川治店	南区川治1	8インチフロッピー
ハードオフ札幌	宮の沢	Lux豊富。Aシリーズとかチャンデバまである。
ハードオフ	南郷20丁目	南郷18丁目駅徒歩4分
ハードオフあいの里	あいの里	あいの里駅徒歩3分
ハードオフ北都	平和	平和
ハードオフ中の島	中の島	
オーム模型	南平岸	ラジコン屋
北へ。に出てきたマンション	平岸→南平岸	歩道橋わたってすぐ。移住するならここしかない
10cc	北26条	トランジスタアンプを高精度で修復する
ウェスタンラボ	新琴似4	アンプ、スピーカーのリペア。310のシールド。
ゲームショップ1983	北35西5	店員さんがルイージ。常にルイージ。通販専門。
坂ビスケット	琴似	坂会館には2階もある。隣はつば八1号店。
キコキコ商店	南12西6	渋さのCDと珈琲。
モスコミュール	宮の杜4条	カレーと夜景とモスコ
シアターキノ	狸小路6南3	2024/6/29から関心領域
赤星	南3西7	500円ラーメン
こぶじ	北2西3	昼ランチ
亜珈里	豊平公園	ザンギ定食大盛900円。朝8:00-
A Table	南7西4	ワインの店。アントニオ猪木さんが吸った葉巻がある地下室
まつくら	増毛	ワグナリア寿司とは何か
国稀酒造	増毛	日本最北の酒蔵
カラハナ	南2条西7	クラフトビール 2021/6/26から19時ラストオーダー。
COOL BEER CRAFT	豊水すすきの1番出口	クラフトビール 12:00-19:00L0
月と太陽	三角市場	クラフトビール 13:00-18:30L0 miredoにもある。
MANDA	狸小路6	世界のビールとピザ 16:00-19:00L0
イケム	南2西5	9時からカフェ/16:00からワインバー。JBLDD66000エベレスト

ギャラリー犬飼	豊平三条一丁目	豊平川近く。粒々研究所とは何か。
かんろ	札幌駅	本店:北8西1→北9西3 北口:北6西6 北一条:北1西8 発寒:5条3 17:00-
CRAFTBEERFOREST	ばんけいスキー場	2021年は中止
北天道書店	北18条西3	日曜定休
六花亭	駅前本店	バターをかうのだ
BUND CAFE	南1条西5	中国茶専門店 12:00-20:00
くるくる寿司	札幌駅APIA地下	11:00-21:30

3.2. 山西さんのカメラ情報

nobugの山西さんに、札幌おすすめカメラ店を教えてくださいました。

[nobug:3970] 市内の中古カメラ店は随分減りました。とりあえずこんなところでしょうか。希望のものがあれば探しておきます。

あすびーず	三越10階	http://asbe-z.co.jp/
カメラのいわもと	白石区本通1丁目南3-7	真空管と真空管式アンプも扱ってる http://iwamoto-camera.com/
カメラ修理センター	北2条東3	http://sapporocamera-repair-center.co.jp/
イシオ	北4西28	http://www.isio28.com/
INHERIT CAMERA	平岸1条9	2024/9開店

3.3. 北海道でのオープンソースカンファレンス

開催年	開催日	参加者	参加グループ	会場
2004 (GODO2004)	7/24			札幌市産業振興センター
2005 in Hokkaido	7/9	250	17	北海道大学 クラーク会館
2006 Hokkaido	7/14	300	24	北海道大学情報科学研究科棟
2007 Hokkaido	6/30	300	22	北海道大学 学術交流会館
2008 Hokkaido	6/28	250	21	札幌産業振興センター
2009 Hokkaido	6/20	370	29	札幌市産業振興センター
2010 Hokkaido	6/26	450	42	札幌市産業振興センター
2011 Hokkaido	6/11	600	27	札幌市産業振興センター
2012 Hokkaido	6/16	650	48	札幌市産業振興センター
2013 Hokkaido	9/14	600	56	札幌コンベンションセンター
2014 Hokkaido	6/13-14	820	53	札幌市産業振興センター
2015 Hokkaido	6/12-13	710	55	札幌コンベンションセンター
2016 Hokkaido	6/17-18	710	57	札幌コンベンションセンター
2017 Hokkaido	7/14-15	580	54	札幌コンベンションセンター
2018 Hokkaido	7/6-7	730	66	札幌コンベンションセンター
2019 Hokkaido	5/31-6/1	720	54	札幌コンベンションセンター
2020 Hokkaido	6/27	450	22	オンライン開催
2021 Hokkaido	6/26	510	22	オンライン開催
2022 Hokkaido	6/25	325	18	オンライン開催
2023 Hokkaido	6/17	225	15	オンライン開催
2023 Hokkaido	6/24	278	30	札幌市産業振興センター
2024 Hokkaido	6/29	400	34	札幌市産業振興センター
2025 Hokkaido	7/5	440	33	札幌市産業振興センター
2026 Hokkaido	6/27			札幌市産業振興センター

3.4. 観光ガイドバックナンバー

これまですべてのバックナンバーは <https://github.com/ebijun/osc-demo> にあります。

No	イベント	URL
228.	OSC2026北海道	http://www.re.soum.co.jp/~jun/OSC2026hokkaido.pdf
213.	OSC2025北海道	http://www.re.soum.co.jp/~jun/OSC2025hokkaido.pdf
199.	OSC2024北海道	http://www.re.soum.co.jp/~jun/OSC2024hokkaido.pdf
187.	OSC2023北海道	http://www.re.soum.co.jp/~jun/OSC2023hokkaido.pdf
177.	OSC2022北海道	http://www.re.soum.co.jp/~jun/OSC2022hokkaido.pdf

166.	OSC2021北海道	http://www.re.soum.co.jp/~jun/OSC2021hokkaido.pdf
155.	OSC2020北海道	http://www.re.soum.co.jp/~jun/OSC2020hokkaido.pdf
142.	OSC2019北海道	http://www.re.soum.co.jp/~jun/OSC2019hokkaido.pdf
126.	OSC2018北海道	http://www.re.soum.co.jp/~jun/OSC2018hokkaido.pdf
109.	OSC2017北海道	http://www.re.soum.co.jp/~jun/OSC2017hokkaido.pdf
89.	OSC2016北海道	http://www.re.soum.co.jp/~jun/OSC2016hokkaido.pdf
72.	OSC2015北海道	http://www.re.soum.co.jp/~jun/OSC2015hokkaido.pdf
58.	OSC2014北海道	http://www.re.soum.co.jp/~jun/OSC2014hokkaido.pdf
47.	OSC2013北海道	http://www.re.soum.co.jp/~jun/OSC2013hokkaido.pdf
26.	OSC2012北海道	http://www.re.soum.co.jp/~jun/OSC2012hokkaido.pdf

3.5. togetherまとめ

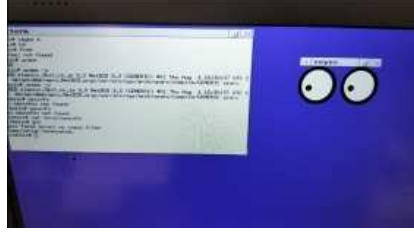
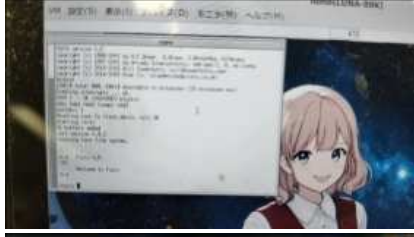
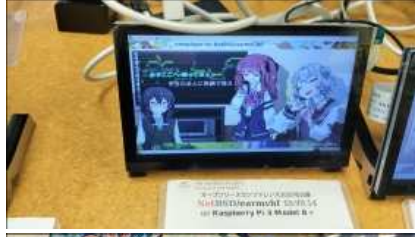
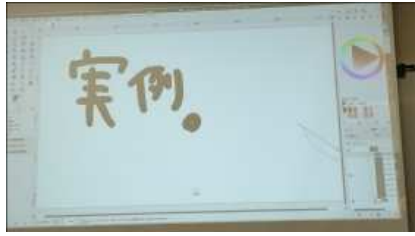
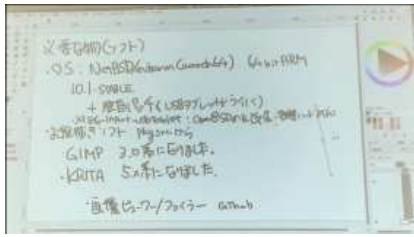
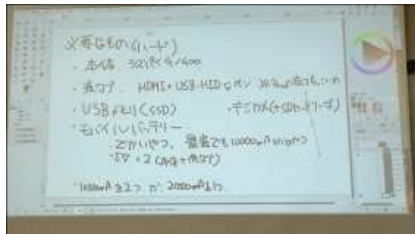
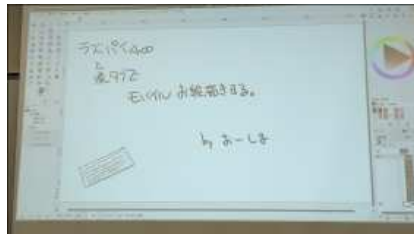
OSC2026北海道	NetBSDブース展示の記録	https://posfie.com/@ebijun/p/3QUHvhw
OSC2025北海道	NetBSDブース展示の記録	https://posfie.com/@ebijun/p/jobKi6a
OSC2024北海道	NetBSDブース展示の記録	https://together.com/li/2377329
OSC2023北海道	NetBSDブース展示の記録	https://together.com/li/2165257
OSC2022北海道	NetBSDブース展示の記録	https://together.com/li/1905889
OSC2021北海道	NetBSDブース展示の記録	https://together.com/li/1735858
OSC2020北海道	NetBSDブース展示の記録	https://together.com/li/1549704
OSC2019北海道	NetBSDブース展示の記録	https://together.com/li/1361738
OSC2018北海道	NetBSDブース展示の記録	https://together.com/li/1244344
OSC2017北海道	NetBSDブース展示の記録	http://together.com/li/1130009
OSC2016北海道	NetBSDブース展示の記録	http://together.com/li/988853
OSC2015北海道	NetBSDブース展示の記録	http://together.com/li/8341488
OSC2014北海道	NetBSDブース展示の記録	http://together.com/li/680053
OSC2013北海道	NetBSDブース展示の記録	http://together.com/li/563719
OSC2012北海道	NetBSDブース展示の記録	http://together.com/li/318806

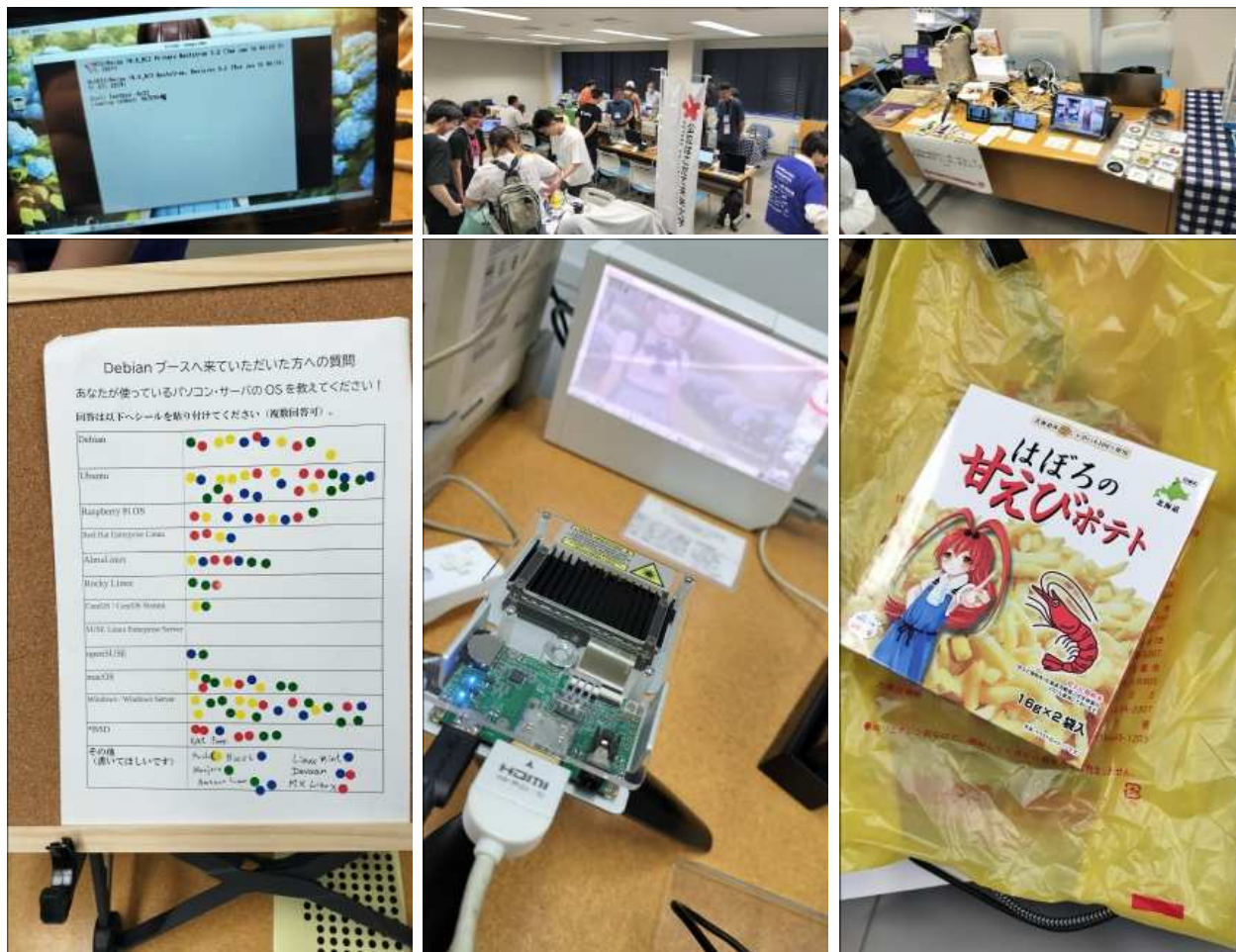
3.6. OSC北海道展示物

1. ドリームキャスト+マラカスコントローラ
2. persona
3. Jornada680/690
4. Jornada710/720
5. VAI0GT+XCAST
6. USL5P+ワンセグチューナー3本
7. BeagleBoard
8. Twitter ticker monitor on 78K0 board via USB Serial
9. ナショナルのクーガー
10. NetBSD/i386 on ThinkPad X200s (VMWarePlayer)
11. FreeNAS on ThinkPad X200s (VMWarePlayer)
12. NetBSD/landisk on USL-5P
13. NetBSD/evbppc on kuro-box
14. NetBSD/evbmips-el on BBR-4MG
15. NetBSD/evbarm on Armadillo-9 + VT220J
16. NetBSD/sparc on Sparcstation IPX
17. MSP430 LaunchPad
18. JMBadgeBoard
19. Processing on NetBSD
20. Building NetBSD on Win7+Cygwin
21. NetBSD/IBM PalmTop110
22. NetBSD/X68k on XM6i
23. mikutter on NetBSD/WZER03
24. NetBSD/Zaurus
25. Making install-image Presentation by つついさん

<http://www.ceres.dti.ne.jp/tsutsui/osc12do/NetBSD-cross-liveimage.html>

3.6.1. 2025年



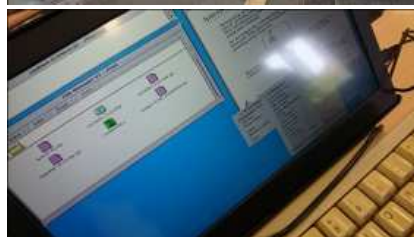


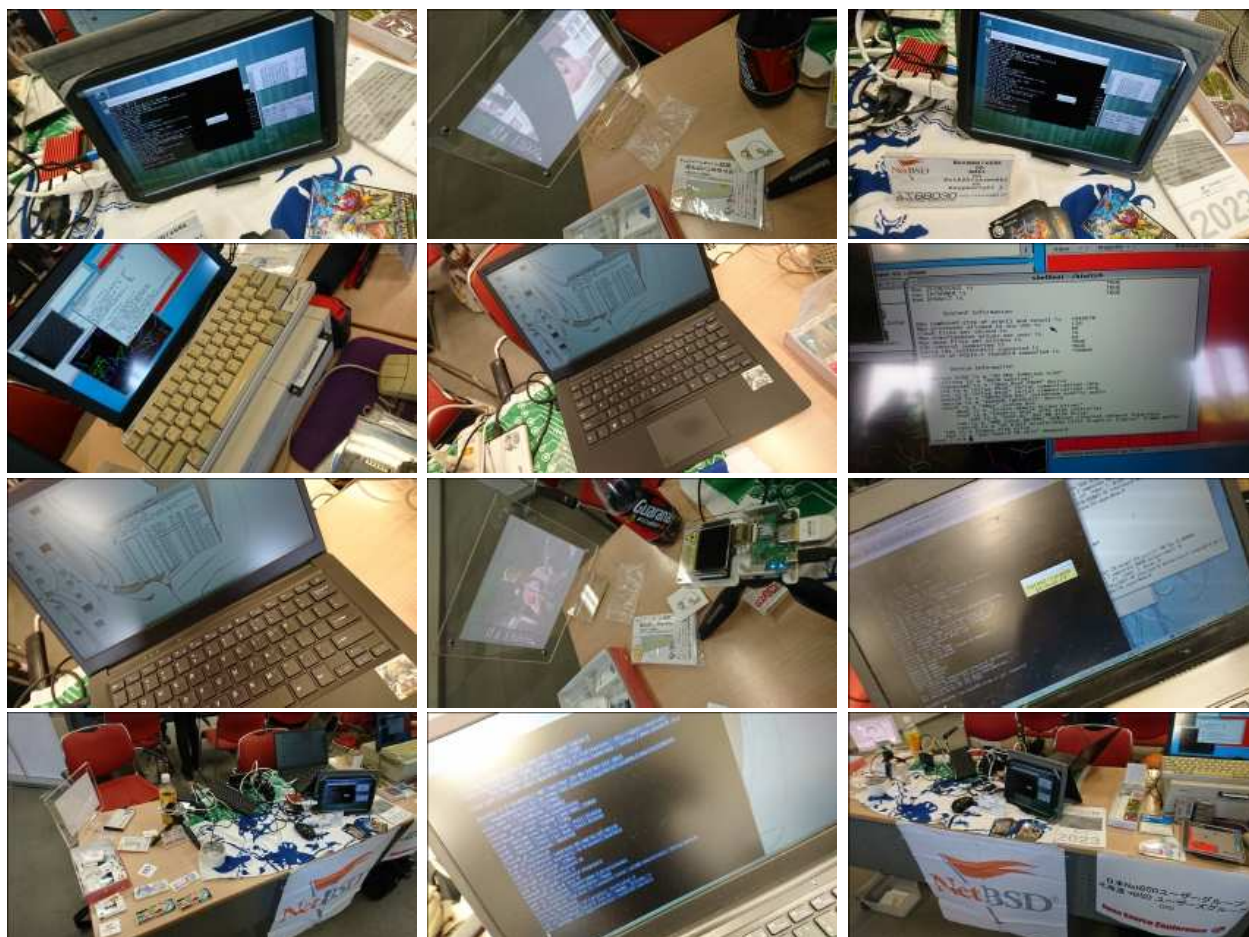
3.6.2. 2024年





3.6.3. 2023年





3.6.4. 2019年





3.6.5. 2018年



3.6.6. 2017年



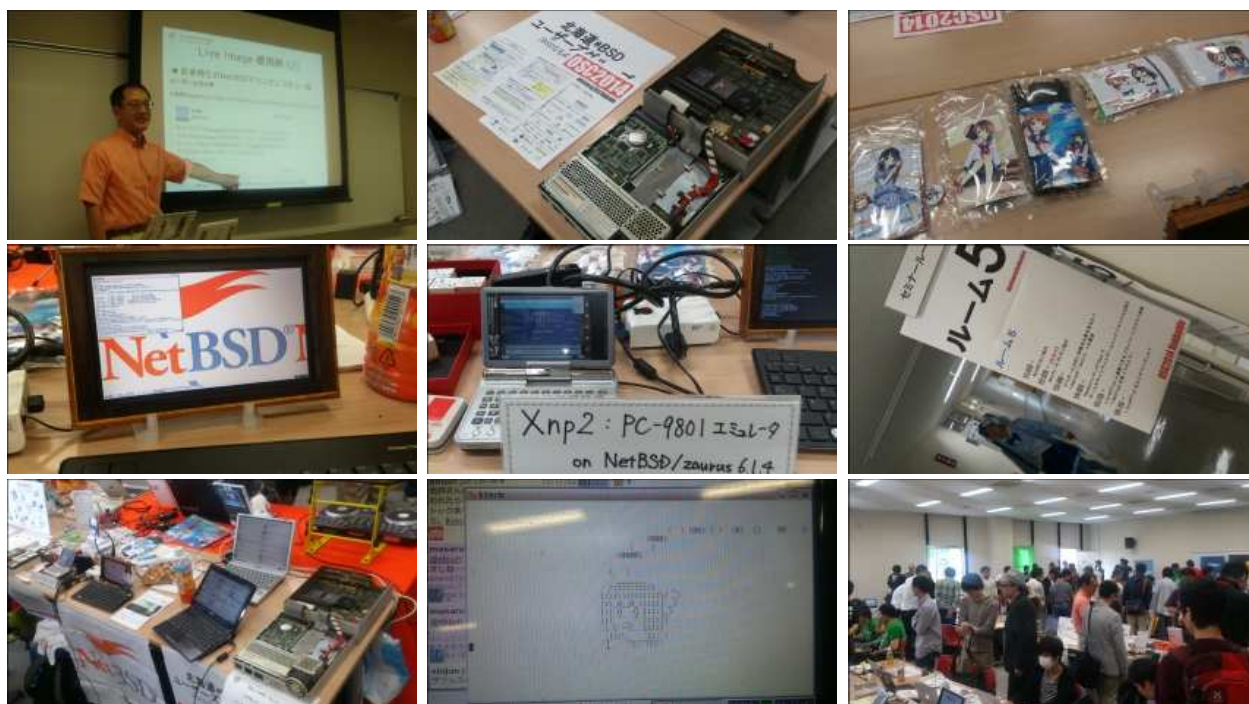
3.6.7. 2016年



3.6.8. 2015年

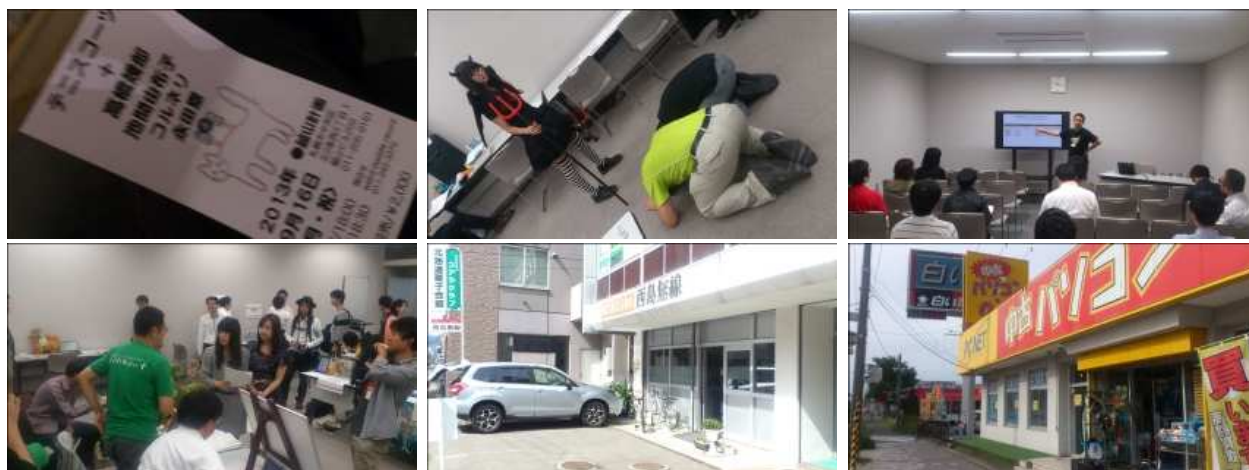


3.6.9. 2014年

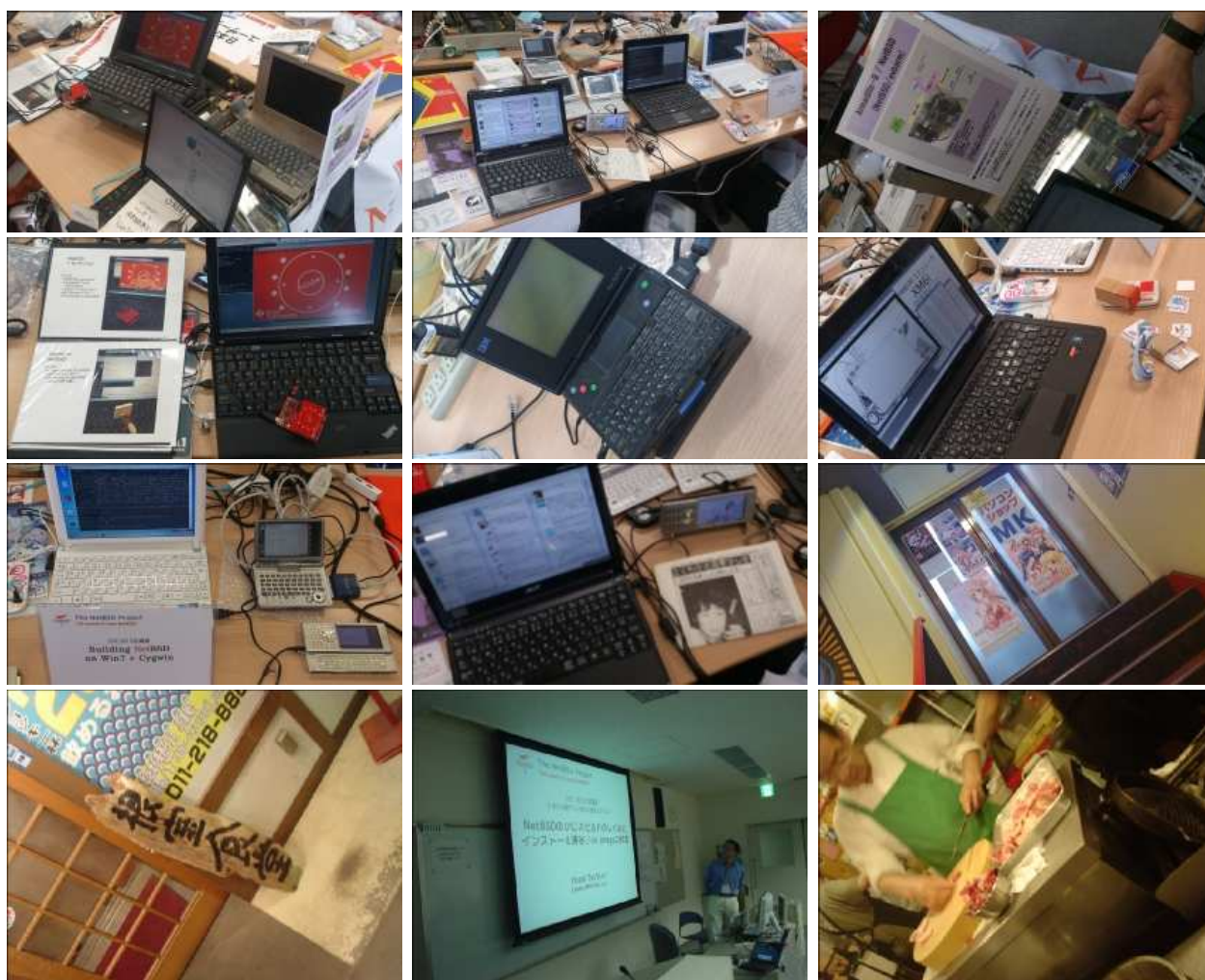


3.6.10. 2013年





3.6.11. 2012年



4. NetBSDの概要

NetBSDは、国際的なコミュニティによって開発されている、完全に**無料**で**オープンソース**のUNIXライクなオペレーティングシステムです。単なる「ディストリビューション」や派生版ではなく、BSDファミリーの完全でユニークなOSとして数十年にわたり進化してきました。

NetBSDは1993年に最初にリリースされ、その品質、クリーンさ、安定性という長い歴史に基づき、多くの意外な環境でそのコードが利用されてきました。コードの起源は、カリフォルニア大学バークレー校の**4.4BSD Lite2**にあります。

4.1. なぜNetBSDを使うのか？

NetBSDのユーザーは、シンプルで文書化が行き届き、完全に統合されたUNIXライクなシステムを享受できます。多くの点で伝統的でありながら、多くの近代的で興味深い機能や最新のハードウェアサポートが含まれています。

主な特徴は以下の通りです。

- セキュリティとメモリの強化機能: PaX MPROTECT (W^X)がデフォルトで全体に適用されるなど、セキュリティ機能が充実しています。ファイル整合性保護には`veriexec`が提供され、従来のBSDの`securelevels`によってスーパーユーザーでさえも操作がさらに制限されます。独自のファイアウォール`NPF`も含まれています。
- 強力なパッケージ管理: NetBSDの`pkgsrc`は、ベースシステムとは独立して四半期ごとの安定ブランチとローリングリリースブランチを持ち、柔軟に組み合わせることができます。`pkgsrc`はNASAを含む高性能科学計算コミュニティで広く採用されています。
- 最新のストレージ機能: `ZFS`ファイルシステム、RAIDframeソフトウェアRAIDシステム、`cgd`ディスク暗号化をサポートしています。
- 広範なARMハードウェアサポート: 単一のカーネルイメージで、オープンで低コスト、ハイエンドな幅広いデバイスをサポートしています。
- 仮想化サポート: Xenや、ハードウェアアクセラレーションを提供するネイティブなカーネルモジュールおよびライブラリ`NVM`ハイパーバイザーが含まれています。
- 移植性 (Portability): 非常に幅広い「レガシー」ハードウェアとABIに対して安定したサポートを継続しており、機能性を損なうことなく、初期のNetBSDリリースとの長期的な後方互換性があります。

4.2. NetBSDプロジェクトの目標

NetBSDプロジェクトは以下の目標を掲げています。

- よく設計され、`安定`し、`高速`なBSDシステムを提供すること。
- ライセンスによる制限を避けること。
- 多くのハードウェアプラットフォームで動作する`ポータブル`なシステムを提供すること。
- 他のシステムとの相互運用性が高いこと。
- 可能な限りオープンシステム標準に準拠すること。

要約: NetBSDプロジェクトは、プロフェッショナル、趣味人、研究者が望むどのような方法でも使用できる、`自由に入手可能で再配布可能なシステム`を提供しています。

4.3. 名前の由来

NetBSDは、オペレーティングシステムを開発するためにネットワークに接続されたバージョン管理システムを使用し、1993年から電子メールでプロジェクトを組織するなど、インターネットを介して完全に共同で組織された最初の主要なオープンソースプロジェクトの1つでした。

- 「Net」: インターネットへの貢献として選ばれました。
- 「BSD」: 4.4BSDおよび386BSDからの派生という、その遺産を認識したものです。

出典: [About NetBSD](<https://www.netbsd.org/about/>)

5. NetBSDがサポートするプラットフォーム (Ports)

NetBSDでは、サポート対象のアーキテクチャを「,ポート (Port),」と呼んでいます。ほとんどのポートは汎用ハードウェアやエミュレータ上で動作しますが、一部の商用ハードウェアも含まれます。

ポートは、その重要性やコミュニティの活動レベルに基づき、以下の3つの「,ティア (Tier),」に分類されます。このティアは、core@netbsd.org の決定により、時間の経過とともに変更されることがあります。

5.1. ティア分類の概要

ティア	名称	サポートの焦点	状態
I	Focus (重点)	NetBSD の戦略の一部としてサポート	現代のサーバー、組み込み、デスクトップアーキテクチャが対象。MI (Machine Independent) な変更は、これらのポートでテストされ、MD (Machine Dependent) なサポートも確実に行われる必要があります。
II	Organic (有機的)	コミュニティのペースで進化	一般的に、ハードウェアは産業的な関連性を失っているか、コミュニティ活動が Tier Iほど活発でないポートが対象。通常は動作していますが、維持管理はユーザーコミュニティの責任です。リリース時に動作しない場合は、Tier IIIに降格します。
III	Life Support (生命維持)	動作しない、または深刻な問題を抱えている	コミュニティの関心の欠如やハードウェアの希少性などにより、機能しなくなったポートが対象。合理的な期間内に修正が示されない場合、Attic (非サポート) に移動します。

5.2. ティア I: Focus (重点ポート)

現在、9つのポートがTier Iステータスです。

Port	CPU	対応マシン	最新リリース
aarch64	aarch64	64-bit ARM CPUs	10.1
amd64	x86_64	64-bit x86-family machines (AMD/Intel)	10.1
evbarm	arm	ARM評価ボード	10.1
evbmips	mips	MIPSベース評価ボード	10.1
evbppc	powerpc	PowerPCベース評価ボード	10.1
hpcarm	arm	StrongARMベース Windows CE PDA	10.1
i386	i386	32-bit x86-family汎用マシン ("PC clones")	10.1
sparc64	sparc	Sun UltraSPARC (64-bit)	10.1
xen	i386 x86_64	Xen Virtual Machine Monitor	10.1

5.3. ティア II: Organic (有機的ポート)

現在、49のポートがTier IIステータスです。

(例: alpha amiga atari macppc pmax sparc vax zaurus など)

5.4. ティア III: Life Support (生命維持ポート)

現在、Tier IIIステータスのポートは、ありません、。

5.5. CPUアーキテクチャ別ポート一覧

同じ `MACHINE_ARCH` (`uname -p`) を持つマシンは、通常同じユーザーランドバイナリを共有します。

CPUティア該当ポート

aarch64	I	aarch64
amd64	I	amd64 xen
arm	I II	acorn32 cats epoc32 evbarm hpcarm iyonix netwinder shark zaurus
i386	I	i386 xen
m68k	II	amiga atari cesfic hp300 luna68k mac68k mvme68k news68k next68k sun3 x68k
mipseb	I II	emips evbmips ews4800mips mipsco newsmips sbmips sgimips
mipsel	I II	algor arc cobalt evbmips hpcmips pmax sbmips
powerpc	I II	amigappc bebox evbppc ibmnws macppc mvmeppc ofppc prep rs6000 sandpoint
sparc64	I	sparc64
vax	II	vax
alpha	II	alpha
hppa	II	hppa
sh3eb/el	II	dreamcast evbsh3 hpcsh landisk mmeye
m68010	II	sun2
riscv	II	riscv (最新リリースなし)
itanium	II	ia64 (最新リリースなし)

5.6. 用語の対照表

コンセプト	<code>/usr/share/mk</code>	変数 <code>uname`</code>	コマンド <code>build.sh`</code>	コマンド
ポート	<code>MACHINE</code>	<code>uname -m</code>	<code>build.sh -m</code>	
CPU	<code>MACHINE_ARCH</code>	<code>uname -p</code>	<code>build.sh -a</code>	

出典:[Platforms supported by NetBSD](<https://wiki.netbsd.org/ports/>)

6. pkgsrcの紹介

pkgsrcは、UNIXライクなシステムでサードパーティ製ソフトウェアを管理するためのフレームワークであり、現在26,000を超えるパッケージを収録しています。

6.1. 概要と特徴

- クロスプラットフォーム: NetBSDとSmartOSのデフォルトのパッケージマネージャーですが、他の多くのUNIXライクなプラットフォームでも利用でき、自由に利用可能なソフトウェアを簡単にビルドできるようにします。
- パッケージ数: 26,000以上のパッケージを含んでいます。
- バイナリパッケージ: ソースからコンパイルすることなく、生成されたバイナリパッケージを使用できます。既存のシステム上のソフトウェアを補完するために簡単に利用できます。
- 柔軟性と設定:
 - 任意のインストールプレフィックス (インストール先ディレクトリ) へのパッケージのビルドをサポートし、非常に汎用性があり、設定可能です。
 - 1台のマシン上で複数のブランチを共存させることができます。
 - ビルドオプションフレームワークや、コンパイラ変換フレームワークなどの高度な機能を備えています。
 - 非特権ユーザーでの利用とインストールもサポートされています。
- リリース: 四半期ごとの安定ブランチ (例: `pkgsrc-2025Q4`) と、最新の開発版であるHEADブランチがあります。

6.2. 利用方法

6.2.1. 動作プラットフォーム (主要な対象)

`pkgsrc`は移植性を重視しており、特に以下のプラットフォームで活発に利用され、多数のアクティブユーザーとビルドが存在します。

- NetBSD
- Solaris / SmartOS / illumos
- Linux
- macOS (Darwin)

6.2.2. バイナリパッケージのインストール (NetBSD)

`pkgin`という高レベルのツールが、他のパッケージマネージャー (`apt`など) のユーザーにも馴染みやすいように設計されています。

```
# PKG_PATH="https://cdn.NetBSD.org/pub/pkgsrc/packages/NetBSD/${uname -p}/${uname -r|cut -f '1 2' -d.}/All/"
# export PKG_PATH
# pkg_add pkgin
# pkgin install zsh nginx vim
```

6.3. ソースパッケージのインストール

1. `pkgsrc`の取得: 安定版ブランチまたはHEADブランチをCVSでチェックアウトするか、HTTPからtarballをダウンロードします。
2. ブートストラップ: NetBSD以外のOSや、デフォルト設定以外のNetBSDでは、まずパッケージ管理ツールをインストールするためにブートストラップ (ビルドとインストール) が必要です。

```
$ cd pkgsrc/bootstrap
$ ./bootstrap --prefix /opt/pkg-2025Q3 --prefer-pkgsrc yes --make-jobs 4
```

3. パッケージのビルド: 依存関係は自動的にダウンロード、ビルド、インストールされます。

```
$ cd pkgsrc/devel/memcached
$ bmake install clean
```

ビルドされたパッケージは、バイナリパッケージと同様に`pkgin`や`pkg_info`などのツールで管理できます。

出典: [pkgsrc - The NetBSD Packages Collection](<https://www.pkgsrc.org/>)

7. NetBSDコミットガイドライン

以下は、**NetBSD Commit Guidelines**に基づき、ソースツリーへのコミットを行う際のプロジェクトの基準をまとめたものです。

1. コミット対象の制限

- 慣れているコードのみをコミットする：
 - コミットする予定のコードが適切であるか確信が持てない場合（例：問題報告で提出されたコードを採用する場合）、そのシステム部分に詳しい開発者にレビューを依頼してください。
 - プロジェクトに参加したばかりの場合は、スポンサーに確認してください。
- 汚染されたコード (Tainted Code) をコミットしない：
 - 自分で書いたものではないコードをコミットする場合、そのライセンスがNetBSDソースリポジトリへのインポートと自由な再配布を許可していることを二重に確認してください。
 - 大規模言語モデル (LLM) やCopilot、**ChatGPT**などの技術によって生成されたコードは、汚染されたコードと推定され、**core**からの事前の書面による承認なしにコミットしてはなりません。
- 外部ツリーからのコードをコミットしない：
 - `cvs.NetBSD.org` からチェックアウトしたものの以外のツリーからチェックアウトされたコードをリポジトリにコミットしないでください。

2. 事前の承認レベル

変更が侵襲的（影響範囲が広い）であるほど、より高いレベルの事前の承認が必要です。

変更の性質	必要とされる承認レベル
明白な修正 (Obvious Fixes)	事前の議論やレビューは不要。（「明白」の定義は「誰にも反対されない可能性が高い修正」）
その他の修正 (Non-obvious Fixes)	レビューが必要です。
(重要な) 新機能の実装	適切な技術メーリングリストでの事前の議論が必要です。
全く新しいパッケージの追加	メーリングリストでの事前の議論と`core`からの承認が必要です。

3. テストの義務

- テストしたコードのみをコミットする：
 - 変更によって影響を受けるコードが、システムのツールでコンパイルおよび実行され、期待通りに動作することを確認してください。
 - マニュアルページを変更した場合は、`groff/\nroff` が期待する形式のマニュアルページを作成することを確認してください。
- ブランチへのプルアップ：
 - ブランチへのプルアップを要求する前に、それぞれのリリースブランチで変更をテストしてください。
 - `/usr/tests` 内のすべての関連テスト、または理想的にはテストスイート全体を実行してください。
- リグレッションの防止：
 - 長期的なリグレッション（ビルドの破損やテストの失敗）は容認されません。原因となった変更は、問題が対処されない場合、バックアウトされます。

4. コミットの構造

- 同じ修正の一部であるコミットをまとめる：
 - 50個の`Makefile`に影響する`make`変数の修正は、すべてをまとめて1つのコミットにするべきです。
- 各コミットを個別のパッチ/修正/追加などに分ける：
 - 1つのコミットで3つのバグを修正し、「いくつかのバグを修正」としてまとめるのは避けてください。1つを修正し、テストし、コミットし、次に進んでください。
- 機能やバグ修正のパッチと、空白/レイアウトの更新を混ぜない：
 - これらは別々に行ってください。機能修正と空白変更を混ぜると、トランクからブランチへのプルアップで問題が発生することがあります。

5. コミットログの記述

- 変更の理由をコミットログに明確に文書化する：
 - 何が変わったか、**なぜ**変更されたかある程度詳細に記述してください。
 - 「なぜ」に焦点を当てるべきです。6か月後にログを読み、差分を見る人にとって有益であることが目標です。
 - 悪い例：「いくつかのものを修正した」または「cvs-1.10.0」
 - 良い例：「`for`ディレクティブの実行中のエラーで役立つ行番号を出力するようにした。」

- PR (問題報告) を修正した場合は文書化する:
 - コミットメッセージで「*PR category/bug-id*」というテンプレートを使用すると、バグデータベースの該当する問題報告にも追記されます (例: *Closes PR bin/6666*) 。
- レビューを受けた場合は文書化する:
 - 「*Reviewed by <mgr>*」のように記述してください。
- 適切なクレジットを与える:
 - PRで提出されたコードをコミットする場合、「*Code submitted in PR lib/393939 by Joe Doe*」のように適切なクレジットを付与してください。
 - コミットメッセージがメーリングリストで公開されるため、PR提出者のメールアドレスの明記は避けるべきです。
 - 他のオープンソースプロジェクトからコードを採用した場合は、「*From FreeBSD*」のようにクレジットを付与してください。

6. 他の開発者のコミットを元に戻さない

- 他の開発者のコミットに同意できない場合でも、****自分で元に戻さない****でください。
- その開発者に連絡し、問題点を説明し、****その開発者自身****に変更をバックアウトしてもらおうよう依頼してください。
- 合意に至らない場合は、調停機関として****Core Team**** (<core@NetBSD.org>) に連絡してください。

出典: [NetBSD Commit Guidelines](<https://www.netbsd.org/developers/commit-guidelines.html>)

8. NetBSDのセキュリティサポート

NetBSDのセキュリティ問題は、Security Team、Security Alert Team、および****Security Officer****の3つのグループによって対応されています。これらのグループは、報告されたセキュリティ問題を調査し、文書化し、コードを更新します。

8.1. コードの品質とセキュリティ機能

NetBSDの開発チームは、何百万行にも及ぶコードの品質とセキュリティを維持・向上させるため、最新のツールと手法を活用しています。

- コードスキャン: 商用のコードスキャナ「Coverity」と、開発者が開発したプライベートなコードスキャナ「Brainy」による定期的なソースツリーの分析が行われています。
- セキュリティ機能:
 - ****IPsec**** (ネットワーク層でのセキュリティプロトコル)
 - ****NPF**** (独自のファイアウォール)
 - ****Veriexec**** (ファイル整合性システム)
 - ****kauth(9)**** (カーネル認証フレームワーク)
 - ****GD**** (ディスク暗号化)

8.2. エクスプロイト緩和策

最新のエクスプロイト緩和策が多数サポートされており、ハードウェアの能力に応じて適用されます。

- ****WX**** (ユーザランドおよびカーネルでの「書き込みと実行の排他」)
- ****Userland ASLR**** (ユーザランドのアドレス空間配置のランダム化)
- ****Kernel ASLR**** (カーネルのアドレス空間配置のランダム化)
- SMEP、****SMAP**** (CPUのセキュリティ機能)
- その他、カーネル内部のバグ検出機能

また、SSH (OpenSSH) やKerberos 5 (Heimdal) といったクラシックなセキュアなネットワークサービスも利用可能で、すべてのサービスは最もセキュアな設定をデフォルトとし、新規インストールではどのサービスもデフォルトで有効になっていません。

8.3. セキュリティ問題の報告

NetBSDに関するセキュリティ問題を報告するには、****Security Alert Team**** (<security-alert@NetBSD.org>) に連絡してください。機密情報については、Security OfficerのPGPキーを使用して暗号化することが推奨されています。

8.4. セキュリティアドバイザリ

NetBSDで深刻なセキュリティ問題が発見・修正された場合、問題の内容と修正方法を記した****セキュリティアドバイザリ****が発

行されます。

- 通知: *netbsd-announce* および *security-announce* メーリングリストなどで公開されます。
- アーカイブ: NetBSDのリリースバージョンごとのアドバイザリがアーカイブされています (例: NetBSD 10.1、10.0、9.4など)。
- 遅延の可能性: 複数の脆弱性をまとめて対応する場合や、他のベンダーとの調整のため、アドバイザリの公開が遅れることがあります。

8.5. pkgsrc (パッケージコレクション) のセキュリティ

NetBSD Packages Collection (pkgsrc) に含まれるサードパーティ製ソフトウェアの脆弱性に対処するため、NetBSDは以下の機能を提供しています。

- 脆弱性リスト: pkgsrc Security Teamによって、既知の脆弱性リストが**NetBSD FTPサイト**で公開・署名されています。
- セキュリティ監査: *pkg_admin* ツールを使用して、インストールされているパッケージに既知の脆弱性がないかを確認できます。

```
# 脆弱性リストをダウンロード
pkg_admin fetch-pkg-vulnerabilities
```

```
# インストール済みパッケージを監査
pkg_admin audit
```

pkg_admin を使用して、この監査を日次のセキュリティスクリプトに含めることも推奨されています。

- pkgsrcのセキュリティ問題報告: ``pkg_admin audit``で検出されないpkgsrcのセキュリティ問題を発見した場合は、**pkgsrc Security Team**に連絡し、GPGキーを使用してレポートを暗号化することが推奨されています。

出典: [Security and NetBSD](<https://www.netbsd.org/support/security/>)

9. NetBSD Foundation メンバーシップ申請プロセス

NetBSD Foundationのメンバーシップ申請は、以下のステップで進行します。

1. スポンサーの決定**

- 既存のFoundationメンバーが、潜在的なメンバーをスポンサーすることに合意します。
 - これはメンバー間の議論の結果、または潜在的なメンバーからの依頼によって発生します。
- スポンサーは潜在的なメンバーにその旨を伝え、メンバーシップ申請フォーム (ディレクトリ内の *membership-app* ファイル) を送付します。

2. フォームの記入と提出**

- 潜在的なメンバーはフォームに記入し、電子メールでメンバーシップ委員会 (``membership-exec@NetBSD.org``) に返送します。
- フォームは主に申請者本人が記入することを意図していますが、一人以上の開発者と相談しながら記入しても構いません。
- この時点で、潜在的なメンバーのエントリーが *completed-forms* サブディレクトリに作成されます。以降の段階では、この情報が最新に保たれます。

3. メンバーシップ委員会による意見募集**

- メンバーシップ委員会は、Foundationの既存メンバーに対し、潜在的なメンバーについての情報と、申請に関するコメントを募集します。
 - 意見募集に使用するフォームは、ディレクトリ内の *rfc-template* ファイルにあります。フォームを自動で記入・送信するためのスクリプト (``send-rfc-form``) もあります。
- **コメント期間は暦日14日間**とします。
- メンバーシップ委員会は、申請者に対して申請書を受け取ったこと、および次に委員会から連絡がある時期を通知します。

4. メンバーシップ委員会の決定**

- メンバーシップ委員会は、申請書と寄せられたコメントを考慮し、申請者へのメンバーシップ提供の可否を決定します。

- メンバーシップが拒否された場合：その理由とともに申請者に伝えられ、このプロセスは終了します。

5. 招待と契約書の送付**

- メンバーシップ委員会は、申請者にメンバーシップへの招待状とメンバーシップ契約書を送付します。
- 申請者のエントリーが *status* ファイルに作成され、*completed-forms* ディレクトリの対応するエントリーから関連情報がすべて記入されます。

6. 契約書の返送とアカウント作成**

- 申請者は、条件を受け入れる場合、署名したメンバーシップ契約書をNetBSDプロジェクトの秘書（招待状に記載された住所）に返送します。
- メンバーシップ委員会は、`admins@NetBSD.org` に新しいメンバーのアカウント作成を依頼します。
- 管理者は新しいメンバーのアカウントを作成し、*status* ファイルおよび *completed-forms* ディレクトリの対応するエントリーを更新します。
- 管理者は、この完了を申請者、スポンサー、およびメンバーシップECに通知します。

7. 最終的な手続き**

- スポンサーはNetBSD開発者メーリングリストで申請者を歓迎します。
- ウェブサイト（`www.NetBSD.org`）は、プログラムに基づいて（例：月に一度など、該当者がいる場合）申請者を公表し、メンバーシップECに通知します。
- メンバーシップECは、新規メンバーシップのアナウンス（ウェブサイトからのテキスト）を `netbsd-announce` に投稿します。
- メンバーシップECは、新規メンバーの申請を確認し、すべてのファイルが更新されていることを確認して、このケースをクローズします。

出典 : [New NetBSD developer application procedure](<https://www.netbsd.org/foundation/policies/application-procedure.html>)

10. nono のバージョンアップ

pkgsrc/emulators/nono 以下を更新して、nonoのバージョンアップを試みます。

10.1. フリーズしているか確認

以下のURLを見て、pkgsrcがフリーズしているか確認します。フリーズしている時はバージョンアップしません。

<https://mail-index.netbsd.org/pkgsrc-changes/index.html> <https://www.pkgsrc.org/is-a-freeze-on/>

10.2. バージョン指定

Makefile 内のバージョンをあげます。

```
DISTNAME= nono-1.8.0
```

10.3. チェックサム作成

```
% make makesum
```

```
drwxr-xr-x  2 jun  wheel   512 Jun  2 20:35 CVS
-rw-r--r--  1 jun  wheel    54 May 14 2024 DESCR
-rw-r--r--  1 jun  wheel  1202 Jun  2 20:36 Makefile
-rw-r--r--  1 jun  wheel   357 Nov  2 2025 PLIST
-rw-r--r--  1 root  wheel   350 Jun  2 20:36 distinfo
```

make package して動作を確認します。

10.4. pkglint

pkglint をして正しいか確認します。

```
pkg_add pkglint
pkglint .
% pkglint .
Looks fine.
```

10.5. commit

バージョンアップ時のアップデート内容をしらべておきます。nonoの場合は、changes.htmlに日英併記された更新内容があるので、英語部分だけを変更点とします。更新ログの一行目は「パッケージ名: Update to バージョン」の形式にします。

<http://www.pastel-flower.jp/~isaki/nono/doc/changes.html>

```
cvs commit
```

```
nono: update to 1.8.0.
```

```
1.8.0 (2026/05/30)
```

```
vm(Update): "Change the VM time resolution from 1GHz to 19.2GHz. This allows the XP processor to run at exactly 6.
vm(New): "Implement tone and volume on SSG(YM2149)."
```

```
vm(New): "Implement PAM behavior on the volume register on SSG(YM2149). This makes NetBSD/luna68k's psgpam(4) audi
GUI(Fix): "Fix possible flickering on the sound monitor."
```

```
GUI(Fix): "Fix an issue where the access indicators of devices connected via SCSI bus would not light up correctly
app(Update): "Support libslirp-4.9."
```

10.6. doc をcommit します

doc をcommit します。コメントは、「doc:Updated カテゴリー/パッケージ名 to バージョン」に設定します。

```
doc: Updated emulators/nono to 1.8.0
```

```
cvs diff -u CHANGES-2026
```

```
Updated emulators/nono to 1.8.0 [jun 2026-06-02]
```

11. stable-diffusion.cppを使ってみる

11.1. 特徴

- キーワードから画像を生成します。
- NetBSD/amd64, NetBSD/aarch64で動作しています。

11.2. 準備するもの

- NetBSD/amd64, NetBSD/aarch64
- 24GB程度の合計スワップ領域(8GB+16GBとか)
- pkgsrc/devel/cmake

11.3. スワップ領域の追加

```
swapctl -lg
dd if=/dev/zero of=swap1 count=20000000
chmod 600 swap101
swapadd swap101
swapctl -lg
```

11.4. cmakeのインストール

```
cd /usr/pkgsrc/devel/cmake
make package-install
```

11.5. stable-diffusion.cppのインストール

```
git clone --recursive https://github.com/leejet/stable-diffusion.cpp
cd stable-diffusion.cpp
mkdir build
cd build
cmake ..
cmake --build . --config Release
```

11.6. stable-diffusion-v3-2のダウンロード

<https://huggingface.co/stabilityai/stable-diffusion-3-medium>
Files and versionsタブをひらく
https://huggingface.co/stabilityai/stable-diffusion-3-medium/blob/main/sd3_medium_incl_clips_t5xxlfp16.safetensors
をダウンロードする。

11.7. 画像の生成例

```
./bin/sd -m ../../models/stable-diffusion-v3-2/sd3_medium_incl_clips_t5xxlfp16.safetensors
--cfg-scale 5
--steps 30
--sampling-method euler
--seed 42
-o test.png
-b 2
-p "A girl of about 17 years old, with blonde hair and blue eyes,
wearing a white dress, standing in a meadow, with the wind blowing.
She is raising her hair with one hand.
In the distance, mountains can be seen,
and in the foreground, a river is flowing.
The sunset is visible above the mountains,
and the girl is smiling in Kancolle style."
```

11.8. 引数の説明

<code>./bin/sd</code>	stable-diffusion.cpp の実行ファイルへのパスです。
<code>-m <モデルパス></code>	使用するモデルファイル (Stable Diffusion 3 Medium) を指定しています。
<code>--cfg-scale</code>	生成される画像がプロンプトにどれだけ忠実になるかを調整するパラメータです。6 という値は、一般的に自然でバランスの取れた画像を得るための標準的な範囲内です。
<code>--steps</code>	画像生成のステップ数を指定しています。30 は、速さと品質のバランスが取れた一般的な設定です。
<code>--sampling-method</code>	使用するサンプラー (ノイズ除去アルゴリズム) を指定しています。euler は最も基本的なサンプラーの一つで、特に問題ありません。他の高速なサンプラー (例: eulera、`dpmp_2m`など) を試すことも可能です。
<code>--seed</code>	乱数シードを指定しています。特定の画像を再現したり、似た画像を連続して生成したりするのに有用です。
<code>-o <出力パス></code>	出力ファイル名を指定しています。`date +%Y%m%d-%H%M` の部分は、実行時に「年-月-日-時-分」の形式のタイムスタンプに置き換えられることを意図しており、連番のファイル名を自動生成する一般的なシェル技法として有効です。

12. GCCフラグの有効性と重複に関する分析

GCCフラグのセット

```
CFLAGS+=-Wno-error=incompatible-pointer-types -Wno-error=declaration-after-statement -Wno-error=deprecated-declarations -Wno-error=implicit-function-declaration -Wno-error=sign-compare -Wno-error=undef -Wno-array-bounds -Wno-return-type -Wno-error=char-subscripts -Wno-cpp -Wno-error=cast-align -Wno-dangling-pointer
```

について、その適切性を評価します。

結論として、「プロジェクトの目的によっては適切だが、新規コードや安全性を重視するプロジェクトでは非推奨」です。

このフラグ群の主な目的は、コンパイルオプションに `-Werror` (すべての警告をエラーとして扱う) が設定されている環境で、特定の警告だけをエラーとして扱わないようにする (つまり、コンパイルを通す) ことにあります。

12.1. フラグの全体的な評価

項目	評価	理由
技術的な目的	適切	既存のコードが生成する多くの警告をエラーとして扱わないようにし、ビルドを強制的に完了させるという目的には適しています。
コードの安全性	非推奨	ポインター、アライメント、配列境界などの潜在的なバグを見逃す可能性が高くなります。これらの警告は本来、無視すべきではないものです。
互換性	適切	古いC標準 (C99以前) や、古いライブラリ (DeprecatedなAPI) を含むコードをコンパイルする際には、しばしば必要になります。

12.2. 個別のフラグに関する詳細な注意点

以下のフラグは、潜在的に深刻なバグや移植性の問題を無視する可能性があるため、特に注意が必要です。

フラグ名	該当する警告の内容	安全性への影響	推奨される対応
<code>-Wno-error=incompatible-pointer-types</code>	互換性のないポインター型間の代入/キャスト。	高。セグメンテーション違反などの実行時エラーにつながる可能性があります。	コード側でキャストを修正するか、警告の原因を特定して修正すべきです。
<code>-Wno-array-bounds</code>	配列の境界を越えたアクセス。	高。メモリ破壊につながる可能性があります。	配列アクセスを見直し、コードを修正すべきです。
<code>-Wno-error=cast-align</code>	不適切なアライメントのポインターをキャスト。	中。特定のアーキテクチャでクラッシュやパフォーマンス低下の原因になります。	キャストを避けるか、アライメントを確保するためにコードを修正すべきです。
<code>-Wno-dangling-pointer</code>	ライフタイムが切れたポインターの使用。	高。最新のGCCで追加された安全機能です。これを無効にすると重大なメモリ安全性の問題を見逃します。	コードを修正すべきです。
<code>-Wno-return-type</code>	非 <code>'void'</code> 関数で <code>'return'</code> 文がない。	高。関数が値を返すべきところで返さないのは、未定義動作を引き起こす重大なバグです。	コードを修正すべきです。

12.3. 互換性確保のために許容されるフラグ

以下のフラグは、古い言語コードベースを現代のGCCでコンパイルするために、やむを得ず使用されることがあります。

フラグ名	理由
<code>-Wno-error=declaration-after-statement</code>	C99以前のC標準では、変数の宣言はブロックの先頭に限定されます。古いコードをC99以降のGCCでコンパイルする際に必要です。
<code>-Wno-error=implicit-function-declaration</code>	関数宣言なしに呼び出しを行う古いCの書き方を許可します。
<code>-Wno-error=deprecated-declarations</code>	使用しているAPIが非推奨 (Deprecated) であっても、コンパイルエラーにしません。
<code>-Wno-cpp</code>	C++スタイルのコメント (<code>//</code>) をCファイルで使用することを許可します。

これらのフラグは、一時的な措置として使用する分には「適切」と言えますが、長期的にメンテナンスされる新しいコードでは、警告を抑制するのではなく、警告の原因となるコード自体を修正していくことが最も推奨されます。

12.4. `-Wno-error=` と `-Wno-` の役割

フラグ名	理由
<code>-Wno-error=<警告名></code>	**特定の警告**が、コンパイルオプション (<code>-Werror</code> など) によって**エラーとして扱われるのを回避**し、単なる警告に戻す (または警告を無効化する) 効果があります。
<code>-Wno-<警告名></code>	**特定の警告**を完全に**無効化**します。

両方ある場合、完全に警告を無効化するという目的では冗長になります。一般的には、エラー化を回避しつつ完全に警告を消したい場合は、シンプルな `-Wno-<警告名>` のみを使用すれば目的は達成されます。

13. PC6001VXを使ってみる

13.1. 特徴

- PC6001エミュレータ PC6001VXを使ってみます。

13.2. 準備するもの

- `pkgsrc/emulators/PC6001VX`

13.3. Tiny みずいろ

- PC6001VX 起動
- How Many Pages? 2
- 右クリック→拡張カートリッジ→PC-6006 拡張ROM/RAMカートリッジ→ROMなし
- <https://bn-x.tripod.com/02tinymizu/>
- <https://tsutsui.hatenablog.com/entry/ar1891638>

機種設定および起動時の MODE (機種選択) は PC-6001(32K)を選択

How Many Page? は 2 を入力

エミュレータのテープファイルとして `tiny_mizuiro_prologue_PSG_01.P6` を選択 (※1)

"CLOAD" で BASIC プログラムロード

ロードが終わったら次のテープファイルとして `tiny_mizuiro_prologue_PSG_02.P6` を選択

"RUN" で BASIC プログラムを実行 →機械語パートのロード開始

ロードが終わって "HIT ANY KEY" の表示が出たら適当なキーを押す

デモスタート! (※2)

13.4. 画像出力

- 画像変換 for PC-6001でVRAMをダウンロードして
- PC6001VXモニタモードで
- `loadm` ファイル名 `0xe000 0xf9ff`
- `out 0xb0 0xf2`
- 事前に `screen 3,2,2` とか `screen 4,2,2` とか実行しておいて BASICで何かしら初期化したあとにデバuggで
- `loadmem` [ファイル名] `0xe000 0xf9ff`
- でロードして、デバuggから戻って PAGEキー (PAGE UPキー) を押すとグラフィック画面出ると思います

13.5. 投稿がしたいのに feat. PC-6001 プログラム

- <https://tsutsui.hatenablog.com/entry/toukoumiku-p6>

13.6. 1000光年

- <https://tsutsui.hatenablog.com/entry/1000kounen-p6>

[F8]キーでにじみ設定トグル。緑色を描くように設定します。

```
.pc6001vx4/pc6001v.ini
```

```
[DISPLAY] Mode4Color = 4 ;MODE4カラーモード 0:モノクロ 1:赤/青 2:青/赤 3:ピンク/緑 4:緑/ピンクを 4にする。
```

13.7. PC-6001 PSG音源ドライバ用 MMLコンパイラ (C言語コマンドライン版)

- <https://tsutsui.hatenablog.com/entry/p6psgmmlc>

13.8. NetBSD/earmv7hf

- 問題点 <https://mail-index.netbsd.org/pkgsrc-bugs/2025/09/07/msg074314.html> の差分での変更点
- つついさん作変更点をファイルに落とした: https://github.com/ebijun/NetBSD/blob/master/pkgsrc/PC6001VX/qt6-qtbase/patch-src_corelib_plugin_qelfparser_p.cpp を `pkgsrc/x11/qt6-qtbase/patches`に置きます。
- `make makepatchsum`
- `make package` でqt6/パッケージを作る
- PC6001VXを一度起動する (キー入力がなかなか入らない)
- `~/pc6001vx4/pc6001vx.ini` の`hwAccel=true`を`false`に変更する
- もういちどPC6001VXを起動する

13.9. rpi-psgplayer

- <https://github.com/tsutsui/rpi-psgplayer>
- カーネルを作り直す
- `git clone git@github.com:tsutsui/rpi-psgplayer.git`

- `git clone git@github.com:tsutsui/p6-odorumiku.git`
- `git clone git@github.com:tsutsui/p6-psgmlc.git`
- <https://github.com/tsutsui/p6psgmlc>

```
p6-odorumiku/mml /usr/local/Github/p6-odorumiku/mml amir63d$ p6psgmlc sm40196287_p6drv.mml sm40196287_p6drv.bin
amir63d$ ls sm40196287_p6drv.bin sm40196287_p6drv.mml
```

14. mlterm-wscons を使ってみる

14.1. mlterm

リソースの限られたマシンでは、Xを起動させるだけでリソースを使いきってしまう場合があります。mlterm (pkgsrc/x11/mlterm) [mlterm] は、X上で動作する多言語ターミナルエミュレータですが、mlterm-fbを有効にしてコンパイルすると、フレームバッファ上で動作する多言語ターミナルエミュレータとして利用できます。



14.2. mlterm-wscons 向けオプション設定

/etc/mk.confに、mltermに与えるオプションを指定しておきます。pkgsrc/x11/mlterm以下を利用して、mltermをインストールします。

```
% grep mlterm /etc/mk.conf
PKG_OPTIONS.mlterm=xft2 mlterm-fb
# cd /usr/pkgsrc/x11/mlterm
# make package-install
# which mlterm-wscons
/usr/pkg/bin/mlterm-wscons
```

14.3. mlterm-wscons フォント設定

mlfc コマンドを実行すると、true typeフォントを検索して、~/mlterm/aafont と ~/mlterm/font-fb ファイルを更新します。

```
$ mlfc
Updating /home/jun/.mlterm/aafont
Updating /home/jun/.mlterm/font-fb
```

14.4. 起動

コンソール画面からmlterm-wsconsを起動します。

```
% mlterm-wscons
Shift-F1: 画面横分割
```

```
Shift-F2: 画面縦分割
Shift-F3: 画面移動
% mlterm-wscons -p RPI2.png --contrast 70
背景をRPI2.pngにします。コントラストを70%に設定します。
```

14.5. PCでのmlterm-fb起動

NetBSD/i386,amd64でmlterm-fbを利用する場合、起動時にVESAを指定します。 [はよーん]

1. 起動オプションで5を選択
2. > プロンプトでvesa を指定
3. boot
4. 起動
5. ログイン
6. mlterm-wscons起動

14.6. テスト画像表示

mlterm-wsconsを起動した状態で、catを利用して、画像イメージを表示します。

```
% ftp http://mlterm.sourceforge.net/vimperator.six
% cat vimperator.six
```

14.7. 画像変換方法

netpbmをインストールして、以下のコマンドを実行するとsixel形式に変換できます。

```
# cd /usr/pkgsrc/graphics/netpbm;make package-install
% jpegtopnm aaa.jpg |pnmquant 256 |ppmtosixel > aaa.sixel
% pngtopnm aaa.png |pnmquant 256 |ppmtosixel > aaa.sixel
```

14.8. mltermソースからmlterm-wsconsをコンパイルする

```
# ./configure --with-gui=xlib,fb
# make install
```

14.9. 回転させて表示

NetBSD/zaurus C1000以降および、NetBSD/hpcarm WS011SHでは、そのまま起動すると、mltermが縦画面で起動してしまいます。この場合、mltermに--rotateオプションをつけて起動するか、起動後に\$ mlcc rotate_display trueすると回転します。。。--rotateオプションの挙動を変更し、--rotate=rightで時計回りに、--rotate=leftで反時計回りに回転します。(WS011SHは反時計回り?) この機能は最新版(<http://t.co/l52DRJGwGf>) に追加されています。

14.10. mltermについて

<http://mlterm.sourceforge.net/> に、より詳しい情報があります。

14.11. mlterm-fb上でsayakaを動かしてみる

mlterm-wscons上で日本語が表示できたら、sayaka [sayaka] を利用してmisskey.ioのタイムラインを表示してみましょう。

14.12. sayakaのインストール

pkgsrcからsayakaをインストールします。

```
# cd /usr/pkgsrc/net/sayaka
# make package-install
```

14.13. nanotodonのインストール

pkgsrcからnanotodon をインストールします。

```
# cd /usr/pkgsrc/net/nanotodon
# make package-install
```

```
[mlterm] mlterm http://mlterm.sourceforge.net/
```

```
[tw] tw http://shokai.github.io/tw/
```

```
[はよーん] http://jnug.net/msg012914ja.html
```

```
[sayaka] https://github.com/isaki68k/sayaka
```

14.14. NetBSD/x68k での表示方法

```
% mlterm-x68kgrf
LANG=ja_JP.UTF-8ja
```

15. Playfield Engine を使ってみる

15.1. Playfield Engine

Playfield Engine(<https://github.com/awemorris/PlayfieldEngine>)は、2Dゲームエンジンです。Playfield Scriptで書かれたスクリプトを使ってグラフィックをレンダリングし、サウンドを再生します。C/JavaScriptのようなスクリプト言語であるPlayfield Scriptと、ポータブルなゲームライブラリであるStratoHALによって駆動され、Windows、macOS、Linux、Chromebook、WebAssembly、iOS、Android、およびコンソールなど、幅広いプラットフォームで動作します。ランタイムはWindowsで約1.4MBと非常に小さく、古いマシンや低スペックのマシンでもスムーズに動作するように設計されており、誰もがゲーム開発にアクセスできるようにすることを目指しています。

以下の手順でNetBSDにインストールしてみます。

15.2. 必要なパッケージのインストール

pkginをインストールします。パッケージの置いてある場所をrepositories.confに追加します。

```
pkg_add pkgin
vi /usr/pkg/etc/pkgin/repositories.conf
http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/`uname -p`/2025
```

```
pkgin install git-base
pkgin install cmake
pkgin install ninja-build
```

15.3. プログラムのインストール

リポジトリをクローンして、cmake経由でプログラムを作成します。

```
git clone --recursive https://github.com/awemorris/PlayfieldEngine.git
cd PlayfieldEngine
git submodule update --init
mkdir build
cd build
cmake -DCMAKE_INSTALL_RPATH=/usr/X11R7/lib ..
make
check /usr/local
make install
check /usr/local
```

15.4. サンプルプログラムの実行

```
cd ../samples/shoot2/
./playfield
```

15.5. 実行ファイルをまとめる

/usr/local/以下に必要なファイルを集めて、配布用にまとめてみます。

```
cp -r samples /usr/local
cd /usr/local
tar czvf /root/playfield-`uname -p`-nightly-20251026-7cbe585.tgz /usr/local
```

16. RaspberryPIのNetBSDイメージ2026進捗どうですか

16.1. 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
2025/5/31	10.99.14		3.9.4nb2		libjpen-turbo	OSC 名古屋	https://mail-index.netbsd.org/port-arm/2025/05/07/msg009168.html
2025/6/21	10.99.14	5.1.1nb2		3.0.15		OSC 島根	https://mail-index.netbsd.org/port-arm/2025/06/14/msg009216.html
2025/8/3	10.99.15	5.1.1nb3		3.5.1	libcrypto/libssl.so.16	OSC 京都	https://mail-index.netbsd.org/port-arm/2025/07/30/msg009355.html
2025/8/4	10.99.15→11.99.1				11.0_BETA	OSC 京都後	https://mail-index.netbsd.org/port-arm/2025/08/04/msg009368.html
2025/9/24	11.99.3	5.11nb4				OSC 新潟	https://mail-index.netbsd.org/port-arm/2025/09/24/msg009415.html
2025/10/10	11.99.3				gcc14	OSC 東京秋	https://mail-index.netbsd.org/port-arm/2025/10/10/msg009448.html
2025/12/01	11.99.4	5.1.1nb5	3.9.4nb4		libc-12.224	OSC 広島	https://mail-index.netbsd.org/port-arm/2025/12/01/msg009479.html
2026/1/31	11.99.5		3.9.4nb4	3.5.1	icu 77.1	OSC 大阪	https://mail-index.netbsd.org/port-arm/2026/01/13/msg009516.html
2026/2/18	11.99.5			3.5.5	icu 78.2	OSC 東京春	https://mail-index.netbsd.org/port-arm/2026/02/18/msg009548.html

2026/3/26	11.99.5	5.1.1nb7	3.9.4nb6	3.5.5	ruby33	OSC 香川	https://mail-index.netbsd.org/port-arm/2026/03/26/msg009569.html
2026/4/30	11.99.5	5.1.2		3.5.6		OSC 名古屋	https://mail-index.netbsd.org/port-arm/2026/04/30/msg009599.html
2026/6/24	11.99.6		3.9.4nb8	3.5.7		OSC 北海道	https://mail-index.netbsd.org/port-arm/2026/06/23/msg009623.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いと答えてくれて、あー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、ソフトウェアの脆弱性が出たとか、理由はいくつかあると思いますが、試しにずっと更新して配りつづけることにしてみました。

イメージのサイズは2GBにしてみました。ダウンロードにかかる時間とか考えると、これ以上でっかくすると使ってもらえませんが、2GBのカードのサイズはこんなくらいにすればいいよとFreeBSDのワナーさんに教えてもらってずっとそのサイズにしていますが、手狭になったので増やしました。

イメージに入れるソフトを何にするか考えたんですが、mikutterとmltermにしてみました。RubyのGUI環境+ネットワーク認証を使うソフトと、基本的なターミナルソフトで、sixelグラフィックも表示できるのでおもしろそうです。

作り方は

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

みたいに作って、あらかじめ作っておいたパッケージを組み込んで動作テストをします。mikutterで「あひる焼き」とつぶやいて返事が帰ってくればネットワーク認証と画面表示とRubyまわりと漢字入力がうまくいっています。

16.2. 新しいハードウェア対応

1. RPI5:2023/10末出荷開始 Broadcom BCM2712 64-bit quad-core Arm Cortex-A76 2.4GHz
2. <https://github.com/worproject/rpi5-uefi> <http://mail-index.netbsd.org/port-arm/2024/01/19/msg008524.html>
3. RPI4:OSC2019島根から：<http://mail-index.netbsd.org/port-arm/2019/10/03/msg006208.html>

<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIImage/Image/aarch64/README>

1. RPI3/RPI0WのBluetooth/無線LAN:OSC2019広島版からテストをはじめました
2. Raspberry PI zero 2 W earmv7hf版で動きます。

<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIImage/Image/armv7hf/README>

16.3. ソフトウェア配布方法

NetBSDのftpサイトはCDN対応のところからダウンロードできるようになりました。漫喫でも楽勝です。 - <http://cdn.netbsd.org/> - <http://nycdn.netbsd.org/>

16.4. OSCでやっているデモ

RaspberryPIっぽいなにかということで、omxplayerを使って動画を流すデモと、nonoで NetBSD/x68k,OpenBSD/luna88k,NetBSD/luna68kを動かすデモをやっています。

16.5. security.pax.mprotect.enabled

```
man security
man paxctl
sysctl -a |grep pax
If application failed, such as omxplayer.
```

```
try to test
sysctl -w security.pax.mprotect.enabled=0
```

16.6. GPIOのドキュメント

GPIOの使い方をまとめてくれた方が。

- NetBSD GPIO DOC by Marina Brown <https://github.com/catskillmarina/netbsd-gpio-doc/blob/master/README.md>
- つついさんがFM音源を

16.7. 64bit対応

ryo@netbsd さんによる rpi64wip実装が進み、NetBSD/aarch64としてRPI3/4で利用できます。

- <https://github.com/ryo/netbsd-src>
- <http://mail-index.netbsd.org/port-arm/2018/02/20/msg004631.html>
- <http://mail-index.netbsd.org/port-arm/2018/12/03/msg005297.html>

16.8. RPI4

- pinebookとpkgsrcを共用しています。
- <http://mail-index.netbsd.org/port-arm/2020/11/18/msg007066.html>
- <https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/README>

16.9. RPI5

- <https://github.com/NumberOneGit/rpi5-uefi/releases/tag/v0.1> を展開したmicroSDを差します。
- USB接続のSSD等にRPI4用のイメージをコピーしてUSBに差して電源入れると起動します。
- m2.SSDから起動したいときはRaspberryPi OSでファームをアップデートする必要があります。

16.10. armv7のいろいろ

NetBSD ARM Bootable Imagesがあります。

- <https://nycdn.netbsd.org/pub/arm/>

16.11. ご注文はなんとかですか（弱点）

- RPI4のGPUとか

16.12. まとめ

OSCごとにイメージをつくっていると、だいたいBINDとOpenSSLの脆弱性に対応できていい感じです。なんでOSCの直前になると脆弱性がみつかるのでしょうか。たまにBSD自体の10年もののバグとかも発掘されて楽しいです。リリース間隔があけばあくほど、ひとりで対応できる作業量を越えてしまう気がするので、いまんとここれでいいのかほんとうに。

17. オープンソースカンファレンスNetBSDブースこの一年

日本NetBSDユーザーグループは2024年もオープンソースカンファレンスを中心とするイベントに参加しています。2020/1のOSC2020大阪以降はZOOMでの参加になりましたが、2023/5のOSC2023名古屋以降ZOOMと展示を分けて開催するようになりました。NetBSD/aarch64からZOOM会議に参加する試みをしました。従来、ブースへの展示機材持ち込み&差し入れに代わり、事前にtwitter上でデモ動画等を投稿されたものをtogetterでまとめておいて紹介するようにしてみました。、引き続きセミナー時間での参加・発表を歓迎します。

17.1. NetBSD環境からのZOOM会議参加

NetBSDからZOOM等の会議に参加するためには、以下の手順をとります。

1. rustが動くようにする。
2. audioが動くようにする。
3. 内蔵カメラが動くようにする。
4. Firefox80以降をpkgsrcからインストールする

5. FirefoxのプラグインでLinuxまたはFreeBSDからインストールしているように見せかける。

17.2. これまでに参加した一覧

これまでに参加した一覧は以下のとおりです。

<https://github.com/ebijun/NetBSD/blob/master/Guide/OSC/OSC100.csv>

このファイルはオープンソースカンファレンス過去来場者数一覧 <http://www.ospn.jp/visitors/> を元に作成しています。

17.3. どのくらい参加しているか

- 2025/12/27までにOSCは256回開催されています。
- JNUGは230回参加しています。89.5%→89.8% (前年比+0.3%)

17.4. OSCこの一年

17.4.1. 2025年まとめ

- 日本全国各地で13回+オンライン開催1回+ODC開催
- 参加者: 113,702人 年間参加者3356人 3115人(2021年)→2383人(2022年)→3304人(2023年)→3296人(2024年)→3356人(2025年)
- 参加団体:7192グループ 年間参加グループ 320グループ
- 133グループ(2021)→127グループ(2022)→299グループ(2022)→317グループ(2023)→357グループ(2024)→320グループ(2025年)

17.5. togetherアクセスで見たNetBSDブース

togetherのアクセスログは以下の場所にあります。

<https://github.com/ebijun/NetBSD/blob/master/Event/together/togetherview/view.csv>

この一年のアクセス数集計は以下の通りです。

OSC2026 Hokkaido	https://posfie.com//@ebijun/p/3QUhHvw	18
OSC2026 Sendai	https://posfie.com//@ebijun/p/NqQAv0o	327
OSC2026 Nagoya	https://posfie.com//@ebijun/p/lp0l8K0	418
OSC2026 Kagawa	https://posfie.com//@ebijun/p/TARTFTx	742
OSC2026 Tokyo/Spring	https://posfie.com//@ebijun/p/QvR9Tx9	773
OSC2026 Osaka	https://posfie.com//@ebijun/p/Vfaj1UC	506
OSC2025 Hiroshima	https://posfie.com//@ebijun/p/WkwSt6x	412
OSC2025 Fukuoka-Yamaguchi-Okinawa	https://posfie.com//@ebijun/p/NgePtck	437
KOF2025	https://posfie.com//@ebijun/p/poMBvId	422
OSC2025 Tokyo/Fall	https://posfie.com//@ebijun/p/KsHuz6A	428
OSC2025 Niigata	https://posfie.com//@ebijun/p/9uZIwuH	432
ODC2025	https://posfie.com//@ebijun/p/tctRUi9	605
OSC2025 Kyoto	https://posfie.com//@ebijun/p/LpFGcI4	739
OSC2025 Hokkaido	https://posfie.com//@ebijun/p/jobKi6a	591

17.6. netbsd-advocacyメーリングリストへの報告

netbsd-advocacyメーリングリストへの参加報告をしました。

<https://wiki.netbsd.org/users/jun/>

NetBSD machines at Open Source Conference 2020 Osaka <http://mail-index.netbsd.org/netbsd-advocacy/2020/01/28/msg000823.html>

17.7. NetBSD観光ガイド作成

イベント毎に観光ガイドを作っています。セミナー参加者に配布しました。

一覧：

<https://github.com/ebijun/osc-demo/blob/master/README.md>

作成方法：

<https://github.com/ebijun/NetBSD/blob/master/Guide/Paper/sphinx.rst>

2020年以降のものは <https://cdn.netbsd.org/pub/NetBSD/misc/jun/OSC/> にあります。

228	OSC2026 北海道	http://www.re.soum.co.jp/~jun/OSC2026hokkaido.pdf
227	OSC2026仙台	http://www.re.soum.co.jp/~jun/OSC2026sendai.pdf
226	OSC2026名古屋	http://www.re.soum.co.jp/~jun/OSC2026nagoya.pdf
225	OSC2026香川	http://www.re.soum.co.jp/~jun/OSC2026kagawa.pdf
224	OSC2026東京春	http://www.re.soum.co.jp/~jun/OSC2026tokyospring.pdf
223	OSC2026大阪	http://www.re.soum.co.jp/~jun/OSC2026osaka.pdf
222	OSC2025広島	http://www.re.soum.co.jp/~jun/OSC2025hiroshima.pdf
221	OSC2025沖縄	http://www.re.soum.co.jp/~jun/OSC2025okinawa.pdf
220	OSC2025山口	http://www.re.soum.co.jp/~jun/OSC2025yamaguchi.pdf
219	OSC2025福岡	http://www.re.soum.co.jp/~jun/OSC2025fukuoka.pdf
218	KOF2025	http://www.re.soum.co.jp/~jun/KOF2025.pdf
217	OSC2025東京秋	http://www.re.soum.co.jp/~jun/OSC2025tokyofall.pdf
216	OSC2025新潟	http://www.re.soum.co.jp/~jun/OSC2025niigata.pdf
215	ODC2025	http://www.re.soum.co.jp/~jun/ODC2025.pdf
214	OSC2025京都	http://www.re.soum.co.jp/~jun/OSC2025kyoto.pdf
213	OSC2025北海道	http://www.re.soum.co.jp/~jun/OSC2025hokkaido.pdf
212	OSC2025島根	http://www.re.soum.co.jp/~jun/OSC2025shimane.pdf
211	OSC2025名古屋	http://www.re.soum.co.jp/~jun/OSC2025nagoya.pdf
210	OSC2025東京春	http://www.re.soum.co.jp/~jun/OSC2025tokyospring.pdf
209	OSC2025大阪	http://www.re.soum.co.jp/~jun/OSC2025osaka.pdf

17.8. 旅費

横浜からの旅費(=交通費+宿泊費),機材配送費,資料印刷費実費をまとめています。機材配送はヤマトの「スマホでかんたん発送」サービスを利用しています。

月	イベント	旅費	機材配送
2020/1	OSC大阪	26080	1469
2023/5	OSC名古屋	19300	0
2023/6	OSC北海道	45392	3891
2023/7	OSC京都	25004	3618
2023/10	OSC島根	56603	0
2023/11	KOF/OSC広島	59459	0
2023/11	OSC新潟	20958	0

2023/12	OSC福岡	37960	3720
2024/1	OSC大阪	28464	0
2024/5	OSC名古屋	20440	0
2024/6	OSC北海道	35993	4091
2024/7	OSC京都	25734	3376
2024/9	ODC	1120	0
2024/9	OSC広島	36604	0
2024/10	OSC長岡	19796	
2024/10	OSC東京	1080	
2024/11	OSC島根	58184	
2024/11	KOF	29704	
2024/11	OSC愛媛	40454	1622
2024/12	OSC福岡	35789	1886
2025/1	OSC大阪	28464	0
2025/2	OSC東京春	500	0
2025/5	OSC白浜	2600	0
2025/5	OSC名古屋	21494	0
2025/6	OSC島根	50014	0
2025/7	OSC北海道	32781	0
2025/8	OSC京都	25110	0
2025/10	OSC新潟	21120	0
2025/11	KOF	28780	0
2025/11	OSC福岡	46000	0
2025/11	OSC山口	26700	0
2025/11	OSC沖縄	35655	0
2025/12	OSC広島	38460	0
2026/01	OSC大阪	28360	0
2026/4	OSC香川	25039	0
2026/5	OSC名古屋	20880	0
2026/6	OSC仙台	22860	0
2026/6	OSC北海道	26719	0
2026/7	OSC島根	30578	0

17.9. 2026年

OSCは2026/1/31のOSC 大阪(<https://event.ospn.jp/osc2026-osaka/>)からはじまります。ブース展示/ミーティング時間での発表/差し入れ等を歓迎します。

18. RaspberryPIでNetBSDを使ってみる

18.1. 特徴

- NetBSDをRaspberryPIで利用するために、ディスクイメージを用意しました。
- Xが動いて、ご家庭のテレビでmikutterが動きます。
- うまく動いたら、動いた記念写真をツイートだ！
- fossil(<http://www.fossil-scm.org/>)も入れてあります。家庭内Webサーバとかチケットシステムとかwikiサーバになるんじゃないかい。

18.2. 準備するもの

- RaspberryPI本体
- HDMI入力のあるテレビ/ディスプレイ
- USBキーボード
- USBマウス
- 有線ネットワーク

ケースは Geekworm Raspberry Pi 3/4ケースをそれぞれ使っています。

18.3. 起動ディスクの作成

- ディスクイメージのダウンロード

```
earmv6hf
# ftp http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/
2016-11-12-earmv6hf/2016-11-12-netbsd-raspi-earmv6hf.img.gz
```

- 2GB以上のSDカードを準備します。
- ダウンロードしたディスクイメージを、SDカード上で展開します。

```
disklabel sd0 ..... 必ずインストールするSDカードか確認してください。
gunzip < 2016-11-12-netbsd-raspi-earmv6hf.img.gz | dd of=/dev/rsd0d bs=1m
```

18.4. NetBSD Arm Bootable Images

NetBSD Arm Bootable Images が、<https://nycdn.netbsd.org/pub/arm/> 以下にあります。RPIと同じ手順で起動できます。

18.5. RaspberryPIの起動

1. HDMIケーブル/USBキーボード/USBマウス/有線ネットワークをRPIにさします。
2. 電源を入れてRPIを起動します。
3. 少し待つと、HDMIからNetBSDの起動メッセージが表示されます。
4. メモリカードの容量にあわせたサイズまでルートパーティションを自動調整します。(現在、RPI2では自動調整プログラムの起動が失敗します)
5. 容量調整後に再起動します。再起動した後は、起動プロセスが最後まで進み、ログインできる状態になります。
6. 起動しない場合、まず基板上のLEDを確認してください。

赤いランプのみ点灯している場合

- OSを正しく読み込んでいません。
- 少なくともMSDOS領域に各種ファームウェアファイルが見えていることを確認する。
- SDカードの接触不良の可能性があるので、SDカードを挿しなおしてみる。
- ファームウェアが古いため起動しない

緑のランプも点灯している場合

- OSは起動しているのに画面をHDMIに表示できていません。
- HDMIケーブルを差した状態で電源ケーブルを抜き差しして、HDMIディスプレイに何か表示するか確認する。
- HDMIケーブル自体の接触不良。ケーブルを何度か差し直してください。
- 電源アダプタ容量には、少なくとも800mA程度の容量を持つアダプタを使ってください。スマートフォン用のアダプタならまず大丈夫です。起動途中で画面が一瞬消えたり、負荷をかけるといきなり再起動したりする場合は、電源やUSBケーブルを気にしてみてください。

18.6. ログイン

rootでログインできます。rootアカウントではリモートからログインすることはできません。

```
login: root
```

```
startxでicewmが立ち上がります。
```

```
# startx
```

18.7. mikutterを使ってみよう

- xtermからdilloとmikutterを起動します。

```
# dillo &
# mikutter &
```

- しばらく待ちます。
- mikutterの認証画面がうまく出たら、httpsからはじまるURLをクリックするとdilloが起動します。
- twitterのIDとパスワードを入力すると、pin番号が表示されます。pin番号をmikutterの認証画面に入力します。
- しばらくすると、mikutterの画面が表示されます。表示されるはずですが、落ちてしまう場合は時計が合っているか確認してください。
- 漢字は[半角/全角]キーを入力すると漢字モードに切り替わります。anthyです。
- 青い鳩を消したいとき：mikutterのプラグインを試してみる

```
% touch ~/.mikutter/plugin/display_requirements.rb
```

すると、鳩が消えます。mikutterはプラグインを組み込むことで、機能を追加できる自由度の高いtwitterクライアントです。プラグインに関しては、「mikutterの薄い本 プラグイン」で検索してみてください。

18.8. fossilを使ってみよう

fossilは、Wiki/チケット管理システム/HTTPサーバ機能を持つ、コンパクトなソースコード管理システムです。fossilバイナリひとつと、リポジトリファイルひとつにすべての情報が集約されています。ちょっとしたメモをまとめたりToDoリストを簡単に管理できます。

```
% fossil help
Usage: fossil help COMMAND
Common COMMANDs: (use "fossil help -a|--all" for a complete list)
add          changes  finfo      merge      revert     tag
addremove   clean    gdiff     mv         rm         timeline
all          clone    help      open      settings  ui
annotate    commit  import    pull       sqlite3    undo
bisect      diff    info      push       stash      update
branch      export  init      rebuild    status     version
cat         extras  ls        remote-url sync
% fossil init sample-repo
project-id: bcf0e5038ff422da876b55ef07bc8fa5eded5f55
server-id: 5b21bd9f4de6877668f0b9d90b3cff9baecea0f4
admin-user: jun (initial password is "f73efb")
% ls -l
total 116
-rw-r--r--  1 jun  users  58368 Nov 14 18:34 sample-repo
% fossil server sample-repo -P 12345 &
ブラウザでポート12345にアクセスし、fossil initを実行した時のユーザとパスワードでログインします。
```

18.9. キーマップの設定を変更する

- ログインした状態でのキーマップは/etc/wscons.confで設定します。

```
encoding jp.swapctrlcaps .... 日本語キーボード, CtrlとCAPSを入れ替える。
```

- Xでのキーマップは.xinitrcで設定します。

```
setxkbmap -model jp106 jp -option ctrl:swapcap
```

18.10. コンパイル済パッケージをインストールする

- コンパイルしたパッケージを以下のURLに用意しました。

```
% cat /etc/pkg_install.conf
```

```
PKG_PATH=http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv6hf/2016-11-12
```

- パッケージのインストール

pkg_addコマンドで、あらかじめコンパイル済みのパッケージをインストールします。関連するパッケージも自動的にインストールします。

```
# pkg_add zsh
```

- パッケージの一覧

pkg_infoコマンドで、インストールされているパッケージの一覧を表示します。

```
# pkg_info
```

- パッケージの削除

```
# pkg_delete パッケージ名
```

18.11. /usr/pkgsrcを使ってみよう

たとえばwordpressをコンパイル/インストールする時には、以下の手順で行います。

```
# cd /usr/  
# ls /usr/pkgsrc          ... 上書きしてしまわないか確認  
# ftp http://cdn.netbsd.org/pub/pkgsrc/current/pkgsrc.tar.gz  
# tar tzvf pkgsrc.tar.gz |head ... アーカイブの内容確認  
# tar xzvf pkgsrc.tar.gz  
# ls /usr/pkgsrc  
# cd /usr/pkgsrc/www/php-ja-wordpress  
# make package-install
```

```
# cd /usr/pkgsrc  
# cvs update -PA
```

18.12. パッケージ管理

pkg_chk コマンドを使って、インストールしたパッケージを管理してみましょう。あらかじめpkgsrcの内容を更新しておきます。どこからパッケージファイルを取得するかは、/etc/pkg_install.confのPKG_PATHに書いておきます。

```
# pkg_info    ... インストールしているパッケージ名と概要を出力します。  
# pkg_chk -g  ... 使っているパッケージの一覧を/usr/pkgsrc/pkgchk.confに作ってくれます。  
# pkg_chk -un ... パッケージをアップデートします。(nオプション付きなので実行はしません)  
# pkg_chk -u  ... パッケージをアップデートします。
```

18.13. ユーザー作成

```
# useradd -m jun  
# passwd jun
```

root権限で作業するユーザーの場合：

```
# useradd -m jun -G wheel
```

```
# passwd jun
```

18.14. サービス起動方法

/etc/rc.d以下にスクリプトがあります。dhcpクライアント(dhcpd)を起動してみます。

```
テスト起動：  
/etc/rc.d/dhcpd onestart  
テスト停止：  
/etc/rc.d/dhcpd onestop
```

正しく動作することが確認できたら/etc/rc.confに以下のとおり指定します。

```
dhcpd=YES
```

/etc/rc.confでYESに指定したサービスは、マシン起動時に同時に起動します。

```
起動：  
/etc/rc.d/dhcpd start  
停止：  
/etc/rc.d/dhcpd stop  
再起動：  
/etc/rc.d/dhcpd restart
```

18.15. vndconfigでイメージ編集

NetBSDの場合、vndconfigコマンドでイメージファイルの内容を参照できます。

```
# gunzip 2016-11-12-netbsd-raspi-earmv6hf.img.gz  
# vndconfig vnd0 2016-11-12-netbsd-raspi-earmv6hf.img
```

```
# vndconfig -l
vnd0: /usr (/dev/wd0e) inode 53375639
# disklabel vnd0
:
8 partitions:
#      size  offset  fstype [fsize bsize cpq/sgs]
a:  3428352  385024  4.2BSD      0    0    0 # (Cyl. 188 - 1861)
b:   262144  122880  swap                # (Cyl. 60 - 187)
c:  3690496  122880  unused        0    0    0 # (Cyl. 60 - 1861)
d:  3813376      0  unused        0    0    0 # (Cyl.  0 - 1861)
e:   114688   8192  MSDOS                # (Cyl.  4 - 59)
# mount_msdos /dev/vnd0e /mnt
# ls /mnt
LICENCE.broadcom  cmdline.txt          fixup_cd.dat        start.elf
bootcode.bin      fixup.dat            kernel.img          start_cd.elf
# cat /mnt/cmdline.txt
root=ld0a console=fb
#fb=1280x1024      # to select a mode, otherwise try EDID
#fb=disable        # to disable fb completely

# umount /mnt
# vndconfig -u vnd0
```

18.16. HDMIじゃなくシリアルコンソールで使うには

- MSDOS領域にある設定ファイルcmdline.txtの内容を変更してください。

<https://raw.githubusercontent.com/Evilpaul/RPi-config/master/config.txt>

```
fb=1280x1024      # to select a mode, otherwise try EDID
fb=disable        # to disable fb completely
```

18.17. 起動ディスクを変えるには

- MSDOS領域にある設定ファイルcmdline.txtの内容を変更してください。

root=sd0a console=fb ← ld0をsd0にするとUSB接続したディスクから起動します

18.18. 最小構成のディスクイメージ

NetBSD-currentのディスクイメージに関しては、以下の場所にあります。日付の部分は適宜読み替えてください。

```
# ftp://nyftp.netbsd.org/pub/NetBSD-daily/HEAD/201502042230Z/evbarm-earmv6hf/binary/gzimg/rpi_inst.bin.gz
# gunzip < rpi_inst.bin.gz |dd of=/dev/rsd3d bs=1m .... sd3にコピー。
```

RaspberryPIにsdカードを差して、起動すると、# プロンプトが表示されます。
sysinst NetBSDのインストールプログラムが起動します。

18.19. X11のインストール

rpi.bin.gzからインストールした場合、Xは含まれていません。追加したい場合は、

<ftp://nyftp.netbsd.org/pub/NetBSD-daily/HEAD/201310161210Z/evbarm-earmv6hf/binary/sets/> 以下にあるtarファイルを展開します。tarで展開するときにpオプションをつけて、必要な権限が保たれるようにしてください。

```
tar xzpvf xbase.tar.gz -C / .... pをつける
```

18.20. クロスビルドの方法

- ソースファイル展開
- ./build.sh -U -m evbarm -a earmv6hf release
- earm{v[4567],}{hf,}{eb} earmv4hf
- <http://mail-index.netbsd.org/tech-kern/2013/11/12/msg015933.html>

acorn26	armv2
acorn32	armv3 armv4 (strongarm)

cats shark netwinder	armv4 (strongarm)
iyonix	armv5
hpcarm	armv4 (strongarm) armv5.
zaurus	armv5
evbarm	armv5/6/7

18.21. 外付けUSB端子

NetBSDで利用できるUSBデバイスは利用できる（はずです）。電源の制約があるので、十分に電源を供給できる外付けUSBハブ経由で接続したほうが良いです。動作しているRPIにUSBデバイスを挿すと、電源の関係でRPIが再起動してしまう場合があります。その場合、電源を増強する基板を利用する方法もあります。

18.22. 外付けSSD

コンパイルには、サンディスク X110 Series SSD 64GB（読込 505MB/s、書込 445MB/s）SD6SB1M-064G-1022I を外付けディスクケース経由で使っています。NFSが使える環境なら、NFSを使い、pkgsrcの展開をNFSサーバ側で実行する方法もあります。RPIにSSDを接続した場合、OSの種類と関係なく、RPI基板の個体差により、SSDが壊れる場合があるので十分注意してください。

18.23. 液晶ディスプレイ

液晶キット (<http://www.aitendo.com/page/28>) で表示できています。

aitendoの液晶キットはモデルチェンジした新型になっています。On-Lap 1302でHDMI出力を確認できました。HDMI-VGA変換ケーブルを利用する場合、MSDOS領域にある設定ファイルcmdline.txtで解像度を指定してください。

<https://twitter.com/oshimyja/status/399577939575963648>

とりあえずうちの1024x768の液晶の場合、`hdmi_group=2 hdmi_mode=16`の2行をconfig.txtに書いただけ。なんと単純。disabl

18.24. inode

inodeが足りない場合は、ファイルシステムを作り直してください。

```
# newfs -n 500000 -b 4096 /dev/rvnd0a
```

18.25. bytebench

おおしまさん(@oshimyja)がbytebenchの結果を測定してくれました。

<https://twitter.com/oshimyja/status/400306733035184129/photo/1>
[status/400303304573341696/photo/1](https://twitter.com/oshimyja/status/400303304573341696/photo/1)

<https://twitter.com/oshimyja/>

18.26. 壁紙

おおしまさん(@oshimyja)ありがとうございます。

<http://www.yagoto-urayama.jp/~oshimaya/netbsd/Proudly/2013/>

--

18.27. パーティションサイズをSDカードに合わせる

2GB以上のSDカードを利用している場合、パーティションサイズをSDカードに合わせることができます。この手順はカードの内容が消えてしまう可能性もあるため、重要なデータはバックアップをとるようにしてください。

手順は、http://wiki.netbsd.org/ports/evbarm/raspberry_pi/ のGrowing the root file-systemにあります。

18.27.1. シングルユーザでの起動

1. /etc/rc.confのrc_configured=YESをNOにして起動します。
2. 戻すときはmount / ;vi /etc/rc.conf でNOをYESに変更してrebootします。

18.28. 参考URL

- http://wiki.netbsd.org/ports/evbarm/raspberry_pi/
- NetBSD Guide <http://www.netbsd.org/docs/guide/en/>
- NetBSD/RPiで遊ぶ(SDカードへの書き込み回数を気にしつつ) <http://hachulog.blogspot.jp/2013/03/netbsd-rpi.html>
- <http://www.raspberrypi.org/phpBB3/viewforum.php?f=86> NetBSDフォーラム
- <http://www.raspberrypi.org/phpBB3/viewforum.php?f=82> 日本語フォーラム

19. sphinxのドキュメントをlatex経由でpdfに変換する

19.1. sphinxのインストール

```
# pkg_add py38-sphinx
# ln -s /usr/pkg/bin/sphinx-build-3.8 /usr/pkg/bin/sphinx-build
# which sphinx-build
/usr/pkg/bin/sphinx-build
```

19.2. sphinxに必要なlatex環境インストール

```
# pkg_add dvipdfmx
# pkg_add latexmk
# pkg_add tex-platex
# pkg_add texlive-collection-langjapanese
# pkg_add texlive-collection-fontutils
# pkg_add tex-cmap
# pkg_add tex-fancyhdr
# pkg_add tex-titlesec
# pkg_add tex-tabulary
# pkg_add tex-varwidth
# pkg_add tex-framed
# pkg_add tex-float
# pkg_add tex-wrapfig
# pkg_add tex-parskip
# pkg_add tex-upquote
# pkg_add tex-capt-of
# pkg_add tex-needspace
# pkg_add tex-kvsetkeys
# pkg_add tex-geometry
# pkg_add tex-hyppcap
# pkg_add tex-hyperref
# pkg_add py-sphinxcontrib-svg2pdfconverter
# pkg_add tex-tex-gyre
# pkg_add py-blockdiag
```

19.3. dvipdfmx設定変更

```
# cd /usr/pkg/etc/texmf/dvipdfm
diff -u -r1.1 dvipdfmx.cfg
--- dvipdfmx.cfg      2021/02/03 08:55:35    1.1
+++ dvipdfmx.cfg      2021/02/03 08:56:21
@@ -215,7 +215,7 @@
 %f psfonts.map

 %% Put additional fontmap files here (usually for Type0 fonts)
 -%f cid-x.map
 +%f cid-x.map

 % the following file is generated by updmap(-sys) from the
 % KanjiMap entries in the updmap.cfg file.
```

19.4. sphinx でlatexpdf起動

```
% gmake latexpdf
```

19.5. uplatexを使う

「jarticle, jreport, jbook を使用していて！ LaTeX Error: Encoding scheme `JY1' unknown. と表示される場合は、それぞ

れ uarticle, ureport, ujbook という upLaTeX で使用するクラスファイルに変更します。」 [1]

```
--- perth.tex.org 2022-04-28 13:47:29.834982187 +0900
+++ perth.tex 2022-04-28 13:47:50.156470962 +0900
@@ -1,6 +1,6 @@
% Generated by Sphinx.
\def\sphinxdocclass{jsbook}
-\documentclass[a4paper,10pt,dvipdfmx]{sphinxmanual}
+\documentclass[a4paper,10pt,dvipdfmx,uplatex]{sphinxmanual}
\ifdefined\pdfpxdimen
\let\sphinxpxdimen\pdfpxdimen\else\newdimen\sphinxpxdimen
\fi \sphinxpxdimen=.75bp\relax
```

[1] <https://texwiki.texjp.org/?LaTeXのエラーメッセージ>

19.6. pip でのモジュール追加

pip は/usr/pkg/bin/pip3.10 にあります。pip+バージョンです。sphinxcontribに追加する場合の例です。

```
% pip3.10 install sphinxcontrib-blockdiag
% pip3.10 install sphinxcontrib-seqdiag
% pip3.10 install sphinxcontrib-actdiag
% pip3.10 install sphinxcontrib-nwdiag
```

20. BSDライセンス

BSDは、Berkeley Software Distributionの略称です。

1. <http://ja.wikipedia.org/wiki/BSD>
2. /usr/src/share/misc/bsd-family-tree

20.1. BSDライセンスとNetBSD

NetBSDのソースコードは、自由に配布したり売ることができます。NetBSDのソースコードから作ったバイナリを売ることができます。バイナリのソースコードを公開する義務はありません。

20.2. 2 条項 BSD ライセンス

<http://www.jp.NetBSD.org/ja/about/redistribution.html>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

20.3. 2条項BSDライセンス(訳)

<http://www.jp.NetBSD.org/ja/about/redistribution.html>

ソースおよびバイナリー形式の再配布および使用を、変更の有無にかかわらず、以下の条件を満たす場合に認める:

1. ソースコードの再配布においては、上記の著作権表示、この条件の列挙、下記の注意書きを残すこと。
2. バイナリー形式の再配布においては、上記の著作権表示、この条件の列挙、下記の注意書きを、配布物に附属した文書および/または他のものに再現させること。

このライセンスの前には著作権表示そのものが付きます。この後には注意書きが付き、このソフトウェアに関して問題が生じて、作者は責任を負わないと述べます。

21. NetBSD

NetBSDは 1 個のソースツリーをコンパイルすることで実行イメージを作ることができます。

21.1. ソースコードから作る

tar 形式のファイルをダウンロード&展開し、build.shというスクリプトでコンパイルすると、NetBSDの実行イメージができます。

このtarファイルの中には、これまでNetBSDがサポートしてきた50種類以上のハードウェアと、無数の周辺機器の様が含まれています。しかもコンパイルすると、実際にハードウェア上でNetBSDが動作します。

NetBSDのコンパイルはNetBSDでも、NetBSDではないOSでも、どのハードウェアでも、ほぼ同じ手順でコンパイルできます。(できるはずです)

```
# ftp ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/src.tar.gz
# tar xzvf src.tar.gz
# ./build.sh -U -m i386 release .... -U:root以外で作成,この場合i386向け
```

21.2. Xを含んだシステムを作る

```
# ftp ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/xsrc.tar.gz
# tar xzvf xsrc.tar.gz
# cd src
# ./build.sh -u -U -m i386 -x -X ../xsrc release ... -u:更新,-x Xも作る
```

21.3. CD-ROMイメージを作る

```
# ./build.sh -m i386 iso-image ... CD-ROMイメージ作成
```

22. pkgsrc - ソースコードからソフトウェアを作る

世界中にあるいろいろなプログラムをコンパイル・インストールする手順は、プログラムごとにまちまちです。世界中のプログラムを、すべて同じ手順でコンパイルして、インストールするためには、どのような枠組みがあればよいでしょうか？

ソースコードからプログラムをコンパイル・インストールする時、NetBSDでは主に、pkgsrcを利用します。pkgsrcでは、13000種類以上のプログラムについて、コンパイル手順を分野ごとにまとめて、収集しています。

pkgsrcの役割を挙げてみます。

1. 適切なサイトからソースコードをダウンロード展開する。
2. 適切なオプションをつけて、コンパイルする。
3. インストールする。
4. コンパイルした結果からパッケージを作る。
5. 他のマシンにパッケージをインストールする。

それではpkgsrcを実際に使ってみましょう。pkgsrc.tar.gzというファイルを展開して利用します。ここでは、すぐれたtwitterクライアントであるmikutterをインストールします。makeコマンドを実行すると、関連するソフトウェアをインストールします。

```
# cd /usr
# ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/pkgsrc.tar.gz
# tar xzvf pkgsrc.tar.gz
(cd /usr/pkgsrc/bootstrap;./bootstrap) .. NetBSD以外のOSで実行する
# cd /usr/pkgsrc/net/mikutter
# make package-install
```

pkgsrc.tar.gz ファイルの中には、12000種類以上のソフトウェアをコンパイルし、インストールする方法が含まれています。↓

22.1. gitをインストールしてみる

```
# cd /usr/pkgsrc/devel/git-base
# make install
# which git
/usr/pkg/bin/git
```

22.2. baserCMSをインストールしてみる

典型的なCMSは、この手順でインストールできます。

```
# cd /usr/pkgsrc/www/ap-php ... php54+apache
# make package-install      .... 関連するソフトウェアが全部コンパイル・インストール
# vi /usr/pkg/etc/httpd/httpd.conf
LoadModule php5_module lib/httpd/mod_php5.so
AddHandler application/x-httpd-php .php

# cd /usr/pkgsrc/converters/php-mbstring
# make package-install

# vi /usr/pkg/etc/php.ini
extension=mbstring.so

baserCMSはMySQLをインストールしなくても利用できますが、利用する場合
# cd /usr/pkgsrc/databases/php-mysql ... php+mysqlインストール
# vi /usr/pkg/etc/php.ini
extension=mysql.so

# vi /usr/pkg/etc/httpd/httpd.conf
DirectoryIndex index.php index.html

# vi /etc/rc.conf
apache=YES
# cp /usr/pkg/share/examples/rc.d/apache/etc/rc.d/apache
# /etc/rc.d/apache start

basercms.netからzipファイルをダウンロード
# cd /usr/pkg/share/httpd/htdocs
# unzip basercms-2.1.2.zip
# chown -R www.www basercms
# http://localhost/basercms
管理者のアカウントとパスワードがメールで飛んでくる！！

pkgsrcを使う場合：
# cd /usr/pkgsrc/www/php-basercms
# make package-install
```

22.2.1. SSL設定

証明書のファイルを指定して、httpd.confのコメントを外して、apacheを再起動します。

```
/usr/pkg/etc/httpd/httpd-ssl.conf
SSLCertificateFile
SSLCertificateKeyFile
SSLCertificateChainFile

/usr/pkg/etc/httpd/httpd.conf
Include etc/httpd/httpd-ssl.conf ... コメントははずす
```

22.2.2. 日本語Wordpress

```
# cd /usr/pkgsrc/www/php-ja-wordpress
# make package-install
```

22.3. LibreOfficeを動かしてみる

LibreOfficeをインストールしてみましょう。

```
# cd /usr/pkgsrc/misc/libreoffice
# make package-install
: 9時間くらいかかります。
# which libreoffice
/usr/pkg/bin/libreoffice
```

22.4. IRCクライアントを動かしてみる

IRCクライアントとして hexchatをインストールします。

```
# cd /usr/pkgsrc/chat/hexchat
# make package-install
#hexchat
# サーバ選択画面で http://Libera.Chat 指定します。
# チャンネルに#netbsd を指定します。
# 設定→設定→インタフェース→外観→一般→フォントでフォント選んで、右端の表示が切れるのを直します。
```

22.5. aws/kubectl コマンド

AWSを操作するコマンドはnet/py-awscliにあります。

```
# cd /usr/pkgsrc/net/py-awscli
# make package-install
# which aws
/usr/pkg/bin/aws
% aws
Note: AWS CLI version 2, the latest major version of the AWS CLI, is now stable and recommended for general use. F
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
```

kubectl は net/kubectl にあります。

```
# cd /usr/pkgsrc/net/kubectl
# make package-install
# which kubectl
/usr/pkg/bin/kubectl
% kubectl
kubectl controls the Kubernetes cluster manager.

Find more information at: https://kubernetes.io/docs/reference/kubectl/overview/
```

22.6. 依存しているパッケージを調べる

```
cd /usr/pkgsrc/pkgtools/revbump
make package-install
finddepends lang/rust .... rustに依存しているパッケージを調べる
```

22.7. インストールするソフトウェアのライセンスを意識する

あるソフトウェアのソースコードをどのように取り扱えばいいのかは、ソフトウェアに含まれるライセンスに書かれています。GNUやBSDやMITやApacheなど有名なライセンスもあれば、有名なライセンスを少しだけ入れ替えて、目的にあったライセンスに作り替えたものなど、まちまちです。pkgsrcでは、pkgsrcに含まれるソフトウェアのライセンスを収集しています。実際に見てみましょう。

```
% cd /usr/pkgsrc/licenses ... ライセンス条項が集まっている
% ls |wc -l
228
% ls |head
2-clause-bsd
3proxy-0.5-license
CVS
acm-license
adobe-acrobat-license
adobe-flashsupport-license
amap-license
amaya-license
amazon-software-license
amiwm-license
:
```

特定のライセンスを持つソフトウェアのインストールを許可するかどうかは、/etc/mk.conf ファイルで定義します。星の数ほどあるソフトウェアのライセンスを受け入れるかどうかを、自分で決めることができます。

```
% grep ACCEPTABLE /etc/mk.conf |head
ACCEPTABLE_LICENSES+= ruby-license
ACCEPTABLE_LICENSES+= xv-license
ACCEPTABLE_LICENSES+= mplayer-codec-license
```

```
ACCEPTABLE_LICENSES+= flash-license
ACCEPTABLE_LICENSES+= adobe-acrobat-license
ACCEPTABLE_LICENSES+= adobe-flashsupport-license
ACCEPTABLE_LICENSES+= skype-license
ACCEPTABLE_LICENSES+= lha-license
ACCEPTABLE_LICENSES+= opera-eula
ACCEPTABLE_LICENSES+= lame-license
```

22.8. pkgsrc/packages

コンパイルしたパッケージは、pkgsrc/packages以下に生成されます。

```
% cd /usr/pkgsrc/packages/All/
% ls *.tgz |head
GConf-2.32.4nb7.tgz
GConf-ui-2.32.4nb11.tgz
ORBit2-2.14.19nb4.tgz
SDL-1.2.15nb7.tgz
SDL_mixer-1.2.12nb5.tgz
acroread9-jpnfont-9.1.tgz
:
# pkg_add gedit-2.30.4nb17.tgz ... インストール
# pkg_info ... 一覧表示
# pkg_del gedit ... 削除
```

22.9. pkgsrcに何か追加したい

```
# cd /usr/pkgsrc/pkgtools/url2pkg
# make package-install
# cd /usr/pkgsrc/ジャンル/名前
# url2pkg ダウンロードURL
Makefileとかができる
```

22.10. /usr/pkgsrc以下のメンテナンス

```
# cd /usr/pkgsrc/pkgtools/lintpkgsrc
# make package-install
# cd /usr/pkgsrc; cvs update -PAd ... /usr/pkgsrcを最新にする
# lintpkgsrc -pr .... 古くなったバイナリパッケージを消す
# lintpkgsrc -or .... 古くなったソースファイルを消す
# lintpkgsrc -mr .... ソースファイルのチェックサムが/usr/pkgsrcと合っているか
```

22.11. pkgsrcの更新

pkg_chkを使う方法

```
# cd /usr/pkgsrc/pkgtools/pkg_chk
# make package-install
# cd /usr/pkgsrc
# cvs update -PAd
# pkg_chk -u .... 古いパッケージをコンパイルして更新する
```

pkg_rolling-replaceを使う方法:依存関係に従って更新する

```
# cd /usr/pkgsrc/pkgtools/pkg_rolling-replace
# make package-install
# cd /usr/pkgsrc
# cvs update -PAd
# pkg_rolling-replace -u
```

22.12. ソースコードの更新

```
http://cvsweb.NetBSD.org/
# cd src
# cvs update -PAd ... 最新に更新
# cvs update -Pd -r netbsd-7 ... NetBSD7.0
```

```
# cd pkgsrc
# cvs update -PAd ... 最新に更新
# cvs update -Pd -r pkgsrc-2015Q3 ... 2015Q3に更新
```

22.13. バグレポート・追加差分

<http://www.NetBSD.org> → Support → Report a bug / Query bug database.

22.14. The Attic Museum

https://wiki.netbsd.org/attic_museum

メンテナンスするのがつらくなってきた機能を削除します。yurexとか。

23. NetBSDとブース展示

日本NetBSDユーザーグループは、日本各地のオープンソースイベントに参加し、ブース出展とセミナー枠を利用して、NetBSD関連の情報をまとめています。オープンソースカンファレンスへの積極的な参加が認められ、2014年2月に「第1回OSCアワード」を受賞しています。

23.1. ブース出展

オープンソース関連のイベントでは、たいてい幅1.8m程度の長机と椅子二つ程度のブースを出展します。各地域でのイベント開催に合わせて、最新の活動成果を展示しようとしています。

23.2. セミナー枠

セミナー枠では、NetBSDに関する情報を紙にまとめて配布して、出版物でカバーできないような情報をイベント毎にまとめています。開催地にある電子部品店・コンピュータショップ・古書店・クラフトビールバー等、生活に必要な情報もまとめています。

23.3. シール関連まとめ

NetBSDブースでは、NetBSDのシールや、NetBSDがサポートしている・サポートしようとしている・みんなが好きで利用しているソフトウェアに関連したシールを持ち寄って配っています。OSの展示は単調になりがちで、OS開発やNetBSDについて通りすがりの数秒で理解してもらうのは不可能でしたが、シールなら数秒で何かわかってもらえます。かさばらないので、誰にも受け取ってもらいやすく、優れたデザインのシールに人気が出ると、ブース全体に活気が生まれて、思いもよらない進展を呼ぶことがあります。

みくったーシールずかん <http://togetter.com/li/566230>
らころこシール作成の記録 <http://togetter.com/li/554138>

24. ルナ式練習帳、またはLunaの薄い本2026 [1]

「私が誰かは わかっているはずだ。」その声は天使の声だった。

--- さまよえる天使 [2] [3] バーナード マラマッド [4]





24.1. History and Background of LUNA

The LUNA hardware had two different operating systems; a 4.3BSD derivative and a SVR3 variant. The first one, named UNIOS-B, was a port of Integrated Solution Inc. UNIX product. ISI manufactured m68k based VME UNIX boxes. Their OS had an interesting feature of TRFS (Translucent Remote File System) as well as the popular SMI's NFS. The paper of TRFS was published at USENIX Technical Conference (late '80, details unknown in this moment). [174]

24.2. nono - LUNA-I emulator

「nono は NetBSD とかで動作する LUNA-I とかのエミュレータです。でもまだ動きません。」 [167]

nono 0.0.3 (2020/05/16) 置いときますね。

<http://pastel-flower.jp/~isaki/nono/>

なんちゃってROM用意したので、実機ROMなくても一応起動はすると思う。けど起動しかできないのと、こっからどうしたもんか

「nonoさんが実機ROMなしでも起動するようなのでとりあえず最小インストールイメージを置きました」 [168]

NetBSD/luna68k 9.0 minimam liveimage 20200518版

<http://teokurebsd.org/netbsd/liveimage/20200518-luna68k/>

- pkgsrc経由でのnonoインストール

```
pkgsrc/emulators/nono
make package-install
https://gnats.netbsd.org/55761
https://twitter.com/isaki68k/status/1315996525919518724
http://www.pastel-flower.jp/~isaki/NetBSD/patch/pkgsrc-nono-20201013.diff
http://www.pastel-flower.jp/~isaki/NetBSD/patch/nono-20201013.diff
add /etc/mk.conf
ACCEPTABLE_LICENSES+= nono-license
cd /usr/pkgsrc/emulators/nono;make ;make package-install
```

- nonoからのNetBSD/luna68k liveimage起動 [169]

1) nono-0.1.1 をダウンロード

<https://twitter.com/isaki68k/status/1261646479816404992>

2) 展開して doc/index.html を読んでビルド

3) liveimage をダウンロードして gunzip

4) nono.cfg を作って置く

<https://gist.github.com/tsutsui/340546bdc064cee786ed2473fb510463>

5) wx/nono で実行

6) Emulated ROM上で以下のコマンドを実行

```
k
[enter]
[enter]
d
boot
g
x
```

```
vmtype=luna
```

```
#ethernet-hostdriver=tap
```

```
#prom-use-rom=0 #外部ROMを指定しなければ内蔵なんちゃってROMで上がるので指定しなくても動く
```

```
#spc0-id6-writeprotect=1 #ディスクライトプロテクト。デモとかで^Cで落とす用。
```

```
spc0-id6-image=hd,liveimage-luna68k-raw-20200518.img
```

- Luna88Kの起動

「設定ファイルでvmtype=luna88kにして、0/luna88kのリリースセットの中のboot を-Aオプションで指定とかまでは出来ます。」 [170]

「it was made from scratch.」 [171]

#OpenBSD/luna88k 6.8-current runs on nono-0.1.4 on #OpenBSD/amd64. Now I can login to virtual luna88k machine! Great! [178]

For anyone interested in nono and luna88k, I put OpenBSD/luna88k live image. (990MB gzip'ed, 2.0G uncompressed) Set this image as spc0-id6-image in nono.cfg, and start nono with OpenBSD/luna88k bootloader, i.e. "nono (other options) -A boot" [179]

```
#VER=6.8
VER=snapshots
ftp https://cdn.openbsd.org/pub/OpenBSD/${VER}/luna88k/miniroot68.fs
ftp https://cdn.openbsd.org/pub/OpenBSD/${VER}/luna88k/bsd
ftp https://cdn.openbsd.org/pub/OpenBSD/${VER}/luna88k/bsd.rd
nono -A bsd.rd
```

```
vmtype=luna88k
#luna-dipsw1=11111111 #ディップスイッチの初期値設定
#ethernet-hostdriver=tap
#prom-use-rom=0
ram-size=64
spc0-id6-image=hd,spc0-id6-image
```

うゑーい、進んだー(° ∇ °)ー!! [173]

MFP通過した。 [175]

ROMやっとな動いた(´Д`) [180]

- library_aslr [181]

/etc/rc.conf.local に library_aslr=NO と書いておくと reordering libraries をスキップします。起動後であれば、 # rcctl disable library_aslr でも良いです。man.openbsd.org/rc.conf

24.3. LUNA前夜 - 誕生と再生のためのテーマ

オムロン株式会社が1990年代初めに発売していたワークステーションLUNA。

「オムロンはインターネットの基礎となるUNIXにパイオニアとして取り組んできました。最初のマシンは1984年に出荷を開始した「スーパーメイト」と言うモトローラ社の68000をCPUとして使ったワークステーションでした。、当時サンマイクロシステムズが設立されたばかりで、10MHzのチップを使っていたのですが、我々は国産の最新の12.5MHzのチップを使い、当時の最高速マシンとして登場しました。

当然UNIXを使うと自然にインターネットを使う事になるのですが、UNIXそのものが技術者しか使えないものだったため、技術者同士の通信手段として細々と使われていたに過ぎません。しかし、それでもUNIXの将来性に気づいていた我々は草の根的にオムロンの中でネットワークを拡大していきました。シグマワークステーション、LUNAワークステーションとマシンは進化していききましたが、その一方で「オムロン・インターネット」が着々と規模を拡大していました。最終的には全国30個所以上、2000人が使うネットワークにまで成長しました。当時はだれもこれが「イントラネット」だと言う意識は無く、ひたすら便利なネットワークとして整備を続けました。もちろん今ではこれが典型的なイントラネットである事は間違いありません。しかし、この段階では依然として技術者専用の情報インフラでありました。草の根的な従って統率の無いものでした。」 [50]

24.3.1. スーパーメイト

LUNAは、SX-9100以降の愛称なのですが、その前のSX-8700の時代は、スーパーメイトという愛称でした。 [149]

そしてソースリーディングを開始して約半年後、いよいよオムロン標準の16ビットボード(68000MPU)を改造して68451MMUを追加したボードコンピュータに、UNIXを移植する作業がスタートしました。開発環境は、EXORmacs上のIdrisを用い、移植中のOSのロードモジュールは、RS-232C接続で、実機にダウンロードしました。

そして、AH(アドバンストハード)プロジェクトと合流して開発したのが、68000搭載のUNIXワークステーションSX-8700でした。当時はUNIX System III注19であり、開発環境としてVAX-780上のSystem IIIを使用しました。VAXと68000のバイトオーダーが逆だったので苦労しました。このマシンがスーパーメイトという商品名で、1984年春のマイコンショーで、オムロンのコンピュータとして華やかにデビューしたのです。そして間もなく発表されたUNIX System Vを移植し直し、その年の秋に出荷が開始されました。 [150]

24.3.2. トラック一台分

引越しのため、収集していた古いワークステーションを廃棄している（しようとしている）ところです。そんななか、希少マシン？はNetBSDの移植に使用されていることを知り、事務局様を通じて、junk-jaへのポストをお願いしました。 [11]

24.3.3. 1985年12月20日

SEA設立総会の夜。「1985年12月20日という日付は、日本のソフトウェアの歴史に永久に記録されよう。」(c)岸田さん 「詳しくはシグマせんとのこと」 [54]

24.3.4. 1986年10月

「SIGMA サイドで作っていたオムロンのLUNA マシンは、そのころ、まだ影も形もなかった。わたしがソニーにアドバイスしたマーケティング戦略は、とりあえず最初のロットでできた何十台かのマシンを日本全国の大学の研究者に無料で配って使ってもらおうというもの。これもみごとに図にあたってと思います。」 [53]

24.3.5. 1982年

「一九八二(昭和五七)年頃の話。ワークステーション開発部長をしていた市原達朗は、その利用法を考えているうちに、ワークステーションを使った産学連携を思いついた。数学のノーベル賞とも言われるフィールズ賞の受賞者・広中平祐をトップに、全国から一〇〇人の教授を選んで、一人一〇大、合計一〇〇〇台のワークステーションを寄付し、それをネットワークで結んで研究成果を無償解法してもらう。そしてそれを企業が事業に活用するというのがその趣旨だった。協力してもらうのは立石電機のほかに、東芝、日立製作所、IBMといった企業四社を想定していた。」 [156]

24.4. LUNAシリーズ概要

24.4.1. SX-9100

1987年発表 for Σプロジェクト [36] [51] [86] 「札幌Σサブセンターに設置されて、地場企業によるΣ CAI ソフトウェアの開発に利用されていた。」 [52]

24.4.2. LUNA [13]

1. 1989年発表 MC68030 20MHz
2. 起動動画 [45] [46] [38]

シリーズ構成 [65]

ディスクレス	ベーシック	スタンダード	スタンダード	ファイルサーバ	ハイエンド
DT10	DT20/25	DT30/35	DT32/37	DT40/45	DT50/55
4MB	4MB	4MB	8MB	4MB	8MB
•	70MB	100MB	100MB	172MB	172MB
LAN	•	LAN	LAN	LAN	LAN
ブラック	ブラック	ホワイト	ホワイト	ホワイト	ホワイト
55万円	88/103万	115/135万円	140/160万円	165/185万円	190/210万円

- PC98インターフェースはホワイトタイプに装備,DT20/25はオプション
- DT25,35,37は、フロッピーの代わりにテープストリーマ付き

24.4.3. オプション

1. ビットマップボード:モノクロ:2048x1024,1プレーン
2. ビットマップボード:カラー:2048x1024,4プレーン,4096色中16色/16階調同時表示
3. 増設メモリボード:最大4MBx2枚
4. LANボード:イーサネット,チーパネット(DT20/25)
5. PC-98インターフェースボード
6. GPXボード: X.25,GP-1B

24.4.4. UNI-OS [37]

UniOS-B

Unix4.3BSDを移植したもの。Lunaで稼動。

UniOS-U

UnixAT&TSystemV R2.1をベースに4.2BSDの機能等を付加し、移植したもの。Luna、Luna-IIで稼動。

UniOS-Σ

ΣOS-VOR1準拠したもの。Luna-Σで稼動。(要出典:Luna-Σという呼び方) [83]

UniOS-Mach

Machをベースに移植したもの。Luna-II、Luna-88Kで稼動。

本校のワークステーションはオムロン株式会社の「LUNA」(DT65及びFS180)というもので、CPUに「MC68030」(メインメモリー16MB)を、基本ソフトウェアに統合化OS(後述します)である「Unios-U」を採用した高性能なものです。(注釈:この部分を読めば、最近のコンピュータの進化が実感できますね) [66]

24.4.5. LUNAI

- 1991/6 MC68040 25MHz
- 68040を搭載したワークステーションLUNA-IIのハードウェア
- 互換性を重視し、CISC CPUを採用したワークステーションについて [21]
- カーネル起動問題

シリーズ構成 [73]

DT2460	DT2465	DT2660	DT2665
8/16MB	8/16MB	8/16MB	8/16MB
250MB	250MB	250MB	250MB

- PC98インターフェースを2スロット装備
- DT2465,2665は、フロッピーの代わりにテープストリーマ付き
- イーサネット/チーパネット(標準)+イーサネット(オプション)

- ビットマップボード:モノクロ:2048x1024,1プレーン
- ビットマップボード:カラー:2048x1024,4プレーン,4096色中16色同時表示
- ビットマップボード:カラー:2048x1024,8プレーン,1670万色中256色同時表示

88Kでも88K2でも、hwplanebits(=ROMモニタのワークエリアの値)は、PW7131(8bpp) → 8 PW7102(4bpp) → 1 となる。 [101] [102] [103]

24.4.6. LUNA88k [10]

- モトローラRISC CPU MC88100(m88k)を採用
- マルチCPU対応(最大4つ) 1CPU時25Mips ,4CPU時100Mips
- Mach2.5,X11.4/X11R5(Luna88K2),Wnn4.1,Motif1.1.4
- PC-98用バス対応
- OpenBSDはm88kの実機とtoolchainがメンテナンスされている唯一のBSD
- 1992/9 「マルチRISCワークステーションLUNA-88K2 - 33MHz MC88100 CPUを最大4個搭載したマルチRISCワークステーションについて」 [16]
- 88Kと88K2ではNVRAM/Timekeeperが違う。 [84]
- 起動動画 [30] [48] [49]
- ユニマガ紹介記事 [74] と、製品仕様 [75] と、まとめ [71]
- miod@openbsd.org さんのOpenBSD/luna88k ページ [90]
- LUNA-88K2 姉妹生存報告。10月にリリースされた #OpenBSD 6.8 [176]
- MC88100 バグ対応の一部 [177]

シリーズ構成

[73]

DT8840	DT8860
8/16MB	32/64MB
250MB	250MB
270万円	350万円

24.4.7. omron3

omron3.sp.cs.cmu.edu (オムロン製 LUNA88k) は 1990年から1997年の間 CMU の日本語コンピュータ環境を提供するべくボランティア達によって運用されてきた計算機です。1997年5月をもって komachi.sp.cs.cmu.edu (Intel Pentium 120Mhz FreeBSD) に役目を引き継ぎ引退しました。 [68]

24.4.8. OEM版

「LUNAのシグマOSのやつで日本無線からでていたOEMのワークステーションというやつを使っていたことがあります。なんか日本語フォントがX-Window立ち上げなくても使えたようなおぼろげな記憶。銀座にあったオムロンのセミナー会場で講習をうけたのだけど、そこではOEMでなくて普通のLUNAだったからなんかちがってとまどったような...」 [64]

24.4.9. LUNA2010 [78]

Introduces Omron Electronic BV's Luna 2010, a multiprocessor Unix workstation that supports configurations of up to four 88110 CPUs. Compatibility with Data General's DG/UX 5.4 Release 2.10 operating system; Other features; Prices. [79]

そして、新しいワークステーション用のチップセットのコードネームにAsteroidという名前をつけました。火星と木星の間にある無数にある小惑星群のことです。今から考えると、何でこんな名前を付けたのだろうかを反省してしまいますが、とにかくそういう名前をつけてしまいました。 [77]

- 88110
- 1993/9ごろ
- DC/UX5.4.X

24.5. OSCを中心とするイベント駆動開発

1. 動きそうなLUNAを探す
2. ハードウェアを整備する
3. ソフトウェアを書くために必要な情報を交換する
4. 行き詰ったらツイッターで相談する
5. 定期的にイベントで展示する
6. 昔使っていた人に直接話を聞く
7. 集めた情報を整理して、公開する

24.5.1. OSC2011Kansai@kyoto - LUNA復活

NetBSD/m68k will never die!

当日いきなりSun/NEWS/Luna展示 [31]

2011/7/16のコミットメッセージ [35]

Revive NetBSD/luna68k.

Even after almost a lost decade since NetBSD/luna68k was switched to using ELF format by default back in 2001, actually only one fix (bus.h) is required for a GENERIC kernel itself to get multiuser login: prompt on a real hardware. Hurrahhh!!!

Demonstrated with a working Xorg mono server on the NetBSD booth at Open Source Conference 2011 Kansai @ Kyoto:
<http://www.ospn.jp/osc2011-kyoto/>

"Very impressed," commented by Tomoko YOSHIDA, Program Committee Chair of the Conference, and some other OMRON guys.

Special Thanks to Tadashi Okamura, for providing a working SX-9100/DT "LUNA" for this mission.

24.5.2. なぜNetBSD/luna68kなのか

LUNAを使っていたわけでもないのになぜNetBSD/luna68kにこだわるのか。それはNetBSDのyamt-kmemブランチマージ作業の際の話にまで遡る。 [91] [92]

違った yamt-km のほうだった orz [93] [94] yamt-km では hp300由来の m68k pmap でカーネルKVA用のセグメントテーブルをKVAの最上位に移動する必要があった。大部分のm68kではKVA空間上位は空いていたが luna68kだけは 0x40000000以降のデバイスアクセスにTTレジスタを使っていた。 [95] で、hp300由来のpmapのセグメントテーブルとページテーブルの初期化は壮絶に何をやっているのかさっぱりわからない記述になっていて、かつ030と040は別の初期化が必要で、yamt-kmマージ当初はyamtさんがそれなりに書き換えたけれど誰もテストしていなかったわけですよ [96] その後 NetBSD 4.0 が出る前に yamt-km マージで動かなくなっていた atari を修正して、そのあとをm68k全部のpmap初期化をすべて読み解いてそれぞれのpmap_bootstrap.c

を初期化意図が読み取れるようにゴリゴリ書きなおしたわけなんですよ [97] 実機テストできない機種ソースを4つも5つも書きなおして、1年後に見直すと致命的なtypoがあったりして、誰も持ってないマシンのコードなんか何の意味があるのか消してしまえなどと言われて、でもOSC2011京都で入手したluna68k実機ではそのままのソースで起動した、というお話 [98]

24.5.3. OSC2011Kansai@kyotoの波紋

- 「オムロンからLUNA88Kが発掘された」 from よしだともこ先生 [29]
- 「LUNA88KはOpenBSD開発者の方へ」→あおやさんと連絡がとれる
- 「ツイッターで『LUNA-IIはありませんか』とつぶやくと」
- 「ふと、横を見ると『LUNA II』と書いたマシンが。。。」

24.5.4. LunaII対応

1. 同じオペランドで68030と68040で違う命令の罫
2. %tt1 (PA/VA透過変換レジスタ)設定値修正
3. M68040共通部分の修正適用
4. 外付けSCSIアタッチ追加
5. LCD表示を「SX-9100/DT2」に変更

24.5.5. KOF2011 - LUNAII展示

- 「NetBSDが謎マシンを動かす理由＝そこに山があるからw 」 [24]
- 「明日11日(金)からのKOFのNetBSDブースで OSC京都のOMRON LUNA展示の後に発見された LUNA-IIで動くNetBSD/luna68kを展示します。」
- 「LUNA資料は手書きだ」
- 「NetBSD/luna68k画面表示の裏でひっそりと活躍する自作LUNA-II内蔵型B/WビデオtoVGA変換。」
- 「「そんなことよりそれはなんだ」と言われそうなLUNAならぬ初代SX-9100 Mr.文具セット。裏によしだ先生サイン(?)入りの貴重品。」

24.5.6. isibootd(8)

LUNA専用ネットブートサーバープログラムisibootd(8)をNetBSDツリーにコミット。

24.5.7. FPU判別ルーチン

1. ローエンド、ベーシックタイプは68881
2. サーバータイプは68882

24.5.8. OSC2012Kansai@Kyoto

1. NetBSD/luna68k近況 [58]
2. wsconsコンソールフレームバッファ修正 (OpenBSD/luna88kから)
3. オムロンフォントで表示
4. 電源トラブル: 電解コンデンサ全交換
5. PROM起動仕様 HDDから起動する条件調査
6. bootarg問題
7. SSD on LUNA
8. Xorgサーバー

24.5.9. 円頓寺LUNAエンカウント

NBUG2013/2月例会。いきなりLuna68K/Luna88k/BigNEWSがNBUG例会にタクシーで持ち込まれる。「掲示板でLUNA88kをNBUG例会に持ち込もうか聞いている人がある」と噂になっていたその人だった。 [56]

24.5.10. OSC2013Tokushima

1. NetBSDこの20年 [55]
2. NetBSD/luna68kブートルoader起動展示 [8]

```
Module Name:    src
Committed By:  tsutsui
Date:          Sat Jan  5 17:44:25 UTC 2013
```

```
Added Files:
  src/sys/arch/luna68k/include: loadfile_machdep.h
```

```
src/sys/arch/luna68k/stand/boot: Makefile autoconf.c bmc.c bmd.c boot.c
boot.ldscript conf.c cons.c device.h devopen.c disklabel.c font.c
getline.c init_main.c ioconf.c kbd.c kbdreg.h locore.S machdep.c
omron_disklabel.h parse.c preset.h prf.c rcvbuf.h romcons.c
romvec.h samachdep.h sc.c screen.c scsi.c scsireg.h scsivar.h sd.c
sio.c sioreg.h status.h stinger.h trap.c ufs_disksubr.c vectors.h
version
```

Log Message:

First cut at NetBSD/luna68k native bootloader.

Based on 4.4BSD-Lite2/luna68k "Stinger" loader revision "Phase-31"

<http://svnweb.freebsd.org/csrc/sys/luna68k/stand/>

and MI libsa glue stuff are taken from hp300 etc.

Tested on LUNA-I and old DK315C SCSI disk drive.

LUNA's monitor PROM can load only an a.out binary in 4.3BSD FFS partition (i.e. created by "newfs -0 0") on disks with OMRON's UniOS disklabel, but now we can load an ELF kernel in root partition via this bootloader. (See luna68k/disksubr.c for details of UniOS label)

TODO:

- LUNA-II support (check 68040 to adjust cpuspeed for DELAY())
- secondary SCSI support for LUNA-II
- netboot via le(4) (should be trivial)
- support boot options on bootloader prompt
- bootinfo (passing info about booted device and kernel symbols)
- support "press return to boot now, any other key for boot menu" method like x86 bootloader (needs cnsan() like functions)
- tapeboot (anyone wants it?)

24.5.11. OSC2013Nagoya - Luna88K&Luna68K

- あおやまさんと江富さんによるLuna88K/Luna68K完全動態展示 [57]
- Luna88K2 & Luna68K プロトタイプ7号機
- OpenBSD/luna88k開発者あおやまさんによるプレゼンテーション [10]

24.5.12. OSC2013 Kansai@Kyoto

非力なマシンで最新のOSを動かすためには、大変な努力と工夫が必要です。その展示を実現させた方は、その努力と工夫を楽しんでおられたというわけです。 [100]

24.5.13. NBUG 2013/9

- OpenBSD/luna88k 近況報告

24.5.14. KOF2013

- 関西オープンソース2013NetBSDブースの記録 [117]

24.5.15. OSC2014 Kansai@kyoto

- OSC2014京都 NetBSDブース展示への道 [134]
- OSC2014京都 NetBSDブースの記録 [135]

24.5.16. LUNAグッズ

- LUNAグッズ持参でブースに遊びに来てくれる元関係者の方が！
- シールとフロッピー [59]
- たれまく
- ペンセット [60]
- ペンケースとバンダナ [76] [121] [130]
- dpNote - 図形グッズ：シール・定規
- ホッチキス [133]
- トレーナー [144]
- LUNAII ボールペン

24.5.17. LUNA関連書籍

LUNAユーザーグループとは何か - mikutterの薄い本 [81] を会場に忘れたら、なぜか一緒に送られてくる
UNIXワークステーションがわかる本 [61]

@tsutsuii LUNAの薄い本2013作れってことすね [7]

24.6. LUNAについて私が知っている二、三の事柄

24.6.1. Project Mach

Project Mach was an operating systems research project of the Carnegie Mellon University School of Computer Science from 1985 to 1994.

"It's never too late. When it's over, you get to tell the story" -- Garrison Keillor [67]

- luna88kカーネルソース [72]

24.6.2. LUNAインストール方法

- インストールマニュアル [27]
- NetBSD/luna68kの起動ディスク作り方メモ [28]

24.6.3. NetBSD/luna68Kのブート方法 [13]

2種類のブート方法 : [27]

1. PROMがUNIOS-B /a.outをロードする
2. PROMが独自プロトコルでサーバからカーネルロード

- NetBSD1.5以降 m68kはELFフォーマット移行 : どうやって起動するか
- 実機がないままソースツリーはメンテされ続ける

24.6.4. PROMモニタ

1. newfs -o で作った4.3BSD形式のFFS上のa.outカーネルは読める。
2. LUNAIIでのネットブートは無理?
3. HDD起動時の制約は?(SCSI ID, カーネルサイズ, ファイルシステム)
4. LUNAIIは外部SCSI HDDから起動できるか

24.6.5. LUNA68Kのブートローダー

1. NetBSD/luna68k の起動ディスクの作り方メモ [28]
2. なんか出た。これでデバッグできる [8]
3. native bootloader update for NetBSD/luna68k [116]

24.6.6. LUNA88Kのブートローダー

1. OpenBSD/luna88k standalone bootloader by @MiodVallat works fine on my LUNA-88K2! [107]

24.6.7. BSD広告条項

4.4BSD-Lite2由来のコードに含まれる3項目(All advertising materials ..)、広告条項削除OKについて、文書で許可を出してくれるOMRONの方がいらっしやると2-clause BSDで配布できる。

24.6.8. 電源問題

1. OMRONワークステーション LUNA-II 電源ユニット修理記 [25]
2. 「KOF本番週の日曜日に電源が不調になり急遽部品手配して展示直前に修理していた」
3. OSC2012京都前に再度補修 [26]
4. LUNA-II, LUNA-88K 電源ユニット(PTD573-51) 四級塩電解コンデンサー覧 [88]
5. 88K2は88Kより分解しやすいような気がする。 [127]

24.6.9. LUNAII

- 1MB SIMM/4MB SIMM切替→SIMM脇に謎ジャンパが
- 4bppフレームバッファのX11R5ソースは？ [85]

24.6.10. PC98-Cバス

1. 86音源ボード on LUNA [138]
2. C-bus拡張ボード on LUNA-88K2 [146]

24.6.11. UniOS-Machと西暦2000年問題

でも、同じマシンでUniOS-Mach立ち上げると時刻が変になる。昔調べたとおりdate(8)コマンドでは2000年以降の日付は設定できないので、OS内部で元々Y2K対応が甘いだけかもしれない。 [82] [84]

24.6.12. 質問日時:2009/10/28 17:51:08

会社でワークステーション（オムロン製LUNAII）を使用していますが、プリンターが不良となりました。エプソンVP-4000です。中古をさがしていますが、これと互換性のあるプリンターはないでしょうか？ [14]

もう捨てようかと思う...が、未練があり捨てられず [10]

Sometimes you get so lonely / Sometimes you get nowhere / I've lived all over the world / I've left every place / Please be mine / Share my life / Stay with me / Be my wife [5]

24.7. LUNA年表 - 月の刃

「次の日からオレとルナ先生の生活がはじまったんだ」 [62]

青：あおやまさん, 江：江富さん, 才：オムロン, 筒：筒井さん, 菅：菅原さん, モ：モトローラ

1986/10		いけない！ルナ先生連載開始
1987	Σ	SX-9100 オムロンから発表 [47]
1988	オ	グッドデザイン賞受賞 [15]
1988/7		いけない！ルナ先生連載終了
1989	オ	SX-9100/DT LUNA MC68030 20MHz
1989/3	オ	LUNAのハードウェア Omron Tech No.29 p.8-15 [19]
1990/7	オ	Luna88k Omron Tech p.81-92 [20]
1991/6	オ	LUNA-II Omron Tech No.31 p. 91-9 [21]
1991/10/11		春奈るなさんの誕生日
1991/11	モ	MC88110の存在が明らかになる [17]
1992/12	オ	LUNA-88K2 Omron Tech No.32 p.336-344 [16]
1992/12	オ	MC88110ワークステーション Omron Tech No.32 p.345-350 [18]
1993/9	オ	LUNA2010
1993/10	オ	UNIX MAGAZINE 1993.11 LUNA2010 記事 [195]
1994		いけない！ルナ先生 復刻版
1994		4.4BSD Lite luna68k
1994	オ	LUNA-IIの生産終了
1994/3	オ	LUNA2010用システム診断プログラムの開発について [80]
1998/6		NetBSD/luna68k やってるひと、いますよ。[netbsd 02006] [23]
1999/12		NetBSD/news68kマージ
2000/1/6		NetBSD/luna68kマージ
2000		いけない！ルナ先生 復刻版
2000/2/18		梶田さんluna68k起動成功 [netbsd 05132] [22]
2000/08	青	LUNA-88K2入手
2001/12	青	LUNAにOpenBSD移植決意
2002/01/27	青	OpenBSD/sparc上でm88kクロス環境構築
2002/03/29	青	シリアルコンソールでカーネルCopyright表示
2002/06/05	青	network bootでIPアドレス取得 [69]
2003/08	青	コンパイラのバグがなおったようなので再開
2003/09/20	青	tarのオプションを間違えてソース消去、CVS導入
2003/10/05	青	なんとかもとの状態に戻る
2003/12/10	青	NFS rootでシングルユーザ&マルチユーザ

2004/02/17	青	SCSI動作
2004/03/21	青	Miod Vallatさん(OpenBSD/mvme88kのport maintainer)に見つかる [70]
2004/04/21	青	OpenBSD本家treeにcommit
2004/11/01	青	OpenBSD 3.6: 初の正式リリース
2007/08/31	青	LUNA-88K2の電源ユニット故障により起動できなくなる
2007/9/5	青	Luna88K検索願い [nbug:10540]
2009/10/28		Yahoo知恵袋にLUNAIIに関する]質問が [14]
2011/07	筒	OSC2011Kansai@KyotoでLUNA/NEWS/Sun3展示 [31]
2011/7	筒	NetBSD/luna68k on OMRON LUNA - Bootstrap [37]
2011/07	オ	LUNA88Kオムロンにて発掘される!
2011/08	青	ご好意により、オムロンで発掘されたLUNA-88Kが届く
2011/08/15	青	上記LUNA-88Kの電源ユニットを移植して復活
2011	筒	KOFでLUNA-II展示 [9]
2012/05/01	青	OpenBSD 5.1: 久しぶりの正式リリース
2012/02/28	青	10年目にして一応マルチプロセッサ対応
2012/08/03	筒	OSC2012関西@京都でLuna&LunaII展示 [32]
2013/01	青	OpenBSD m88k portのELF化&共有ライブラリ化
2013/01/27		Luna88k(白と黒) Luna88K2Luna2010を青山さんに送る
2013/02/16	江	Luna88K&Luna68k&BigNEWSをNBUG例会に持ち込む
2013/03/09	江	Luna88kをあおやまさんに送る
2013/03/09	筒	OSC2013徳島でLuna68K展示 [33]
2013/03/19	筒	筒井さんから江富さんにLuna68k起動ディスクが送られる
2013/04/13	青	Monochrome X serverが動作
2013/06/22	青	Luna88K OSC 2013 Nagoyaで初展示。江富さんのLuna68kも初展示。 [119]
2013/08/2	筒	OSC2013京都でLuna/LunaII tw/mikutter展示。 [120]
2013/08/24	筒	OSC2013島根でLunaII+mterm-fb+mikutterd展示 [118]
2013/09/21	青	OpenBSD/luna88k近況報告 無印/4bpp/リリース [109]
2013/11/8-9	筒	KOF2013 NetBSDブースでLuna+mikutter展示 [117]
2013/12/21	青	OpenBSD/luna88k近況報告 PC98/バス 音源ボード [110]
2014/01/11	筒	native bootloader update for NetBSD/luna68k [129]
2014/03/05		いけない! ルナ先生 DVD全6巻発売開始 [114] [115]
2014/04/19	青	NBUG2014/4例会 OpenBSD/luna88k 2014/4近況報告 [125]
2014/07/05	青	OSC 2014 NagoyaでLuna88K+PC98 86音源ボード展示 [122] [123] [124]
2014/07/13	筒	LunaII+8bpp ボードでカラー表示 [126]
2014/07/20	筒	LUNA's keyboard driver changes from OpenBSD/luna88k [131]
2014/07/21	筒	luna68k 4/8bpp framebuffer as a monochrome server [132]
2014/08/01	筒	lunaII+mikutterd今年はカラーだ展示 [134]
2014/08/13	筒	LUNA framebuffer mod for LCDs without Sync on Green support [142]
2014/11/07-08	筒	関西オープンソース2014 NetBSDブース展示の記録 [143]
2014/11/29	青	yaft x LUNA [140]
2014/12/20	青	86音源ボード on LUNA [138]
2014/12/29	青	86音源ボードコードコミット [139]
2015/02/21	青	LUNA-88K2はPCカードの夢を見るか? [141]
2015/03/21	青	LUNA-88K2はPCカードの夢を見るか? <補遺> [145]
2015/05/01	青	OpenBSD 5.7リリース [148]
2015/05/22	青	C-bus拡張ボード on LUNA-88K2 [146]
2015/07/06		「いけない! ルナ先生」コラボ読切で復活 [147]
2015	青	OpenBSD/luna88k移植物語 [152]
2015/10	青	FM音源の調べ on LUNA http://www.slideshare.net/ao_kenji/nbug201510
2016/3	江	Luna68K AsiaBSDCon2016ブース展示
2016/7	筒	Luna68K PSG Z80 sound driver for PC6001 to NetBSD/luna68k OSC京都 [164]
2016/11	藤	Implimentation of 4.4BSD luna68k by Akito Fujita KOF
2016/11	筒	Luna68K PSG Z80 sound driver for PC6001 to NetBSD/luna68k OSC広島
2017/3	青江	Luna88K&Luna68K AsiaBSDCon2017ブース展示
2018/5	筒	RaSCSI + OMRON 初代LUNA 起動
2018/5	青	LUNA-88K2 OSC2018Nagoya ブース展示
2018/8	菅	LUNA68K OSC2018Kyoto ブース展示 LUNAのPSG音源でPCM再生
2018/8	筒	LUNA68K OSC2018Kyoto ブース展示 sayaka+mterm-fb
2018/7	青	LUNA-88K2 OSC2018Nagoya ブース展示 [165]

2020/3	井	nono-0.0.1リリース
2020/4		オムロン元社長・立石義雄氏逝去 [172]
2020/3	井	nono-0.0.2リリース
2020/5	井	nono-0.0.3リリース [167]
2020/6	井	nono-0.0.4リリース
2020/7	井	nono-0.1.0リリース
2020/10	井	nono-0.1.1リリース フロントパネルLED
2020/10	井	nono-0.1.2リリース
2020/11	井	nono-0.1.3リリース
2020/11	井	nono-0.1.4リリース
2020/12	井	nono-0.1.5リリース OpenBSD/Luna88k起動
2021/04	井	nono-0.1.6リリース
2021/5	青	Here come LUNA-88K emulators! [182]
2021/6	井	nono-0.2.0リリース [181]
2021/8	井	nono-0.2.1リリース
2021/9	井	nono-0.2.2リリース
2021/10	井	nono-0.2.3リリース
2021/11	井	nono-0.2.4リリース
2021/12	井	nono-0.2.5リリース
2022/2	井	nono-0.2.6リリース
2022/4	井	nono-0.3.0リリース
2022/5	筒	psgpam(4)マージ [183]
2022/6	井	nono-0.4.0リリース NetBSD/x68kブート
2022/7	井	nono-0.4.1リリース [185]
2022/7	筒	NetBSD/luna88k シングルユーザブート [184]
2022/8	井	nono-0.4.2リリース
2022/8	井	nono-0.4.3リリース
2022/9	井	lunafb(4) improvement [186]
2022/10	井	nono-0.4.4リリース [187]
2023/1	井	nono-0.5.0リリース [188]
2023/2	井	nono-0.5.1リリース [189]
2023/2	青	LUNA88K Boot 0.8 [190]
2023/3	井	nono-0.6.0リリース
2023/4	井	nono-0.6.1リリース
2023/8	井	nono-0.6.2リリース
2023/11	井	nono-0.6.3リリース
2023/11	井	nonoの話 [191]
2023/12	井	nono-0.6.4リリース
2024/2	井	nono-0.7.0リリース
2024/3	井	nono-0.7.1リリース
2024/5	井	nono-0.7.2リリース
2024/5	青	OSC2024NagoyaでLUNA88K展示 [192]
2024/7	井	nono-0.7.3リリース
2024/8	青	FUZIXを88kのXP上で起動実験
2024/10	青	FUZIX 0.5 on OpenBSD/Luna88k [193]
2024/10	井	nono-1.0.0リリース
2024/12	青	OpenBSD/amd64 での FUZIX ビルド環境構築ガイド [194]
2024/12	井	nono-1.1.0リリース
2024/12	井	nono-1.1.1リリース
2025/1	井	nono-1.1.2リリース
2025/1	井	nono-1.2.0リリース
2025/2	井	nono-1.3.0リリース
2025/3	井	nono-1.4.0リリース
2025/4	井	nono-1.4.1リリース
2025/5	井	nono-1.4.2リリース
2025/9	井	nono-1.5.0リリース
2025/10	井	nono-1.6.0リリース
2025/11	井	nono-1.6.1リリース
2025/11	井	nono-1.6.2リリース

2025/11	井	nono-1.6.3リリース
2025/12	井	nono-1.6.4リリース
2026/1	井	nono-1.6.5リリース
2026/2		ワークステーションにCバス?オムロン「LUNA」に見る1980年代末の興味深い市場戦略 [196]
2026/2		ワークステーション愛好家必見! nonoで体験するLUNA 68K/88Kの世界 [197]
2026/3	井	nono-1.6.6リリース
2026/3	井	nono-1.7.0リリース
2026/4		miod さんFirst Compiler on the Moon [198]
2026/4	井	nono-1.7.1リリース
2026/5	井	nono-1.8.0リリース SSGサポート開始

24.8. 最近のLUNA

24.8.1. メモリ64M

というわけで LUNA-IIの64MB設定でも NetBSD/luna68kカーネル起動した。これでしばらく耐久テストするか… [12] [34]

24.8.2. mlterm-fb & tw

ツイッタークライアント! [39] [40] [41] [42] [43] [44] [111]

24.8.3. mlterm-fb + mikutterd

LunaIIならmlterm-fbとmikutterdを組み合わせてタイムラインを展示できます。

24.8.4. 画像の2値化

モノクロ画面で効率的にデモ画面を作る方法: (サーベイする)

24.8.5. LUNA-88K:NVRAM and Timekeeper registers

On 'original' LUNA-88K, NVRAM contents and Timekeeper registers are

mapped on the most significant byte of each 32bit word. (i.e. 4-bytes stride) Also, add small 'Wrong year set by UniOS-Mach after Y2K' hack. [89]

24.8.6. LUNA88K謎ボード

- PWB7183 [99]
- 専用チップが載っている [102]

24.8.7. KOF2014におけるLUNA展示

LUNAフォントとSONYフォント [137]

24.8.8. yaft X LUNA

yet another frame buffer terminal [136]

24.8.9. LUNA88KのXでカラー背景

- 1280x1024の画像を作る
- ImageMajikのconvertでconvert 何とか.png -colors 32 32.xpm
- .xinitrc にxpmroot 32.xpm を追加する。

24.8.10. 老ハード介護問題

- 電源修理
- SCSI HDD確保
- ブラウン管を知らない子どもたち
- 3ボタンマウスを知らない子どもたち [128]

- 液晶接続問題 [63]
- ハード保守
- 詳しい人がいなくなる

24.8.11. pkgsrc on Luna88K/OpenBSD

Luna88K/OpenBSD 上で、pkgsrcを使っていくつかパッケージを作ってみました。

```
nono88k# ls /usr/pkgsrc/packages/All/
bash-5.2.21nb1.tgz      libiconv-1.17.tgz      pcre2-10.42.tgz
bison-3.8.2nb1.tgz     libidn2-2.3.4.tgz      perl-5.38.2.tgz
bsdtar-3.4.3.tgz       libtool-base-2.4.7nb1.tgz  pkgconf-2.0.3.tgz
bzip2-1.0.8.tgz        libunistring-1.1.tgz   tcsh-6.24.10.tgz
digest-20220214.tgz   libxml2-2.10.4nb6.tgz  tradcpp-0.5.3.tgz
fetch-1.9nb2.tgz       lz4-1.9.4.tgz          xmlcatmgr-2.2nb1.tgz
gettext-lib-0.22.3.tgz m4-1.4.19nb1.tgz       xorg-cf-files-1.0.8.tgz
gettext-tools-0.22.3.tgz mandoc-1.14.6nb1.tgz  xorgproto-2023.2.tgz
gmake-4.4.1.tgz        ncurses-6.4.tgz       xz-5.4.5.tgz
grep-3.11.tgz          ncursesw-6.4.tgz      zlib-1.2.13.tgz
libfetch-2.39nb2.tgz  nghttp2-1.58.0nb1.tgz
```

失敗する主なパッケージ

パッケージ 失敗するパッケージ

sayaka	gcc7	
nanotodon	openssl	undefined reference to `memcpy'

「それなら、なぜ、先祖代々の墓を守って山間に生活したがる農民を、ダム工事のためにおいだすんだね？それぞれの人間にそれぞれの幸福がある。それなら、なぜ、彼らを一般化の中に投げ込むんだ。君はなぜ、そういう役割をひきうけるんだ？」 [6]

24.9. 付録: OpenBSD on Motorola 88000 processors

Miod Vallatさんによる「OpenBSD on Motorola 88000 processors」のPart 3からPart 5までの概要です。

Author: Miod Vallat

Source: OpenBSD Stories (<http://miod.online.fr/software/openbsd/stories/m88k3.html>) [198]

24.9.1. PART 3: First Compiler on the Moon

24.9.1.1. Kenji Aoyama氏によるLuna-88Kへの移植

2004年3月、Paul Weissmann氏より青山健治 (Kenji Aoyama) 氏がOpenBSDをLuna-88Kへ移植しているとの報を受け、Miod氏は直ちに連絡を取りました。

- 協力体制: 青山氏はMiod氏のGCC関連の活動に感謝し、コードの公開と統合に同意しました。
- コード統合: mvme88kのコードベースを共通部分と機種固有部分に分離するリファクタリングが進められました。
- 公式採用: Apr 21, 2004、青山氏自身の手によってツリーにインポートされました。
- 継続性: このポートは現在も維持されており、青山氏は2013年より毎年日本のオープンソースカンファレンス (OSC) で展示を行っています。

24.9.1.2. コンパイラ (GCC) のデバッグとABI変更

OpenBSD 3.5リリースに向けたビルド作業中に、複数の内部コンパイラエラー (ICE) が発見されました。

- 原因: ビット演算子と論理演算子の混同や、64ビット型配列添字による最適化の失敗などが原因でした。
- 最適化バグ: -O2 指定時にスタック上のローカル変数の保存場所を誤るバグに直面しました。
- ABIの変更:
 - 問題の根本解決のため、880open ABIが定める「32バイトのスタック予約領域 (REG_PARM_STACK_SPACE)」を0にするという、ABIを破壊する変更を断行しました。
 - これにより、スタック消費量が削減され、バイナリサイズも縮小しました。変更は Jul 28, 2004 にコミットされました。

24.9.2. PART 4: A Failed Promise

24.9.2.1. OpenBSD/aviion ポートの開始

2006年4月、Miod氏はData General AViiON 400システムのドキュメントがMVME188に近いことに気づき、実機なしでの移植作業 (OpenBSD/dg88k、後のaviion) を開始しました。

- リモート開発: 米国のChris Tribo氏が所有するAV410実機を使い、海を越えた共同作業が行われました。
- 進捗:
 - May 08, 2006、約100個目のテストカーネルでマルチユーザーモードへの起動に成功しました。
 - これにより、適切なドキュメントとテスターがいれば、ハードウェアなしでもBSDを移植できることが証明されました。

24.9.3. PART 5: Good Times Bad Times

24.9.3.1. 2007年の現状と進展

2007年、各ポートの状況は一進一退でした。

- mvme88k: 187/188/8120システムは良好に稼働。
- luna88k: 青山氏の所有マシンの電源ユニットが故障し、一時停滞。
- SMPの進展: 2007年秋、88100 SMP (MVME188) の致命的バグを修正し、4CPU構成でのマルチユーザー起動に漕ぎ着けました。

24.9.3.2. 88110プロセッサのバグ特定

長年MVME197 (88110搭載) で発生していた「バイナリの最初の数命令がスキップされる」という謎のバグが解明されました。

- 解決: わずか2行のコード変更 (パイプライン設定の修正) により、Nov 20, 2007 に修正されました。
- エラッタ対策: 88110の多数のプロセッサエラッタに対し、バージョン依存のロジックで機能を無効化するなどの対策が施されました。

24.9.3.3. AViiON実機の入手

2007年末、Miod氏はeBayでAViiON 4300を入手し、Marco Peereboom氏の協力を得てフランスへ搬送しました。

- 新たな課題: 4605モデル (プロセッサあたり6つのCMMUを持つ特殊構成) のサポートにも着手し、2010年4月にマルチユーザー稼働を達成しました。

24.10. 脚注

Luna関連コメントは筒井さんに紹介してもらったものです。

- [1] カフカ式練習帳 http://www.bunshun.co.jp/cgi-bin/book_db/book_detail.cgi?isbn=9784163813301
- [2] The Angel Levine: <http://www.blackmovie-jp.com/movie/angellevine.php?act=a#.Uei7I9f75z0>
- [3] Look Back in Anger: http://en.wikipedia.org/wiki/Look_Back_in_Anger_%28song%29
- [4] バーナード・マラマッドに関する研究 <http://www.ishikawa-nct.ac.jp/lab/G/koguma/www/ehp/suzukihp.pdf>
- [5] Be My Wife: http://en.wikipedia.org/wiki/Be_My_Wife
- [6] 高橋和巳『散華』論 -生活人としての大家- , 東口昌央, 1988, <https://osaka-kyoiku.repo.nii.ac.jp/records/2081338>
- [7] <https://twitter.com/ebijun/status/231983148118970368>
- [8](1,2) NetBSD/luna68kブートローダー実装作業日記, 2013/1/4 <http://togetter.com/li/433650>
- [9] NetBSDブース @ 関西オープンソース 2011, 2011/11/13 <http://togetter.com/li/213724>
- [10](1,2,3) OpenBSD/luna88k on LUNA-88K2 <http://www.nk-home.net/~aoyama/osc2013nagoya/OpenBSD-luna88k.pdf>
- [11] トラック一台分? <http://www.jp.netbsd.org/ja/JP/ml/junk-ja/201301/msg00005.html>
- [12] というわけで <https://twitter.com/tsutsui/status/357219819289985024/photo/1>
- [13](1,2) 展示マシン紹介(3) <http://www.ceres.dti.ne.jp/tsutsui/osc2011kyoto/>
- [14](1,2) Yahoo! 知恵袋 http://detail.chiebukuro.yahoo.co.jp/qa/question_detail/q1132299146 まだ動いていたのか! 人々に勇気を与えた質問。

- [15] GOOD DESIGN AWARD <http://www.g-mark.org/award/describe/15097> … 価格にマルが一つ足りない
- [16](1,2) 「マルチRISCワークステーションLUNA-88K2 - 33MHz MC88100 CPUを最大4個搭載したマルチRISCワークステーションについて」 http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902066730130379 … これも1992年12月でさっきのと同じだから同時進行だったのかな
- [17] MC88110とは <http://en.wikipedia.org/wiki/MC88110> "... the MC88110 was ultimately unsuccessful and was used in few systems." 諸行無常
- [18] 「MC88110を搭載したワークステーションのハードウェア - 64bits, Superscalerを採用したMC88110 CPUを搭載したワークステーションのハードウェアについて」 http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902068908677809 … 幻のLUNA88K3計画なのだろうか
- [19] 「LUNA(デスクトップWS)のハードウェア 従来機に比べて小形化, 低価格化を実現したハードウェアについて」 http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902088071486407
- [20] 「マルチプロセッサワークステーションのハードウェア - RISCマルチプロセッサのワークステーションへのインプリメンテーション技術について」 http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902066853173587 実はLUNA88Kの設計のほうかLUNA-IIより先なんですよね
- [21](1,2) 「68040を搭載したワークステーションLUNA-IIのハードウェア」 http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902048488142806 … , Omron Tech 31巻 2号 91-97頁, 1991年06月
- [22] 古文書に見る現実逃避パワー <http://www.re.soum.co.jp/~fukaumi/ml/netbsd/200002/msg00122.html> …
- [23] 古文書に見る謎の痕跡 <http://www.re.soum.co.jp/~fukaumi/ml/netbsd/199806/msg00068.html>
- [24] なぜ山に登るのか <http://www.ceres.dti.ne.jp/tsutsui/kof2011/Why-enigmatic-machines.html>
- [25] OMRONワークステーション LUNA-II 電源ユニット修理記 <http://togetter.com/li/215988>
- [26] OMRONワークステーション LUNA-II 電源ユニット修理記 その2 <http://togetter.com/li/354562>
- [27](1,2) インストール方法 <http://www.jp.netbsd.org/ports/luna68k/install.html>
- [28](1,2) NetBSD/luna68k の起動ディスクの作り方メモ <https://gist.github.com/tsutsui/5196134> とかですが netboot の説明を isibootd(8) に合わせて更新するのをさぼっている (ブーメラン)
- [29] LUNA88K, オムロンにて発掘される! <http://www.jp.netbsd.org/ja/JP/ml/port-mac68k-ja/201107/msg00011.html>
- [30] Luna88k 起 動 動 画 https://twitter.com/ao_kenji/status/353469599871799296 https://twitter.com/ao_kenji/status/353476705521905664
- [31](1,2) "NetBSD/m68k will never die!" <http://www.ceres.dti.ne.jp/tsutsui/osc2011kyoto/NetBSD-m68k-will-never-die.html>
- [32] OSC2012京都NetBSDブース展示の記録 <http://togetter.com/li/350035>
- [33] OSC2013徳島 NetBSDブース展示の記録 <http://togetter.com/li/468577>
- [34] OMRONワークステーションLUNA 工作日記 <http://togetter.com/li/535307>
- [35] コミットメッセージ <http://mail-index.netbsd.org/source-changes/2011/07/16/msg024675.html>
- [36] Σプロジェクト <http://ja.wikipedia.org/wiki/Σプロジェクト>
- [37](1,2) [http://ja.wikipedia.org/wiki/Luna_\(ワークステーション\)](http://ja.wikipedia.org/wiki/Luna_(ワークステーション))
- [38] NetBSD/luna68k on OMRON LUNA - Bootstrap http://www.youtube.com/watch?v=c1_e-A90sr0
- [39] Twitter timeline on NetBSD/luna68k and mlterm-fb (final) http://www.youtube.com/watch?v=djbEw0G_LMI 2013/5/24
- [40] mlterm-fb demonstration on NetBSD/luna68k (revised) <http://www.youtube.com/watch?v=BP8AIceWgxA> 2013/5/18
- [41] Twitter timeline on NetBSD/luna68k and mlterm-fb (take 4) http://www.youtube.com/watch?v=yKKT_Z1P9Xo 2013/05/04
- [42] Twitter timeline on NetBSD/luna68k and mlterm-fb (take 3) <http://www.youtube.com/watch?v=C11Ca05scHY> 2013/05/01
- [43] Twitter timeline on NetBSD/luna68k and mlterm-fb (take 2) <http://www.youtube.com/watch?v=8sC5XpK-Hxs> 2013/04/29
- [44] Twitter timeline on NetBSD/luna68k and mlterm-fb <http://www.youtube.com/watch?v=nzD0A279mcg> 2013/04/27

- [45] NetBSD/luna68k on OMRON LUNA - Start X.Org <http://www.youtube.com/watch?v=NRh60c420Mc> 2011/07/30
- [46] mlterm-fb demonstration on NetBSD/luna68k wsfb console <http://www.youtube.com/watch?v=jHU876RexCo> 2013/04/25
- [47] ワークステーション(SX-9100)の概要 <http://jglobal.jst.go.jp/public/20090422/200902068890346915> 1987/9/30
Omron Tech p.207-213
- [48] OpenBSD/luna88k on OMRON LUNA-88K2 - bootstrap screen <http://www.youtube.com/watch?v=btwiiZw3B2s> 2013/07/06
- [49] OpenBSD/luna88k on OMRON LUNA-88K2 - starting X.org http://www.youtube.com/watch?v=_EUpSpUD0Qw 2013/07/06
- [50] オムロンのイントラネットの歴史 <http://www.masuda.org/intra/rekisi.html>
- [51] 【IT】日本のITの歴史—SONY『NEWS』の戦略(3)(1989-03-20) <http://www.miraikeikaku-shimbun.com/article/13282000.html>
- [52] さっぽろコンピュータ博物館 <http://www.sec.or.jp/electec/museum/>
- [53] 歴史的コンピュータとソフトウェアプロジェクトに関する昔話(社外公開版) <http://katsu.watanabe.name/doc/comphist/>
- [54] SEA Mail Vol.1 No.1 http://www.sea.jp/office/seamail/1986/1986_1_honan.pdf
- [55] NetBSDこの20年 <http://www.slideshare.net/tsutsui/osc2013tokushima-net-bsd20th>
- [56] 名古屋*BSDユーザグループ(NBUG)2013/2月例会の記録 <http://togetter.com/li/456972>
- [57] OSC2013名古屋 NBUG&NetBSDブース展示の記録 <http://togetter.com/li/522396>
- [58] NetBSD/luna68k 近況 <http://www.ceres.dti.ne.jp/tsutsui/osc2012kyoto/NetBSD-luna68k-updates.html>
- [59] LUNAシールとフロッピー <http://movapic.com/pic/2013062214270151c535a5bd627>
- [60] LUNAペンセットと本 <https://twitter.com/tsutsui/status/135565130372104192>
- [61] UNIXワークステーションがわかる本(LUNAの本シリーズ) <http://www.amazon.co.jp/dp/4526029963>
- [62] いけない!ルナ先生 全5巻 http://www.comicpark.net/readcomic/index.asp?content_id=COMC_AK01155_SET
- [63] PS2Linux Kit(Sync on Green)対応モニター一覧 <http://www.ps2linux.dev.jp/monitor.html>
- [64] かやまさん https://www.facebook.com/jun.ebihara.18/posts/692735874076690?comment_id=306433585&offset=0&total_comments=1
- [65] LUNAのカタログ Holonic Workstation LUNA[マニュアル・データシート類] <http://www.h2.dion.ne.jp/~dogs/collect/ds/luna.html>
- [66] ワークステーション操作入門 http://www.kumamotokokufu-h.ed.jp/kokufu/comp/ws_tx1.html
- [67] "It's never too late. When it's over, you get to tell the story" -- Garrison Keillor <http://www.cs.cmu.edu/afs/cs/project/mach/public/www/mach.html>
- [68] ピツバーク便利帳 サーバーの歴史 <http://komachi.sp.cs.cmu.edu/benricho/Komachi#.E3.82.B5.E3.83.BC.E3.83.90.E3.83.BC.E3.81.AE.E6.AD.B4.E5.8F.B2> 「1993年、オムロンのワークステーション業務撤退に伴い藤田さんと作業マシン達に突然の引き上げ命令が下りました。」
- [69] OpenBSD/luna88k 「network bootでIPアドレス取得」(2002/06/05)の頃のページ <http://t.co/VRxXgWWpTO>
- [70] 同じく「Miod Vallatさんに見つかる」(2004/03/21)の頃のページ <http://t.co/3QmzWm7reR>
- [71] OMRON Luna88Kについてのまとめ <http://t.co/rt5kUB74VG> 作者も忘れてる説
- [72] <http://www.cs.cmu.edu/afs/cs/project/mach/public/src/mkernel/src/kernel/luna88k/>
- [73](1,2) LUNA-II スペック表というサーベイ漏れ <https://t.co/KV9f6XS8bU>
- [74] ユニマガのluna88k発売の記事。 <https://twitter.com/a1kawa/status/360427576717611008>
- [75] LUNA-88K2 の製品仕様 プロセッサ以外は同じという見方もある <https://twitter.com/tsutsui/status/361463750982778880/photo/1>
- [76] もうひとつあった。2011年OSC京都 わざわざ2日目に持ってきていただいた超重要LUNAグッズ ペンケースとバンダナ <http://movapic.com/ebijun/pic/3812352> たしか、来場者の方の奥様の所有で、「持って行くのはいいけれど絶対に持って返ってくるように」と申し渡された、というお話だったような
- [77]

名は体を表す <http://ameblo.jp/hirokun39/entry-11345138649.html>

- [78] LUNA2010 Good Design Award <http://www.g-mark.org/award/describe/20641>
- [79] Omron spins four 88110s at Data General Aviiion line <http://connection.ebscohost.com/c/articles/9402180800/omron-spins-four-88110s-data-general-aviion-line>
- [80] システム診断プログラムの開発 LUNA2010用システム診断プログラムの開発について <http://jglobal.jst.go.jp/public/20090422/200902172571690192>
- [81] mikutterの薄い本製作委員会 <http://home1.tigers-net.com/brsywe/mikutter.html>
- [82] https://twitter.com/ao_kenji/status/360775880198459394/photo/1
- [83] Wikipediaの「LUNA-Σ」という呼称は果てしなく要出典という感想。 <https://twitter.com/tsutsui/status/360430992638492672>
- [84](1,2) "RTC" の stamp のオフセットをそれぞれ x4してやればいいような気がします <https://twitter.com/tsutsui/status/360418015600312320>
- [85] まずはDIP SW操作してみても変わるかどうか <https://twitter.com/tsutsui/status/360416804876722177>
- [86] マンガソフトウェア革命—プロジェクトの全貌 <http://www.amazon.co.jp/dp/4339022543>
- [87] 仁和寺 <http://randen.keifuku.co.jp/map/17.html>
- [88] <https://gist.github.com/tsutsui/6203477> OMRON LUNA-II および LUNA-88K の電源ユニットに使用されている要交換な四級塩電解コンデンサのリスト。
- [89] <http://marc.info/?l=openbsd-cvs&m=137617369920936>
- [90] [miod@openbsd.org](http://gentiane.org/~miod/software/openbsd/luna88k/) さんのOpenBSD/luna88k resource page <http://gentiane.org/~miod/software/openbsd/luna88k/>
- [91] <https://twitter.com/tsutsui/status/365121355001237505>
- [92] <http://nrx.netbsd.org/xref/src/doc/BRANCHES?r=1.330#623>
- [93] <https://twitter.com/tsutsui/status/365121528309891072>
- [94] <http://nrx.netbsd.org/xref/src/doc/BRANCHES?r=1.330#1611>
- [95] <https://twitter.com/tsutsui/status/365121928526184448>
- [96] <https://twitter.com/tsutsui/status/365122443951616001>
- [97] <https://twitter.com/tsutsui/status/365122859305140225>
- [98] <https://twitter.com/tsutsui/status/365123833402896384>
- [99] https://twitter.com/ao_kenji/status/366154076565680128/photo/1
- [100] 20年前のコンピュータで最新のOSを動かす意味とは? <http://notredameningen.kyo2.jp/e422862.html>
- [101] 88Kと88K2のグラフィックボードを交換して調査。 https://twitter.com/ao_kenji/status/366066990093303809
- [102](1,2) 専用のゲートアレイでしょうか。 https://twitter.com/ao_kenji/status/366056571609939968
- [103] PWB7102 https://twitter.com/ao_kenji/status/366009479285854208
- [104] UniOS-Mach を参考に 1bpp/4bpp/8bpp を自前で識別するようにした。 https://twitter.com/ao_kenji/status/368294458996948992
- [105] Luna88k マニュアル https://twitter.com/ao_kenji/status/395857381818519552
- [106] Luna88k FaceBook ページ <https://www.facebook.com/Luna88k>
- [107] OpenBSD/luna88k standalone bootloader by @MiodVallat works fine on my LUNA-88K2! https://twitter.com/ao_kenji/status/395551245563219969
- [108] LUNA-88KのPC-98拡張バス(Cバス)についてのメモ <https://gist.github.com/ao-kenji/7843096>
- [109] OpenBSD/luna88k 近況報告 NBUG 2013/9 http://www.slideshare.net/ao_kenji/openbsdluna88k-news-at-nbug-meeting-2013
- [110] OpenBSD/luna88k近況報告 NBUG 2013/12 http://www.slideshare.net/ao_kenji/nbug201312

- [111] Best tweet from OpenBSD/luna88k https://twitter.com/ao_kenji/status/482151248502591488
- [112] OMRONワークステーションLUNA 工作日記 <http://togetter.com/li/535307>
- [113] OMRONワークステーション LUNA-II 電源ユニット修理記 その3 <http://togetter.com/li/548989>
- [114] 「いけない！ルナ先生」実写化！6人のアイドルが先生に <http://natalie.mu/comic/news/105048>
- [115] <http://www.cinemart.co.jp/ikenai-run/>
- [116] native bootloader update for NetBSD/luna68k <http://mail-index.netbsd.org/port-luna68k/2014/01/11/msg000038.html>
- [117] (1,2) 関西オープンソース2013NetBSDブース展示の記録 <http://togetter.com/li/587422>
- [118] オープンソースカンファレンス2013島根 NetBSDブース展示の記録 <http://togetter.com/li/553529>
- [119] OSC2013名古屋 NBUG&NetBSDブース展示の記録 <http://togetter.com/li/522396>
- [120] オープンソースカンファレンス2013関西@京都 NetBSDブース展示記録 <http://togetter.com/li/542885>
- [121] 泣いて喜びそうなもの発掘 https://twitter.com/goinkyu_hacker/status/482528142930620416
- [122] OpenBSD/luna88kのご紹介 http://www.slideshare.net/ao_kenji/osc2014-nagoya
- [123] OpenBSD/luna88kのご紹介 パンフレット <http://www.nk-home.net/~aoyama/osc2014nagoya/obsd-luna88k-leaflet.pdf>
- [124] オープンソースカンファレンス2014 Nagoya&NBUG7月例会 の記録 <http://togetter.com/li/688742>
- [125] OpenBSD/luna88k 2014/4近況報告 http://www.slideshare.net/ao_kenji/nbug201404
- [126] Setup Bt458 color palette to support ANSI color text on 8bpp framebuffer. <http://mail-index.netbsd.org/source-changes/2014/07/13/msg056309.html>
- [127] 88K2電源交換 https://twitter.com/ao_kenji/status/485393846314872832
- [128] どうしてこのマウスにはボタンが3つもあるのか https://twitter.com/ao_kenji/status/485275421768814592
- [129] native bootloader update for NetBSD/luna68k <http://mail-index.netbsd.org/port-luna68k/2014/01/11/msg000038.html>
- [130] 泣いて喜びそうなバンドナ https://twitter.com/goinkyu_hacker/status/482528142930620416/photo/1
- [131] LUNA's keyboard driver changes from OpenBSD/luna88k <http://mail-index.netbsd.org/source-changes/2014/07/20/msg056548.html>
- [132] luna68k 4/8bpp framebuffers as a monochrome server <http://mail-index.netbsd.org/source-changes/2014/07/21/msg056590.html>
- [133] LUNAホッチキス https://twitter.com/goinkyu_hacker/status/497392417478156288/photo/1
- [134] (1,2) OSC2014京都NetBSDブース展示への道 <http://togetter.com/li/703494>
- [135] OSC2014 Kansai@Kyoto NetBSDブース展示の記録 <http://togetter.com/li/700617>
- [136] yaftxLaan http://www.slideshare.net/ao_kenji/nbug201411-yaft-42177561
- [137] KOFにおけるLUNA展示 <https://speakerdeck.com/tsutsui/kof-and-luna-at-netbsd-booth>
- [138] (1,2) PC-9801-86 sound board on LUNA http://www.slideshare.net/ao_kenji/nbug201412
- [139] OpenBSD/luna88k用の86音源ボードドライバを整理してcommit。 https://twitter.com/ao_kenji/status/549203137001553921
- [140] yaftxLUNA http://www.slideshare.net/ao_kenji/nbug201411-yaft-42177561
- [141] LUNA-88K2はPCカードの夢を見るか http://www.slideshare.net/ao_kenji/nbug201502
- [142] LUNA framebuffer mod for LCDs without Sync on Green support <http://mail-index.netbsd.org/port-luna68k/2014/08/13/msg000043.html>
- [143] 関西オープンソース2014 NetBSDブース展示の記録 <http://togetter.com/li/742243>
- [144] Lunaトレーナー <http://movapic.com/ebijun/pic/5232493>

- [145] LUNA-88K2はPCカードの夢を見るか <補遺> http://www.slideshare.net/ao_kenji/nbug201503
- [146] (1,2) C-bus拡張ボード on LUNA-88K2 http://www.slideshare.net/ao_kenji/osc2015-nagoya/
- [147] 伝説のHコメディ「Oh! 透明人間」x「いけない! ルナ先生」 コラボ読切で復活 <http://natalie.mu/comic/news/152961>
- [148] INSTALLATION NOTES for OpenBSD 5.7 <http://ftp.openbsd.org/pub/OpenBSD/5.7/luna88k/INSTALL.luna88k>
- [149] LUNAは、SX-9100以降の愛称なのですが、その前のSX-8700 の時代は、スーパーメイトという愛称でした。 <https://www.facebook.com/events/1062729970410808/permalink/1153405211343283/>
- [150] このマシンがスーパーメイトという商品名で <http://www.tomo.gr.jp/root/new/root82.html>
- [151] LUNA88Kに付属しているxzoomというデモの画像です。 <http://moon.hanya-n.org/comp/luna/luna88k.html>
- [152] OpenBSD/luna88k移植物語 http://www.slideshare.net/ao_kenji/a-story-of-porting-openbsdluna88k
- [153] FM音源の調べ on LUNA http://www.slideshare.net/ao_kenji/nbug201510
- [154] PSG音源の調べ <https://speakerdeck.com/tsutsui/osc2016-kyoto-psg-tunes-on-netbsd-luna68k>
- [155] LUNAのPSGというか647180実装 <https://twitter.com/tsutsui/status/759793635898515456>
- [156] 「できません」と云うな一オムロン創業者 立石一真 <https://www.amazon.co.jp/dp/4478006334/>
- [157] <http://www.ustream.tv/recorded/90107872>
- [158] <https://speakerdeck.com/tsutsui/osc2016-hiroshima-psg-tunes-on-netbsd-luna68k-again>
- [159] <https://speakerdeck.com/tsutsui/osc2016-kyoto-psg-tunes-on-netbsd-luna68k>
- [160] <https://speakerdeck.com/tsutsui/osc2014-kansai-at-kyoto-netbsd-luna68k-report>
- [161] http://www.ceres.dti.ne.jp/tsutsui/kof2013/NetBSD_bootloader.html
- [162] http://www.ceres.dti.ne.jp/tsutsui/osc2013kyoto/NetBSD-luna68k_m1term-fb_Twitter.html
- [163] <https://twitter.com/tsutsui/status/991191717050118144>
- [164] <https://speakerdeck.com/tsutsui/osc2016-kyoto-psg-tunes-on-netbsd-luna68k>
- [165] https://www.slideshare.net/ao_kenji/osc2019-nagoya
- [166] <http://www.pastel-flower.jp/~isaki/nono/>
- [167] (1,2) <https://twitter.com/isaki68k/status/1261646479816404992>
- [168] <https://twitter.com/tsutsui/status/1262429647364427783>
- [169] <https://twitter.com/tsutsui/status/1262430960718508033>
- [170] <https://twitter.com/isaki68k/status/1262375954883772418>
- [171] <https://twitter.com/isaki68k/status/1262949576362930180>
- [172] <https://ja.wikipedia.org/wiki/%E7%AB%8B%E7%9F%B3%E7%BE%A9%E9%9B%84>
- [173] <https://twitter.com/isaki68k/status/1317441952107827201>
- [174] http://wiki.netbsd.org/ports/luna68k/luna68k_info/#behindthescene
- [175] <https://twitter.com/isaki68k/status/1322807313741148160>
- [176] https://twitter.com/ao_kenji/status/1324952816884985857
- [177] https://twitter.com/ao_kenji/status/1324990436390268928
- [178] https://twitter.com/ao_kenji/status/1330473862686003202
- [179] https://twitter.com/ao_kenji/status/1330504720516063235
- [180] <https://twitter.com/isaki68k/status/1330124516333412361>
- [180] https://twitter.com/ao_kenji/status/1330019763775365120
- [181] (1,2) <https://twitter.com/isaki68k/status/1406522668321366022>

[182] Here come LUNA-88K emulators! https://www.slideshare.net/ao_kenji/osc2021-nagoya-248742459

[183] <http://mail-index.netbsd.org/source-changes/2022/06/10/msg139155.html>

[184] <https://twitter.com/tsutsuii/status/1551955355101691904>

[185] <https://twitter.com/isaki68k/status/1551020192037097473>

[186] <http://mail-index.netbsd.org/source-changes/2022/09/25/msg141239.html>

[187] <https://twitter.com/isaki68k/status/1585612735932674049>

[188] <https://twitter.com/isaki68k/status/1618215457722617856>

[189] <https://twitter.com/isaki68k/status/1624651445684563968>

[190] https://twitter.com/ao_kenji/status/1625846099394330625

[191] <http://www.pastel-flower.jp/~isaki/NetBSD/osc23hi/>

[192] https://event.ospn.jp/slides/OSC2024_Nagoya/OSC2024Nagoya_BSD.pdf

[193] <https://youtu.be/40fjWRmDh2U>

[194] <https://gist.github.com/ao-kenji/842605a8a25eb96a8d86a910d80da142>

[195] UNIX MAGAZINE 1993.11

[196] <https://retropcnews.com/archives/1561>

[197] <https://retropcnews.com/archives/1696>

[198](1,2) <http://miod.online.fr/software/openbsd/stories/m88k3.html>

24.10.1. このページ

- <https://github.com/ebijun/NetBSD/blob/master/Guide/Luna.rst> にあります。
- `/usr/pkgsrc/textproc/py-sphinx` をインストールして、`make html`とか。

25. NetBSD AGM2026: Annual General Meeting, Jun 06, 15:00 UTC

<https://mail-index.netbsd.org/netbsd-announce/2026/05/17/msg000397.html>

America/Los_Angeles	Sat Jun 06 07:00 PDT
America/Denver	Sat Jun 06 08:00 MDT
America/Winnipeg	Sat Jun 06 09:00 CDT
America/New_York	Sat Jun 06 10:00 EDT
Europe/London	Sat Jun 06 15:00 BST
Europe/Berlin	Sat Jun 06 16:00 CEST
Europe/Sofia	Sat Jun 06 17:00 EEST
Europe/Moscow	Sat Jun 06 17:00 MSK
Asia/Calcutta	Sat Jun 06 14:00 IST
Asia/Bangkok	Sat Jun 06 21:00 UTC+07
Asia/Manila	Sat Jun 06 22:00 PST
Asia/Tokyo	Sat Jun 06 23:00 JST
Australia/Adelaide	Sat Jun 06 23:30 ACST
Australia/Melbourne	Sun Jun 07 00:00 AEST
Pacific/Auckland	Sun Jun 07 02:00 NZST

* Topic for #netbsd-agm is: The NetBSD Foundation Annual General Meeting - Next Meeting Today at 14:00 UTC

```
<krelz> lusers
<spz> and rusers
<leot> Hello!
<fvc70r> good morning
<sborrill> Hi!
<jklos> Good meeting day to everyone :)
<PGoyette> a sleepy hello from America/Los_Angeles
<SnoopyOnZNCNow> Hi
<ktnb> Hi all
<schbax> hello everyone
```

```
<leot> We will start in around 5 minutes, sorry! (some last-time presentation and adjustment to the schedule!)
<Cryo> Welcome!
<Cryo> leot will be moderating today. We will take questions at the end of the presentations.
<SnoopyOnZNCNow> What is this? is there some sort of introduction to this apart from https://freedom.github.io/ ?
<Cryo> We have the executive committees presenting today on what has been happening since the last AGM
<racoon> If you somehow ended up here without seeing the announcement, read: https://mail-index.netbsd.org/netbsd-
<SnoopyOnZNCNow> racoon: oh ok thank you
<leot> OK, we are about to start... sorry for the delay!
<leot> Hello everyone!
<Cryo> I'll start off by thanking you for coming.
<Cryo> and handing it to leot!
<leot> Thanks Cryo and thanks for coming!
<leot> .
```

25.1. Introduction

```
<leot> Welcome to The NetBSD Foundation Annual General Meeting 2026!
<leot> .
<leot> In the agenda we will have reports from:
<leot> .
<leot> - board (<billc>)
<leot> - core (<kre>)
<leot> - admins (<spz>)
<leot> - finance-exec (<riastradh>)
<leot> - membership-exec (<martin>)
<leot> - releng (<martin>)
<leot> - security-team (<martin>)
<leot> - pkgsrc-pmc (<wiz>)
<leot> - pkgsrc-security (<tm>)
<leot> .
<leot> If there are any last-minute additions please /msg me!
<leot> .
<leot> The Q&A session will be at the end of all the presentations.
<leot> .
<leot> When Q&A begins please /msg me with "I have question for <team>" or
<leot> "I have question for <nick>" and I will give you voice when it is
<leot> your turn.
<leot> .
```

25.2. Board of Directors

```
<leot> Next presentation is prepared by <billc> and board@ for board!
<leot> I will present on <billc> behalf
<leot> .
<leot> Welcome to the 24th Annual General Meeting of The NetBSD Foundation.
<leot> .
<leot> 2025 progress:
<leot> - Recent stable releases include NetBSD 10.1 (Dec 2024), 9.4, and 9.3
<leot> - Development is currently focused on the imminent transition to NetBSD-11 [RC4]
<leot> .
<leot> We are preparing for:
<leot> - BSDcan in Ottawa, Canada
<leot> - The ISF Common Good Cyber Fund (CGCG) application window, which runs
<leot> from June 23 to August 4, 2026.
<leot> .
<leot> We recognize that different avenues may well be available to us regarding grants
<leot> and funding, and we are looking for volunteers to help us investigate, apply,
<leot> and deliver for the programs available. This includes, but is not limited to,
<leot> potential opportunities from the Internet Society, the Linux Foundation through
<leot> Alpha-Omega, Germany's Sovereign Tech Agency and Prototype Fund, or grants from
<leot> the European Union through NLnet.
<leot> .
<leot> The NetBSD Foundation Board of Directors presents a consolidated list
<leot> of the relevant and major actions that occurred since last AGM.
<leot> Quite a few discussions, actions, and follow-ups crossed multiple meetings.
<leot> Very few meetings resulted in not reaching quorum.
<leot> During this period, new director(s) were elected by the members and
<leot> officers were renewed or installed.
<leot> We continued with our Bronze level sponsorship support of BSDcan,
<leot> AsiaBSDcon, and EuroBSDcon to improve our representation at conferences
```

```
<leot> and developer summits.
* ktnb has quit (Ping timeout: 246 seconds)
<leot> .
<leot> We participated in the Google Summer of Code for 2025 and we attended
<leot> the Google Summer of Code Mentor Summit in Munich, Germany.
<leot> We are currently participating in GSoC this year with 5 students!
<leot> .
<leot> For 2025, these are the projects that passed:
<leot> - Asynchronous I/O Framework
<leot> - Using bubblewrap to add sandboxing to NetBSD
<leot> - Enhancing Support for NAT64 Protocol Translation in NetBSD
<leot> .
<leot> For 2026, these projects have been chosen:
<leot> - Improving and Stabilizing the racoon2 IKE Daemon in NetBSD
<leot> - Port the Enlightenment desktop environment to NetBSD
<leot> - improving RAIDframe
<leot> - Testing Compat Linux: Syscall testing
<leot> - Convert a Wi-Fi driver to the new Wi-Fi stack
<leot> .
<leot> We continued to improve our interaction and relationships with
<leot> vendors, as well as participating in industry PSIRT/CSIRT
<leot> with commercial vendors and other open-source projects.
<leot> .
<leot> We successfully completed the large-scale migration of our repository
<leot> infrastructure from CVS to a Git/Mercurial ecosystem, including the
<leot> launch of live hgweb and gitweb test environments.
<leot> .
<leot> We also advanced our security and compliance posture by initiating CNA (CVE Numbering
<leot> Authority) onboarding with MITRE and ensure readiness for the EU Cyber
<leot> Resilience Act (CRA).
<leot> .
<leot> We also implemented "Anti-Slop" protocols to protect codebase integrity
<leot> against code not written by humans.
<leot> .
<leot> The funded contracts continued for:
<leot> - improvements in release engineering
<leot> .
<leot> We are 12% through a fundraising campaign. *Please* consider
<leot> donating, as we are a US IRS 501(c)3 charitable organization.
<leot> .
<leot> EOF
```

25.3. Core

```
<leot> Next in the agenda we have... core@ presentation by <kre>! krelz, please go ahead!
<krelz> Hi everyone, before I begin, any other core members who want to add something
<krelz> to what I am about to present, msg leot and I'm sure you can be snuck in when I
<krelz> am done, which won't take long...
<krelz> .
<krelz> Report from core for 2026 NetBSD AGM
<krelz> .
<krelz> core is tasked with technical management of the NetBSD project.
<krelz> .
<krelz> The current members of core are:
<krelz> .
<krelz>         Christos Zoulas         christos@
<krelz>         Chuck Silvers           chs@
<krelz>         Robert Elz               kre@
<krelz>         Martin Husemann          martin@
<krelz>         Matthew Green            mrg@
<krelz>         Taylor R Campbell        riastradh@
<krelz>         Rin Okuyama              rin@
<krelz> .
<krelz> Actual technical management is difficult in a volunteer project,
<krelz> as developers work on whatever interests them. One aspect
<krelz> which is sometimes important is in setting disputes between
<krelz> developers. Fortunately there was only one such dispute in
<krelz> the past year, which was easily amicably settled.
<krelz> .
<krelz> Core doesn't hold regular formal meetings, issues are discussed
<krelz> when they arise, otherwise we're mostly fairly dormant.
```

```
<krelz> .
<krelz> core, as a group, can be reached at core@netbsd.org
<krelz> .
<krelz> That's it for me, for this year's core report, I will be here for questions later
<krelz> .
<leot> Thank you kre!
```

25.4. Admins

```
<spz> good localtime() all
<spz> ,
<spz> admins is the following people:
<spz> christos, dogcow, kim, mspo, phil, riastradh, riz, seb, soda, spz, tls
<spz> ,
<leot> Next in the agenda... we have the admins@ presentation from spz! Please go ahead!
<spz> Statistics:
<spz> - admins runs the following TNF systems:
<spz> @ TastyLime
<spz> + 8 hardware systems, 6 'regular' Xen guests and 3 repotest Xen guests
<spz> = 1 earmv7hf, the rest amd64
<spz> - public services, the repo(s), sundry
<spz> @ AOA
<spz> + 6 hardware systems
<spz> = all amd64
<spz> - the NetBSD build farm
<spz> @ Washington University
<spz> + 7 hardware systems
<spz> = 2 aarch64 and the rest amd64
<spz> - two pkg builders, the repo conversion and a CI system, sundry
<spz> @ Regensburg
<spz> + 2 hardware systems, one of them with 2 Xen guests
<spz> = all amd64 (+ a sparc64 serving consoles)
<spz> - the offsite backup, archive, wip.pkgsrsrc.org and a CI system
<spz> ,
<spz> - CDN services donated by Fastly
<spz> - Housing donated by TastyLime, Two Sigma, WWU, and spz
<spz> ,
<spz> NetBSD versions in use:
<spz> 6 10.0_STABLE (1 earmv7hf, 1 aarch, 4 amd64)
<spz> 6 10.1 (5 amd64)
<spz> 13 10.1_STABLE (13 amd64)
<spz> 1 11.0_RC3 (amd64)
<spz> 2 11.0_RC4 (aarch64, amd64)
<spz> ,
<spz> Changes:
<spz> Riastradh spent even more time on the mail system so we can still send
<spz> mail to Google mail accounts.
<spz> Also Riastradh has been developping the future reposerver setup.
<spz> ,
<spz> Notable issues:
<spz> - spam suppression by technical means, which makes life hard(er) for
<spz> legitimate mailing lists and hasn't stopped spammers (or phishing) yet.
<spz> - LLM scraping. Anti-social "all your resources are belong to us",
<spz> total disregard for robots.txt, and backed by lots of money
<spz> so they can buy all the shady "residential proxies" they want,
<spz> so IP blacklists aren't feasible. Their capacity to scrape
<spz> vastly outnumbers our capacity to serve, they are very aggressive,
<spz> so the chance of a human getting to use the wip.pkgsrsrc.org website
<spz> is slim. And to add insult to injury they could just download the repo
<spz> instead of diffing every possible version of every file against every
<spz> other version via the web interface:
<spz> the point is they don't want to use resources carefully, because that
<spz> would require thought and LLMs are all about not expending that.
<spz> (if you detect some foaming at the mouth here: yes. aarrggghh!)
<spz> ,
<spz> We are very sorry but we'll have to add countermeasures like for
<spz> archive.NetBSD.org, if possibly not the same, or shut the
<spz> web interface down like we did with the cvsweb access to wikisrc.
<spz> ,
<spz> The other NetBSD web sites survive thanks to having a limited
<spz> number of links (the scrapers only visit each one twice a day),
```

```
<spz> and CDN caching.
<spz> - hardware aging at TastyLime, both the TNF servers and the network
<spz> equipment. The latter is being dealt with, there will be a downtime
<spz> for sometime soon. The former suffers from requiring half a week time
<spz> on-site and roughly two weeks from off-site to get anything working
<spz> properly again, and the activation energy required to do that is a lot.
<spz> - the perennially full /pub/pkgsrc/packages on ftp.NetBSD.org.
<spz> we have a plan, it "just" needs implementing.
<spz> ,
<spz> We often get asked:
<spz> - why don't you use a Cloud provider or rent servers instead:
<spz>     + did that with the offsite backup server, the provider ceased
<spz>     operations and just shut everything off: our data? sucks to be us.
<spz>     If we own the server it might get switched off, but we could get
<spz>     it (and thus our data) back
<spz>     + but you could have backups: having 50TB in total backed up and
<spz>     not paying an arm and a leg for retrieval and making certain the
<spz>     backup provider isn't going funny is either expensive or difficult
<spz>     + in the long run renting servers is not cheaper if they actually
<spz>     are busy all the time
<spz>     + we should always consider if this (whatever this) is a good use
<spz>     of TNF funds.
<spz> - we could sponsor you a server
<spz>     Thanks, kind of you to offer. However:
<spz>     If it's just one server we couldn't do OS updates. Having IPMI
<spz>     on the open Internet for console access is not a good security
<spz>     stance. Thus we are at a server and a console server, and having
<spz>     a console server and several servers just scales better.
<spz>     Plus we'd like to have at least one member of admins in viable
<spz>     site visit distance and an expectation of duration: site moves
<spz>     aren't much less work than hardware renewals.
<spz> ,
<spz> Thanks to riz, tls and phil for their resources, time
<spz> and blood sacrifices, too. :)
<spz> ,
<spz> Back to moderator.
<leot> Thank you very much spz!
```

25.5. Finance-exec

```
<leot> Next in the agenda we have... Riastradh with the finance-exec@ presentation!
<Riastradh> Hi, folks!
<Riastradh> Finance-exec hoards the cash, keeps the books, sends
<Riastradh> thank-you notes to donors, and pays out contracts and
<Riastradh> reimbursements.
<Riastradh> .
<Riastradh> We are:
<Riastradh> - christos (Christos Zoulas)
<Riastradh> - reed (Jeremy C Reed)
<Riastradh> - riastradh (Taylor R Campbell)
<Riastradh> .
<Riastradh> The NetBSD Foundation's public 2025 financial report is at:
<Riastradh> https://www.NetBSD.org/foundation/reports/financial/2025.html
<Riastradh> We produce this from an internal ledger maintained with
<Riastradh> ledger(1) <https://www.ledger-cli.org/>.
<Riastradh> .
<Riastradh> Highlights:
<Riastradh> - We have net assets of a little over 400k USD as of today
<Riastradh>   (we received a large donation in 2026).
<Riastradh> - In 2025, we received about 80k USD -- far surpassing our
<Riastradh>   usual donation target of 50k USD!
<Riastradh> - We spent 21k USD, mainly on:
<Riastradh>   o supporting conferences and sending developers to them
<Riastradh>   o release engineering
<Riastradh> .
<Riastradh> That was a lot more income and a lot less expenses than we
<Riastradh>   usually have. But forecasting:
<Riastradh> - We expect to purchase some more hardware replacements this
<Riastradh>   year, and components like RAM have gotten much more
<Riastradh>   expensive recently.
<Riastradh> - We have more funds for funded projects now, and while core
```

```
<Riastradh> or pkgsrc-pmc directs the funds, they're really driven by
<Riastradh> the developer proposals that are available -- so if you
<Riastradh> want to work on a funded project, send a proposal!
<Riastradh> .
<Riastradh> Happy to answer any questions about what finance-exec does,
<Riastradh> or swap notes on using ledger(1)!
<Riastradh> Thanks,
<Riastradh> -Riastradh, on behalf of finance-exec
```

25.6. Membership-exec

```
<leot> Thanks a lot Riastradh!
<leot> Next presentation is from <martin> with the membership-exec@ presentation!
<__martin> thanks
<__martin> The current members of membership-exec are:
<__martin> - Christos Zoulas <christos>
<__martin> - Martin Husemann <martin>
<__martin> - Lex Wennmacher <wennmach>
<__martin> - Thomas Klausner <wiz>, and
<__martin> - Ken Hornstein <kenh> who is on sabbatical.
<__martin> -
<__martin> Membership-exec is responsible for all aspects of
<__martin> "membership", but in practice the main task is to handle
<__martin> membership applications. The number of active developers
<__martin> (as of 2026-06-06) is 138. Note that this number is a
<__martin> bit outdated, as the membership activity validation process
<__martin> required for the board election has not yet happened.
<__martin> -
<__martin> Since the last AGM on 2025-05-17 we gained only 5 new
<__martin> developers, which is (again) way too few. We need to invite
<__martin> more people, please help active users and encourage them to
<__martin> apply.
<__martin> -
<__martin> The difference between developers and active developers
<__martin> is explained in the bylaws - an active developer has
<__martin> actually committed something in the last year, or contributed
<__martin> in an active way, like admins.
<__martin> -
<__martin> We'd like to emphasize that we appreciate all your replies
<__martin> to our membership RFC e-mails, although we do not usually
<__martin> acknowledge them. Please keep on providing feedback to
<__martin> the RFC mails.
<__martin> thanks, back to moderator
<leot> Thank you Martin!
```

25.7. Release Engineering

```
<leot> Next presentation... always from Martin but this time with the releng@ hat! :) Please go ahead __martin!
<__martin> hi again
<__martin> We are:
<__martin> abs agc bouyer he jdc martin msaitoh phil reed riz
<__martin> sborrill snj
<__martin> -
<__martin> Since the last meeting, we have:
<__martin> o Branched netbsd-11.
<__martin> o Not released any formal release (only four release
<__martin> candidates for 11.0).
<__martin> o Processed hundreds of pullup requests.
<__martin> o Streamlined the process of cutting a release.
<__martin> -
<__martin> Currently we are about to release the fifth (and
<__martin> this time definitvely last) release candidate for 11.0.
<__martin> 11.0 has had bad luck with security updates of 3rd party
<__martin> components last minute and slow progress on making
<__martin> these components updatable on relelase branches
<__martin> (like libssh moving to /usr/lib/private/).
<__martin> -
<__martin> We have only two issues open for 11.0:
<__martin> (1) the missing unbound import (catch-up to current)
<__martin> (2) a new expat release that has not made it into
```

```
<__martin> -current, but fixes a few security issues
<__martin> -
<__martin> Volunteers are welcome to help with both - please
<__martin> contact me directly if you have some time to help.
<__martin> -
<__martin> I hope to cut RC5 later this weekend or early next week,
<__martin> and then the final release maybe 10 days later. If one
<__martin> of the above items does not make it in, so be it.
<__martin> -
<__martin> We have streamlined the process of actually cutting a
<__martin> release (or release candidate) and admins made it possible
<__martin> to completely stay out of this process now. Only one releng
<__martin> member and one security-office member are needed now.
<__martin> -
<__martin> A release still takes realistically slightly less than 24h
<__martin> wall clock time, the biggest time consumers are all fully
<__martin> automated: 4h build time, 6h network transfer to ftp, 1h
<__martin> generating hashes. Plus various minor manual things like
<__martin> editing the web page and posting the release announcement.
<__martin> -
<__martin> We are still processing a huge amount of pullups.
<__martin> This is only possible because developers take the time
<__martin> to test their changes on the branch and submit a
<__martin> pullup request. We have been pretty good with this,
<__martin> and pulled up lots of security and usability
<__martin> improvements, as well as bug fixes to the various
<__martin> active branches. This is good for our users, thank you
<__martin> to everyone who cared and made it possible.
<__martin> -
<__martin> The following paragraph is (unfortunately) a verbatim
<__martin> copy from last year - and still valid.
<__martin> -
<__martin> The biggest current issue is the over-aged netbsd-9 branch.
<__martin> We need to get the NetBSD 11 release out ASAP to be
<__martin> able to move NetBSD 9.x out of support.
<__martin> -
<__martin> After the 11.0 release (and probably the repository switch)
<__martin> I plan to start a discussion about rules and processes,
<__martin> trying to make the time from branching to first release way
<__martin> smaller. A slow release cycle is not that bad overall (IMO)
<__martin> but a year long delay between branching and first release
<__martin> is clearly wrong.
<__martin> -
<__martin> That is all from release engineering for this year, we are
<__martin> hoping to have a list of several releases in next
<__martin> years report and also be close to the final release of
<__martin> 12.0.
<leot> Thank you Martin!
```

25.8. Security-team

```
<leot> Next in the agenda we have... Again Martin, but with the security-team@ hat! Feel free to go ahead __martin
<__martin> This is a brief report for security-team.
<__martin> -
<__martin> We are: agc billc cherry christos chs cyber hgutch joerg js
<__martin> kre martin maya mrg riastradh rin shm spz
<__martin> -
<__martin> Since last AGM we have not published any security
<__martin> advisories. We have fixed (and pulled up) one issue that
<__martin> has an SA pending, but it has not been finalized.
<__martin> -
<__martin> There have been numerous bug fixes applied to the tree, and
<__martin> pulled up to NetBSD-9, NetBSD-10 and NetBSD-11 release
<__martin> branches. We also have updated lots of 3rd party components
<__martin> in the tree when they had new releases fixing security
<__martin> issues. Right now only the expat library needs an update in
<__martin> -current.
<__martin> -
<__martin> Most security work goes on "behind the scenes" and we
<__martin> usually concur with request of reporters for a specific
<__martin> publication date.
```

```
<__martin> -
<__martin> Where needed we also involve NetBSD developers outside the
<__martin> team when special expertise is needed. While we try to
<__martin> assess all reported issues timely, we sometimes struggle
<__martin> with doing so. Currently we have (if I did not miscount)
<__martin> two open reports that need to be addressed.
<__martin> -
<__martin> To improve our own process, becoming more reliable and more
<__martin> transparent we are currently applying to become a CNA (CVE
<__martin> number authority). This will allow us to assign and publish
<__martin> our own CVE records. The process forces us to have public
<__martin> statements of response times and processes for issues
<__martin> reported to us. We might need to introduce a ticket system
<__martin> to help with doing timely responses.
<__martin> -
<__martin> NetBSD continues to be represented in a product security
<__martin> incident response working group with other operating system
<__martin> vendors, as well as a direct contact team with other BSD
<__martin> projects. This framework allows us to work better with
<__martin> vendors requiring an embargoed and/or coordinated release
<__martin> with other operating systems. We can begin working on
<__martin> issues that affect NetBSD much faster, instead of only
<__martin> being notified after an embargo is lifted. We are expanding
<__martin> the number of vendors as time goes on, as well as
<__martin> participating in FIRST.
<__martin> -
<__martin> This is teaching us quite a bit of where we need to
<__martin> improve our process, which is currently on-going.
<__martin> -
<__martin> Thanks to everyone helping with security issues!
<leot> Thank you very much Martin!
```

25.9. pkgsrc-pmc

```
<leot> Next in the agenda we have... pkgsrc-pmc presentation, written by <wiz>!
<leot> Unfortunately <wiz> could not attend the AGM so I will present it
<leot> The pkgsrc team kept thousands of packages in pkgsrc up to date and in
<leot> good working order, and delivered four -- the 87th through 90th --
<leot> stable branches. Great work, and thank you to bsiegert@ and maya@ for
<leot> handling the branches!
<leot> .
<leot> The pkgsrc team has welcomed one new developer, kikadf, who takes good
<leot> care of chromium and wayland.
<leot> .
<leot> The current roster is:
<leot> - agc (emeritus member)
<leot> - dholland (board representative)
<leot> - schmonz
<leot> - wiz
<leot> .
<leot> Thank you for working on pkgsrc!!
<leot> -- wiz, for pkgsrc-pmc
<leot> Thanks!
```

25.10. pkgsrc-security

```
<leot> Next in the agenda... we have pkgsrc-security@ presentation, prepared by <tm>!
<leot> He's only online via a mobile, so I will present it!
<leot> The mission of the pkgsrc Security Team is to ensure that the ever-growing
<leot> ecosystem of third party software is either safe to use or at least be sure
<leot> people are aware of the known vulnerabilities.
<leot> -
<leot> Our members monitor publicly available vulnerability feeds, mainly CVE.
<leot> -
<leot> We aggregate received advisories believed to impact pkgsrc into the pkgsrc
<leot> vulnerability list. When time allows we try to notify individual package
<leot> MAINTAINERS and locate, commit patches to fix the vulnerabilities.
<leot> -
<leot> Since 2021 our ticket handling crew is currently only 2 people, unfortunately
<leot> pretty understaffed. We are looking and welcome people volunteering to join
```

```
<leot> us!  
<leot> -  
<leot> Currently handling tickets are:  
<leot> - Leonardo Taccari <leot>  
<leot> - Thomas Merkel <tm>  
<leot> -  
<leot> The other current members of the team are:  
<leot> - Thomas Klausner <wiz>  
<leot> - Tobias Nygren <tnn>  
<leot> - Tim Zingelman <tez>  
<leot> -  
<leot> The year in numbers:  
<leot> In 2024, the vulnerability list had 9482 lines added to it (8967 more than last  
<leot> year) for a total of 30231 known vulnerabilities.  
<leot> In 2025, the ticket queue received 50050 new advisories (9330 more than last  
<leot> year). Of these 50050 new advisories:  
<leot> new:          302 ( 0.6%) (not able to handle in 2025)  
<leot> stalled:       0 ( 0.0%)  
<leot> resolved: 1697 ( 3.4%) (affecting pkgsrc packages)  
<leot> rejected: 48051 (96.0%) (no impact or duplicates)  
<leot> -  
<leot> Zafer Aydogan <zafer> also joined pkgsrc-security rotation list for several  
<leot> months in 2025 and helped us. Thanks Zafer!  
<leot> -  
<leot> The current count of vulnerable packages in pkgsrc-current is 787 (138 more  
<leot> than last year), in pkgsrc-stable is 809 (144 more than last year).  
<leot> See the periodic email to packages@NetBSD.org for the list.  
<leot> But we've 3548 vulnerabilities to review!  
<leot> We can always use help locating and committing security patches, in particular  
<leot> for the many of these that are maintained by pkgsrc-users.  
<leot> -  
<leot> We encourage all developers to help us keep the vulnerability list up-to-date.  
<leot> If you become aware of a security issue or perform a security update in pkgsrc  
<leot> please edit the list. You don't need any special privilege for this.  
<leot> You'll find the list in pkgsrc CVS repository:  
<leot> pkgsrc/doc/pkg-vulnerabilities  
<leot> -  
<leot> Please join the pkgsrc Security ticket handling crew, we're pretty understaffed  
<leot> at the moment! Feel free to get in touch with us for additional details or an  
<leot> introduction.  
<leot> -  
<leot> EOF  
<leot> Thank you very much <tm>!  
<leot> We have another presentation that was not in the agenda!
```

25.11. gnats

```
<leot> There is a gnats@ presentation by <dholland>!  
<leot> Feel free to go ahead David!  
<nbdholland> (This got held up by a schmozzle yesterday. Thanks to riastradh@ for running my dodgy scripts for me.  
<nbdholland>  
<nbdholland> Here's the bug database report since the last AGM (12 months):  
<nbdholland>  
<nbdholland> GNATS statistics for 2025 (as of June 6 2026)  
<nbdholland>  
<nbdholland> New PRs this year: 880, of which 578 are still open.  
<nbdholland> Closed PRs this year: 445. Net change: +435.  
<nbdholland> Total PRs touched this year: 946.  
<nbdholland> Oldest PR touched this year: 5514.  
<nbdholland> Oldest open PR: 1677; PR ignored for the longest: 4691.  
<nbdholland>  
<nbdholland> Total number open: 7313  
<nbdholland>  
<nbdholland> (Recall that this isn't github: in NetBSD "PR" means "problem report",  
<nbdholland> not "pull request".)  
<nbdholland>  
<nbdholland> This is the weekly plot:  
<nbdholland>  
<nbdholland>  
<nbdholland> * 6900  
<nbdholland> *****  
<nbdholland> *****
```


<cagney_> leot, tks, yes; and hello all
<cagney_> I'm just wondering if, once NetBSD makes it off CVS, if the next big plan is the bug database? Any plans
<Riastradh> heh
<Riastradh> We have had so many grandiose plans for bug database migration I lost count!
<spz> yes, but it's got goats feet and then some
<spz> since we do not want to lose old info
<nbdholland> This has a long and unfortunate history
<Riastradh> So, yes, it would be nice to migrate off gnats but we don't have a plan.
<spz> otherwise: gnats should die. die die die die already. :-P
<nbdholland> and what spz said.
<Riastradh> But maybe we can start planning after we're done with CVS.
<nbdholland> We've already done a lot of planning. The problem has always been getting any real work done on it
<Riastradh> (Actually we won't quite be _done_ with CVS because there'll still be a read-only CVS front end!)
<cagney_> My only experience is that while it matters to preserve old bugs, it matters less to migrate them to a n
<cagney_> anyway, looking forward to move ment
<leot> Thanks cagney_, spz, Riastradh and nbdholland!
<leot> We have another question, from ktnb... probably for security-team@ / core@ I think!
<ktnb> Hello!
<ktnb> it seems like there are endless numbers of bugs and security bugs being around daily nowadays. I'm not sure
<leot> If anyone would like to answer and has not been voiced, feel free to /msg me!
<__martin> I'll try to answer that
<__martin> we currently receive real bug reports at still moderate rates
<__martin> we see a few "spamish" things that first ask for bug bounty programs and then never come back with real
<__martin> so right now I'd say it is still handable w/o additional measures
<nbdholland> In the long run we would also like to use formal verification tools to get ahead of the game
<krelz> I didn't mention it in the core report, as I'm not a finance person and didn't
<krelz> want to commit TNF to spend money, but core can receive proposals for projects
<krelz> which can be funded if they seem worthwhile (at moderate rates)
<krelz> If there are any proposals for how we could do active audits of the code,
<krelz> rather than just waiting for someone else to find bugs and tell us about them,
<krelz> that seems like something which might be worthy of some expenditure
<krelz> .
<ktnb> That was kind of my concern: are we not getting a lot of bugs because of lack of usage or are we just _that
<krelz> It is probably some of both of those, much of our codebase is old, and fairly
<krelz> satble, there aren't a lot of bugs (even less security issues) to find probably,
<Riastradh> Perhaps but we shouldn't get cocky...
<krelz> and most of what does exist, is relatively harmless (unlikely, and not catastrophic)
<krelz> But also, our user base isn't all that huge, compared to other systems, so
<krelz> stray bugs can take longer to be encontered.
<krelz> But that's also (in some respects) a good thing, as finding bugs in NetBSD
<krelz> isn't so profitable for hackers, that they are less likely to bother
<krelz> .
<khorben> I'd like to add and emphasize on a few things: in NetBSD we rely on third-party components
<khorben> some of these components have a security impact and subject to scrutiny (and CVEs)
<khorben> so regardless of our relevance, we are targets too and should fund efforts ourselves
<khorben> as mentioned earlier in board's summary, we are looking for volunteers to help us do that
<khorben> thanks!
<khorben> .
<krelz> Agreed. Send proposals to core@
<ktnb> Thank folks!
<leot> Thanks ktnb, __martin, nbdholland, krelz, Riastradh and khorben!
<leot> We have another question! From Ltning... probably for some pkgsrc folks! (maybe I can answer it, but if you
<Ltning> Hey all, I am a first-ish time pkgsrc patch submitted, specifically 60114
<Ltning> It's a simple patch, of which I'd like to contribute more from time to time, but it seems "stuck"
<Ltning> I guess my questions are 1) What can I do differently to get it unstuck, and 2) is there documentation on
<racoona> Ltning: since mail is a non-live medium, and irc is, my main suggestion would be to poke us on irc
<Ltning> (I realise the pkgsrc team, like all others, are understaffed and overworked, so this is not meant to be
<racoona> it's also always helpful to say which platforms you've tested on
<racoona> just because it shows confidence in the patch
<Riastradh> Ltning: One thing that would be helpful is to make sure the 'make test' target works, in addition to s
<Ltning> Yeah - I have tried a couple times, but I guess not insisntly enough. So perhaps the documentation coul
<nbdholland> Another thing is, as per the discussion above, we all dislike gnats, and one of the reasons is that i
<Ltning> Roger that - thanks. Will follow up with that.
<nbdholland> So if you file a patch in a gnats PR, and it doesn't get attention quickly, chances are you need to p
<krelz> Also remember that everything in netbsd (incl pokgsrsc)
<krelz> is done by volunteers - the best way to get someone to look
<Riastradh> .o0(pokage source)
<krelz> at a patch, is to find a developer with similar interests
<krelz> (sorry for typos...)
<krelz> and convince them to take a look - any developer will do,
<Ltning> Yea. I guess the last comment from me then is - this is useful information, and I wish I didn't have to w
<krelz> the "pkgsrc team" (I believe) are generally more interested in
<Ltning> Don't forget the impostor syndrome - I may be brave enough to poke randomly on IRC, but not everyone will

<krelz> the workings of pkgsrc itself, rather than individual packages
<krelz> Finally, for an upgrade, which your PR is, it really helps to include
<krelz> info on what has changed, why someone would want the upgrade
<krelz> .
<leot> Thanks Ltning, racoon, nbdholland Riastradh and krelz!
<leot> We have another question, probably for finance-exec@
<nbdholland> Stuff like this about prodding people about patches being forgotten does appear on the lists at times
<nbdholland> and it's ok to ask procedural questions there
<Uilebheist> Hi all, Hi NetBSD
<Uilebheist> You mentioned that you are a US IRS 501(c)3 charitable organization - which is great for US people wa
<Riastradh> We have discussed forming a potential nonprofit organization in Europe.
<Riastradh> The main question is: How much administrative overhead does this bring on us (recall we're pretty much
<Riastradh> And, is that administrative burden worth the additional fundraising it would bring in?
<spz> specifically, there are EU-wide nonprofits, but that's a lot of red tape
<khorben> I can add to that, we have tried to revive an existing NetBSD structure in Germany to help with this
<khorben> unfortunately it hasn't brought fruition as of now
<Riastradh> And the administrative burden is likely to be more than just the sum of the administrative burden of t
<khorben> and indeed there are already broader OSS structures in Europe and elsewhere
<khorben> (and what Riastradh says)
<Uilebheist> Thank you. I guess for now we might just make a slightly smaller donation and not get tax back!
<Riastradh> For example, you may be familiar with the FSF (Free Software Foundation) and FSFE (Free Software Found
<Uilebheist> Ah yes, noticed these.
<leot> Thanks Uilebheist, spz, Riastradh, nbdholland and khorben!
<leot> I think the questions queue via my /query is currently empty!
<leot> Any other questions?
<leot> (And/or if I've missed any questions, please /msg them again!)
<Cryo> Alright, thanks everyone for coming.
<leot> Cryo: wait!
<leot> We have another question! :)
* leot gives voice to wiedi
<wiedi> Hi, is there a status update on the repo migration? (Thanks to everyone working on it!)
<Riastradh> We have infrastructure in place, just requiring tying up some loose ends for deployment, and we need t
<krelz> Also, there is the test infrastructure, that not enough developers have been using
<Riastradh> The infrastructure has taken a while because we're doing it a little differently from before, so we ca
<wiedi> Thank you, looking forward to using it :)
<wiedi> does the test infra also have a pkgsrc repo? I forgot... will have a look
<Riastradh> yes, it does
<wiedi> amazing, thanks for your work and answers :)
<leot> Thanks wiedi, Riastradh and krelz!
<leot> Any other questions? :)
<Riastradh> There are currently two test deployments, not all aligned on repository data (will change that soon),
<Riastradh> Developer access is at hg.test.n.o or git.test.n.o over ssh, and anonymous access is at anonhg.test.n.
<krelz> Please, developers, use that, so you can be familiar with
<Riastradh> and there's a test repository called testsrc which is small to mess around with
<krelz> how things will work. Less issues after the real change happens.
<Riastradh> Notes on usage: <https://www.netbsd.org/developers/mercurial/> <https://www.netbsd.org/developers/git/>
<krelz> Nothing can be :bad: in the tests, you can play safely
<Cryo> Alright, again, thanks for coming. We are excited about the roadmap ahead and look forward to achieving the
<Cryo> See you next year!
<leot> Thank you!
<Riastradh> There's also still read-only CVS access via anoncv.s.test.n.o (testsrc only for now, will be everything
<khorben> thanks @all!
* Cryo turns up the lights
<Cryo> o/ have a great rest of your day

nono 1.8.0 (2026/05/30)

nono は NetBSD とかで動作する OMRON LUNA-I/LUNA-88K のエミュレータです。何故か SHARP X68030 と virt-m68k も動いたりするかも知れません。 [nono is OMRON LUNA-I/LUNA-88K emulator runs on NetBSD and etc. It can also emulate SHARP X68030 and virt-m68k.]

Index of this page:

- [1. ビルド方法 \[How to build\]](#)
- [2. 実行方法 \[How to execute\]](#)
- [3. 設定 \[Configuration\]](#)
- [4. VM について \[About VM\]](#)
- [5. 実行してみる \[Try it\]](#)
- [6. ホストネットワーク設定例 \[Example of host network setup\]](#)
- [7. 既知の問題 \[Known Issues\]](#)
- [8. 過去のバージョンからの移行方法 \[How to migrate from old versions\]](#)
- [9. 変更履歴 \[Changes\]](#)
- [10. 連絡先、ライセンス等 \[Contact, License, etc\]](#)

緑背景は新規または目立った更新のあった箇所です。 [Green Background is new or updated paragraph.]

▼ 1. ビルド方法 [How to build]

ビルドには以下が必要です。 [The followings are required for build.]

- make (BSD make, not GNU make)
- C/C++ compiler which supports -std=c++14.
(For gcc, at least 7.4 or later. For clang, at least 7.0 or later.)
- wxWidgets 3.2.x "stable" branch. (3.0.x "old stable" branch would also work.)
- gettext
- iconv
- libslirp 4.7.0 or later (4.8.0 or later is preferred).

NetBSD(pkgsrc) なら必要なパッケージは x11/wxGTK32 と net/libslirp です。 wxGTK32 の代わりに wxGTK30 でも動くはずですが。 [If you use NetBSD(pkgsrc), x11/wxGTK32 and net/libslirp are needed. wxGTK30 would also work.]

OpenBSD ports では以下のパッケージが必要だと思います。 [You may need the following packages on OpenBSD.]

```
gettext-tools, libiconv, libslirp, wxWidgets(-gtk3?)
```

ちなみに Ubuntu 24.04 ではたぶん以下のパッケージが必要だと思います。 [You may need the following packages on Ubuntu 24.04 though I'm not sure.]

```
bmake, build-essential, gettext, libasound2-dev, libbsd-dev, libkqueue-dev, libslirp-dev, libwxgtk3.2-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。 -DRELEASE がないとハードモードになるので必ず指定してください。 [Extract the nono's source archive and build as following. If you don't specify -DRELEASE, it will be hard mode.]

```
% ./configure [<options>]
% make -DRELEASE depend
% make -DRELEASE
% su
# make install
```

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。 wx-config が標準的な名前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。また configure のオプションとして以下が指定できます。 [You can specify C/C++ compiler using environment variable CC and CXX if configure cannot find standard name suitable compiler. You can specify wx-config path using environment variable WX_CONFIG if configure cannot find wx-config. Also, you can specify the following option for configure.]

- --disable-avx2 ... amd64(x86_64) で AVX2 対応コードを無効にします。デフォルトでは、コンパイラが AVX2 に対応していることを configure が検出できれば AVX2 対応コードを生成します。 [Disable AVX2 support on amd64(x86_64). By the default, it will generate AVX2 supported binary only if configure detects that compiler supports AVX2.]
- --disable-neon ... aarch64 で NEON 対応コードを無効にします。デフォルトでは、コンパイラが NEON に対応していることを configure が検出できれば NEON 対応コードを生成します。 [Disable NEON support on aarch64. By the default, it will generate NEON supported binary only if configure detects that compiler supports NEON.]

make install により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。 [make install will install two executables. nono is the GUI executable and nono-cli is the command line executable.]

▼ 2. 実行方法 [How to execute]

nono は複数機種に対応しているため設定なしでは起動できません。設定は設定ファイルかコマンドラインオプションで指定します。詳細は以下の設定の章を参照してください。 [nono supports multiple architectures so that it needs configuration. See the following Configuration section for details.]

▼ 2.1. 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。設定の **fast-mode=1** と等価です。 [Boot as the fast mode. You can change this mode on GUI menu after boot, and the option only changes its initial state. This option is equivalent to **fast-mode=1** in configuration.]

--fd0 *file*

--fd1 *file*

(X68030 Only) フロッピーイメージを指定します。 *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。設定の **fd0-image**、 **fd1-image** とは相対パスの起点の違いを除いて同一です。 [Specifies the floppy image. If *file* is relative path, it is path from the current directory. This option is equivalent to **fd0-image**, **fd1-image** in configuration except for base directory of the relative path.]

--fontsize *height*

GUI 版のみ。全サブウィンドウの起動時のフォントサイズを指定します。起動後にメニューから変更できます。設定の **monitor-fontsize** と等価です。 [GUI Only. Specifies the initial fontsize of all sub windows. You can change this value on GUI menu after boot. This option is equivalent to **monitor-fontsize** in configuration.]

--initrd *file*

(virt-m68k Only) カーネルに渡す初期 RAM ディスクのパスを指定します。 *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。設定の **exec-initrd** とは相対パスの起点の違いを除いて同一です。 [Specifies the initial ramdisk image passed to the kernel. If *file* is relative path, it is path from the current directory. This option is equivalent to **exec-initrd** in configuration except for base directory of the relative path.]

-s *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。設定の **mainview-scale** と等価です。起動後はメニューからプリセットされた倍率と **mainview-scale** で指定された倍率には変更可能です。 [GUI Only. Specifies the initial main window scale in real number. This is equivalent to **mainview-scale** in configuration. You can change this scale on GUI menu after boot.]

--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。 *name* が正しくない場合はエラー終了します。 [Use this **name=value** instead of **name=configvalue** specified in configuration file. If *name* is not correct, it will exit on error.]

-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。設定の **exec-file** とは相対パスの起点の違いを除いて同一です。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) ファイル形式は以下の通りですが、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。また、いずれも起動元デバイスが取得できないなどの問題はあるかも知れません。 [Loads and executes host's *file*. If *file* is relative path, it is path from the current directory. This option is equivalent to **exec-file** in configuration except for base directory of the relative path. If the file is **gzip'd**, it is automatically extracted. The supported file format (after extracting) is the following. Actually, it only assumes bootloaders or kernels. And note that it may not obtain some information that where did I boot from, for example.]

- **a.out (OMAGIC)** 実行ファイル (おそらくブートローダのみサポート) [a.out (OMAGIC) executable (It probably supports bootloaders only)]
- **ELF** 実行ファイル (おそらくカーネルのみサポート。 NetBSD の実行ファイルのようであれば、カーネルだと思ってシンボルテーブルも読み込みます) [ELF executable (It probably supports the kernel only. If the file is assumed to be NetBSD executable, it will also load symbol tables as the bootloader does to the kernel.)]
- **ELF object (Experimental)**

LUNA では設定の `prom-image` (後述) に基づかず内蔵の互換 ROM で起動します。 `luna-dipsw1` の `DIPSW` 設定が `dipsw-autoboot=yes` 相当の状態ならそのまま直ちにホストファイルを実行し、そうでなければ互換 ROM のプロンプトで停止します。後者の場合でも LUNA-I なら "g" コマンド、LUNA-88K なら "b" コマンドによるロードはここで指定したホストファイルをロードします。X68030 では、IPLROM 起動の後の起動デバイスに細工がしてあり、そこからホストファイルを実行します。 `virt-m68k` (と NEWS) はこの方法でしか起動できませんので、実行ファイルの指定は必須になります。 [On LUNA, regardless of `prom-image` configuration (see below), it boots the internal emulated PROM. If `luna-dipsw1` configuration meets `dipsw-autoboot=yes`, it will immediately load and execute the host file. Otherwise, it will wait in prompt. Even in this case, "g" command (in LUNA-I) or "b" command (in LUNA-88K) will load the host file that is specified by this option. On X68030, nono hacks the boot device after normal boot from IPLROM, and loads and executes the host file. On `virt-m68k` (and NEWS), this is the only way to boot.]

以下開発用。 [For developers:]

```
-b, --b hexaddr[,skipcount]
    デバッガのブレイクポイントを 16進数で指定します。
-bi inst[:mask][,skipcount]
    デバッガの命令ブレイクポイントを指定します。
-bv vector[,skipcount]
    デバッガの例外ブレイクポイントを指定します。
-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。過去との互換性のために存在していますが、-V debugger-driver=stdio と等価です。
-H
    Human68k モードです。→ human.html
-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。
```

▼ 3. 設定 [Configuration]

`nono` の設定はいずれも以下の順序で適用されます。 [nono's configurations are always applied in the following order.]

1. デフォルト値 [Default value]
2. `~/nono.cfg` があればその内容 [Contents of `~/nono.cfg` if exists]
3. VM ディレクトリ内の `nono.cfg` (または `-c` で指定したファイル) があればその内容 [Contents of `nono.cfg` in the VM directory (or the file specified by `-c` option) if exists]
4. コマンドラインオプション [Command line option]

ファイルの書式はどちらも `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。知らないキーは警告を出した上で無視します。同じキーが複数回現れた場合、上に列挙した順に後から書いたほうで上書きし、同じファイル内でも同様に後に書いたほうで上書きします。コマンドラインオプション `-v` 等はこれをさらに上書きします。コマンドラインオプションで同じキーが複数回現れた場合も後に書いたほうが上書きします。 [The syntax of both files is `key = value` format, one per line. White spaces before and after `key` and `value` are ignored. Blank lines, lines beginning with `#` are also ignored. The lines with unrecognized key are ignored with a warning. If the same key appears more than once, the latter overwrites the former in the above order. If the same key appears in a file, the latter overwrites the former in the same manner. Then, command line option `-v` etc. overwrites them. If the same key appears more than once in the command line option, do in the same manner.]

`vmtype` を除くすべての設定項目はそれぞれデフォルト値を持っています。つまり少なくとも `vmtype` だけは設定ファイルかコマンドラインオプションで指定する必要があります。 [All configuration items except `vmtype` have default value. It means, you need to specify at least only `vmtype` by configuration file or commandline option.]

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             ... LUNA-88K
    x68030              ... X68030
    news                ... NWS-1750 (Just a joke)
    virt-m68k (or virt68k) ... virt-m68k

cgrom-image = path
    (X68030 Only) X68030 の外部 CGROM イメージファイルのパスを指定します。CGROM は 768KB です。 path
```

Port-arm archive

[[Date Prev](#)][[Date Next](#)][[Thread Prev](#)][[Thread Next](#)][[Date Index](#)][[Thread Index](#)][[Old Index](#)]

2026-06-23-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)

- **To:** port-arm%netbsd.org@localhost
- **Subject:** 2026-06-23-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)
- **From:** Jun Ebihara <jun%soum.co.jp@localhost>
- **Date:** Wed, 24 Jun 2026 04:53:43 +0900

I've updated 2026-06-23-netbsd-raspi-earmv6hf.img.gz for RPI.

<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2026-06-23-earmv6hf/2026-06-23-netbsd-raspi-earmv6hf.img.gz>
<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2026-06-23-earmv6hf/MD5>

Update:

- NetBSD-11.99.6
- pkgsrc/www/dillo failed. try www/palemoon.
- fstab
 - To `resize_root,disable` log from `/etc/fstab`.
- add net/nanotodon
 - Nanotodon is a light-weight CUI/C99 mastodon client.

arm64\$ nanotodon

Hello! Welcome to nanotodon!

First, Please tell me the server where you live.

([https://\[please enter this part\]/](https://[please enter this part]/))

>social.mikutter.hachune.net

Next, I will do application authentication.

Please access to following URL, then after authorization, please input displayed authorization code.

https://social.mikutter.hachune.net/oauth/authorize?client_id=XXX...

> [input authorization code]

:

- NetBSD 11.99.6 evbarm-earmv6hf 20260622222717Z rpi.img from nyftp.
- add pkgsrc/net/sayaka : SNS called Misskey client support
 - to show misskey.io timeline with sayaka:
 - # mlterm-wscons
 - # sayaka --local --server misskey.io
- armv6 on QEMU (was Re: armv6 support for rust)
 - <http://mail-index.netbsd.org/port-arm/2024/09/01/msg008901.html>
 - <http://mail-index.netbsd.org/tech-pkg/2022/10/21/msg026888.html>
- Re: Raspberry Pi camera under NetBSD current
 - <http://mail-index.netbsd.org/current-users/2021/11/16/msg041683.html>

1. firmware copy start* and fix*

2. sync dtb

- Firmware update

commit afc8dbd74865c6d367dba5505c5a863252588ca8 (HEAD -> master, origin/master, origin/HEAD)

Author: Serge Schneider <serge%raspberrypi.com@localhost>

Date: Sun Jun 21 19:02:13 2026 +0100

- Raspberry PI zero 2 W

<http://mail-index.netbsd.org/port-arm/2022/02/14/msg007593.html>

earmv7hf works well.earmv6hf not yet.

- Raspberry Pi [0-3] have been supported in big-endian mode

<https://mail-index.netbsd.org/port-arm/2026/02/23/msg009553.html>

"because now vcaudio(4) works even on aarch64eb :-)"

- Fw: gstreamer1 updated, raspberry help needed

<http://mail-index.netbsd.org/port-arm/2021/11/17/msg007498.html>

- multimedia/gst-plugins1-egl-gil

- multimedia/gst-plugins1-egl-opengl

- multimedia/gst-plugins1-omx

PR

#55505 RaspberryPi3A+ can't find Wi-Fi module

<http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=55505>

#54941

Raspberry Pi Zero W serial console corrupted when CPU frequency changed

<http://gnats.netbsd.org/54941>

- RPI4: testing on NetBSD/aarch64

<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/README>

- pkgin support

check `/usr/pkg/etc/pkgin/repositories.conf`.

I add

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

Port-arm archive

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

2026-06-23-netbsd-raspi-aarch64.img

- **To:** port-arm%netbsd.org@localhost
- **Subject:** 2026-06-23-netbsd-raspi-aarch64.img
- **From:** Jun Ebihara <jun%soum.co.jp@localhost>
- **Date:** Tue, 23 Jun 2026 15:57:46 +0900

I've updated 2026-06-23-netbsd-raspi-aarch64.img.gz for RPI4.

<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2026-06-23-aarch64/2026-06-23-netbsd-raspi-aarch64.img.gz>
<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2026-06-23-aarch64/MD5>

Topics:

- <https://github.com/aphor/FreeBSD15-RPi5-modules>
 - Raspberry Pi 4 support on aarch64 FDT boot (without ACPI)
<https://mail-index.netbsd.org/port-arm/2026/03/19/msg009556.html>
<https://mail-index.netbsd.org/port-arm/2026/04/30/msg009598.html>
 1. fdtbus: handle /chosen compatible "simple_bus" for Raspberry Pi 4
 2. aarch64/pmap: also prepare MD aarch64_mmap_flags() in PMAP_MI case
 3. bcm283x: skip bcmgenfb specific initialization in !bcmgenfb case
 4. evbarm/conf: add bcm2838pcie and bcm2838rng for Raspberry Pi 4B
 5. aarch64: fix /dev/mem mmap cookie handling for MMIO devices
 - gcc-14.3.0
 CFLAGS+=-Wno-error=incompatible-pointer-types
 -Wno-error=declaration-after-statement
 -Wno-error=deprecated-declarations
 -Wno-error=implicit-function-declaration -Wno-error=sign-compare
 -Wno-error=undef
 and more if need.
 - dillo build failed:
 checking whether the C compiler works... no
 configure: error: in '/mnt/aarch64/pkgsrc/www/dillo/work.aarch64/dillo-3.2.0':
 configure: error: C compiler cannot create executables
 See 'config.log' for more details
 *** Error code 77
 - for RPI5:
 1. Download RPI5-D0v1
<https://github.com/NumberOneGit/rpi5-uefi/releases/tag/v0.1>
 Download RPI5_D0.zip
 2. Extract RPI5_D0.zip with msdos format MicroSD
 mount_msdos -l /dev/sd0e /mnt
 cd /mnt
 7z e RPI5_D0.zip
 3. copy this RPI4 image to USB disk
 4. Connect RPI5:
 MicroSD:(RPI5_D0)
 USB disk(RPI4 image)
 USB Keyboard/Mouse
 HDMI
 5. Connect Power Supply and boot comes are you are,as you were.
 - dmesg:
<https://dmesgd.nyccbug.org/dmesgd?do=view&id=8519>
 - [Test] Firmware update /boot/fixup4*.dat,start4*.elf
 commit afc8dbd74865c6d367dba5505c5a863252588ca8 (HEAD -> master,
 origin/master, origin/HEAD)
 Author: Serge Schneider <serge%raspberrypi.com@localhost>
 Date: Sun Jun 21 19:02:13 2026 +0100
 - SD/MMC Configuration
 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).
- In this image, a symbolic link added
- ```
/libdata/firmware/if_bwfm:
ln -s brcmfmac43455-sdio.raspberrypi,4-model-b.txt
"brcmfmac43455-sdio.Raspberry Pi 4 Model B.txt"
to avoid
bwfm0: autoconfiguration error: NVRAM file not available
```
- only for RPI4  
 (fill dtb for RPI3?)
  - build script

Port-arm archive

[[Date Prev](#)][[Date Next](#)][[Thread Prev](#)][[Thread Next](#)][[Date Index](#)][[Thread Index](#)][[Old Index](#)]

## 2026-06-22-netbsd-armv7-earmv7hf.img

- **To:** [port-arm%netbsd.org@localhost](mailto:port-arm%netbsd.org@localhost)
- **Subject:** 2026-06-22-netbsd-armv7-earmv7hf.img
- **From:** Jun Ebihara <[jun%soum.co.jp@localhost](mailto:jun%soum.co.jp@localhost)>
- **Date:** Mon, 22 Jun 2026 19:46:25 +0900

I've updated 2026-06-22-netbsd-armv7-earmv7hf.img.gz for RPI Zero2W/2/3.

<http://cdn.netbsd.org/pub/NetBSD/misc/jun/armv7/2026-06-22-earmv7hf/2026-06-22-netbsd-armv7-earmv7hf.img.gz>  
<http://cdn.netbsd.org/pub/NetBSD/misc/jun/armv7/2026-06-22-earmv7hf/MD5>

Topic:

- YM2149 PC6001 PSG Player on Raspberry Pi 3B/Zero  
<https://speakerdeck.com/tsutsui/osc2026tokyo-spring?slide=4>  
 - Raspberry Pi 4 support on 32 bit earmv7hf GENERIC kernel  
<https://mail-index.netbsd.org/port-arm/2026/03/19/msg009557.html>  
 - Raspberry Pi 3B: /dev/mem mmap GPIO works on earmv7hf but not on aarch64  
<https://mail-index.netbsd.org/port-arm/2026/02/16/msg009541.html>

Update:

- firefox  
 pkgsrc/www/firefox/work.earmv7hf/firefox-138.0.4/third\_party/aom/aom\_ports/aarch32\_cpudetect.c:75:2:  
 error: #error "Runtime CPU detection selected, but no CPU detection  
 method " "available for your platform. Rerun cmake with -  
 DCONFIG\_RUNTIME\_CPU\_DETECT=0."

- add net/nanotodon  
 Nanotodon is a light-weight CUI/C99 mastodon client.

\$ nanotodon

Hello! Welcome to nanotodon!

First, Please tell me the server where you live.

([https://\[please enter this part\]/](https://[please enter this part]/))

>social.mikutter.hachune.net

Next, I will do application authentication.

Please access to following URL, then after authorization, please input  
 displayed authorization code.

[https://social.mikutter.hachune.net/oauth/authorize?client\\_id=XXX...](https://social.mikutter.hachune.net/oauth/authorize?client_id=XXX...)

> [input authorization code]

:

- NetBSD 11.99.6 evbarm-armv7hf 20260621130417Z armv7.img from nyftp.  
 - Raspberry PI zero 2 W support  
 need internal Wi-Fi support

- Firmware:

commit afc8dbd74865c6d367dba55055c5a863252588ca8 (HEAD -> master,  
 origin/master, origin/HEAD)

Author: Serge Schneider <[serge%raspberrypi.com@localhost](mailto:serge%raspberrypi.com@localhost)>

Date: Sun Jun 21 19:02:13 2026 +0100

sysinfo:

awk-20260426 bind-9.20.24 bozohtp2-20260508 NetBSD-11.99.6 openssh-10.3  
 opensshd-10.3 openssl-3.5.7 sh-20220122-20260621130417Z sqlite3-3.53.0  
 tzdata-2026bgtz unbound-1.25.1 userland-NetBSD-11.99.6/evbarm xz-5.8.3  
 yacc-2.020260126

pkgsrc:

at-spi2-core-2.60.4 bash-5.3.15 curl-8.20.0nb2 freetype2-2.14.3  
 gdk-pixbuf2-2.44.6 giflib-6.1.3 git-base-2.54.0nb1 glib2-2.88.1  
 gnutls-3.8.13nb1 gobject-introspection-1.86.0nb3 graphite2-1.3.15  
 gst-plugins-bad-1.28.3 gst-plugins-base-1.28.2 gstreamer1-1.28.3  
 gtk2+-2.24.33nb38 gtk3+-3.24.52nb1 gtk4-4.22.4 harfbuzz-14.2.1  
 hicolor-icon-theme-0.18 icu-78.3 imlib2-1.12.5nb2 lcms2-2.19.1  
 libcups-2.4.19nb1 libgcrypt-1.12.2 libgpg-error-1.61 libjpeg-turbo-3.1.4.1  
 libpaper-2.2.8 libproxy-0.5.12nb3 libunistring-1.4.2 libxkbcommon-1.13.2  
 lintpkgsrc-2026.03.21 md4c-0.5.3 mlterm-3.9.4nb7 mozilla-rootcerts-1.1.20260412  
 nanotodon-0.5.0nb6 nettle-4.0 nhttp2-1.69.0 pango-1.57.1 perl-5.42.2nb3  
 pkgin-26.4.0 png-1.6.58nb1 python313-3.13.14 qt5-qtbase-5.15.19  
 qt5-qtxmlextras-5.15.19 qt5-qtxmlpatterns-5.15.19 qt6-qt5compat-6.11.1nb1  
 qt6-qtbase-6.11.1nb2 qt6-qtdeclarative-6.11.1nb1 qt6-qtimageformats-6.11.1nb1  
 qt6-qtlanguageserver-6.11.1nb1 qt6-qtshadertools-6.11.1nb1 qt6-qtsvg-6.11.1nb1  
 raqm-0.10.5 ruby33-3.3.11nb2 ruby33-addressable-2.9.0 ruby33-atk-4.3.6  
 ruby33-cairo-1.18.5 ruby33-cairo-gobject-4.3.6 ruby33-gdk3-4.3.6nb1  
 ruby33-gdk-pixbuf2-4.3.6 ruby33-gio2-4.3.6 ruby33-glib2-4.3.6  
 ruby33-gobject-introspection-4.3.6 ruby33-gtk3-4.3.6nb1  
 ruby33-mikutter-5.1.2nb1 ruby33-nokogiri-1.19.4 ruby33-pango-4.3.6 sayaka-3.8.7  
 sqlite3-3.53.2 uim-1.9.7pre20251027nb4 wayland-1.25.0 wayland-protocols-1.49  
 zsh-5.9.1

- packages:

<http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv7hf/2026/>

Port-arm archive

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

## pinebook status update (20260502)

- **To:** [port-arm%netbsd.org@localhost](mailto:port-arm%netbsd.org@localhost)
- **Subject:** pinebook status update (20260502)
- **From:** Jun Ebihara <[jun%soum.co.jp@localhost](mailto:jun%soum.co.jp@localhost)>
- **Date:** Sat, 2 May 2026 09:03:46 +0900

NetBSD Arm Bootable Images: NetBSD9/NetBSD10/NetBSD-current  
<https://www.armbsd.org>

### Problems:

- pinebook:  
Using audio interface, rebooting.  
PR port-arm/57111
- Pinebook Pro graphics noticeably slower since DRM update  
fixed: PR port-arm/56596  
<http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=56596>
- gimp with -python option: don't depends on python27 packages
- seamonkey 2.53.10 can't start
- gnucash build failed
- USB Audio 2.0 Support  
<http://mail-index.netbsd.org/current-users/2022/06/03/msg042467.html>  
<https://raw.githubusercontent.com/openbsd/src/master/sys/dev/usb/uaudio.c>

### Topics:

- Tow-Boot on Pinebook Pro  
<https://tow-boot.org/devices/pine64-pinebookPro.html>  
need more testing on NetBSD.
- pkgsrc update as after 2022Q1.  
firefox 98.0.2 and libreoffice 7.3.1.3  
<https://twitter.com/ebijun/status/1513773516402790403>
- graphical mixer for NetBSD audio.

### on pinebook:

Select a mixer device:  
[\*] /dev/mixer0: sun50i-a64-audio 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:

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

- 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 libreoffice 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

Port-arm archive

[[Date Prev](#)][[Date Next](#)][[Thread Prev](#)][[Thread Next](#)][[Date Index](#)][[Thread Index](#)][[Old Index](#)]

## apple m1 status update (20260502)

- **To:** [port-arm%netbsd.org@localhost](mailto:port-arm%netbsd.org@localhost)
- **Subject:** apple m1 status update (20260502)
- **From:** Jun Ebihara <[jun%soum.co.jp@localhost](mailto:jun%soum.co.jp@localhost)>
- **Date:** Sat, 2 May 2026 06:27:18 +0900

Install:

<https://wiki.netbsd.org/ports/evbarm/apple/>

Update:

```
bind-9.20.22
openssh-10.3
openssl-3.5.6
boost-libs-1.90.0nb3
clang-21.1.8
cmake-4.3.2
firefox-148.0
harfbuzz-14.2.0nb1
icu-78.3
inkscape-1.4.3nb1
llvm-21.1.8
mate-1.26.1nb20
mlterm-3.9.4nb6
nanotodon-0.5.0nb5
nss-3.123.1
perl-5.42.2
poppler-25.08.0nb8
python313-3.13.13
ruby33-3.3.11
rust-1.94.1nb1
wxGTK32-3.2.10
```

Topics:

```
- stable-diffusion.cpp works well.
 git clone --recursive https://github.com/leejet/stable-diffusion.cpp
 cd stable-diffusion.cpp
 mkdir build
 cd build
 cmake ..
 cmake --build . --config Release
```

- USB2 audio

works ok

```
uaudio0 at uhub5 port 1 configuration 1 interface 0
uaudio0: LUXMAN LXU-OT2 (0x08bb) LUXMAN LXU-OT2 (0x2704), rev
1.10/1.00, addr 17
uaudio0: audio rev 1.00
audio0 at uaudio0: playback
audio0: slinear_le:16 2ch 48000Hz, blk 11520 bytes (60ms) for playback
```

recognize ok ,but sound output awful just like boss overdrive compressor.

```
uaudio0 at uhub5 port 1 configuration 1 interface 0
uaudio0: Topping (0x152a) D10s (0x8750), rev 2.00/1.04, addr 5
uaudio0: ignored setting with format 0x80000000
uaudio0: audio rev 2.00
audio0 at uaudio0: playback
audio0: slinear_le:16 2ch 48000Hz, blk 11520 bytes (30ms) for playback
```

- pkgsrc/textproc/ruby-nokogiri

```
make package
ok with ar 2.34
failed with ar 2.39
```

- X

```
startx failed with signal 11,
till NetBSD-10.99.2-evbarm-aarch64-202302150640Z
```

X.Org X Server 1.21.1.7

```
[40.591] (EE) Backtrace:
[40.592] (EE) 0: /usr/X11R7/bin/X (xorg_backtrace+0x3c) [0x9761b8c]
[40.592] (EE) 1: /usr/X11R7/bin/X (os_move_fd+0x78) [0x975d4f8]
[40.592] (EE) 2: /usr/lib/libc.so.12 (__sigtramp_siginfo_2+0x0)
[0xf7d48807e1e8]
[40.592] (EE) 3: /usr/X11R7/bin/X (CreateColormap+0x418) [0x9644a68]
[40.592] (EE) 4: /usr/X11R7/bin/X (miCreateDefColormap+0x84) [0x96b7744]
[40.592] (EE) 5: /usr/X11R7/lib/modules/drivers/wsfb_drv.so
(+0x147c) [0xf7d4874e322c]
[40.592] (EE) 6: /usr/X11R7/bin/X (AddScreen+0xac) [0x9639c6c]
[40.592] (EE) 7: /usr/X11R7/bin/X (InitOutput+0x26c) [0x96716c]
[40.592] (EE) 8: /usr/X11R7/bin/X (dix_main+0x1a0) [0x9608fd0]
[40.592] (EE)
[40.592] (EE) Segmentation fault at address 0xf7d487aa07f0
```

```
[40.592] (EE)
Fatal server error:
[40.592] (EE) Caught signal 11 (Segmentation fault). Server aborting
```

"Updating to a more recent release of libunwind helped,"  
<https://mail-index.netbsd.org/port-arm/2024/11/21/msg009036.html>

startx but colormap  
<https://twitter.com/ebijun/status/1583622834990956545>

startx -- -depth 30  
<http://mail-index.netbsd.org/port-arm/2021/10/21/msg007458.html>  
<https://twitter.com/ebijun/status/1480042814704873473>

application checked via tigervnc.  
<https://twitter.com/ebijun/status/1615544069161357314>

```
- pkgsrc/net/tigervnc build failed depends:
x11/libXfont2:
ERROR: This package has set PKG_SKIP_REASON:
ERROR: Package set is using native X11 component
works with comment out
#.include "../x11/libXfont2/buildlink3.mk"

- pkgsrc/x11/xterm failed
ld: fontutils.o: undefined reference to symbol 'FT_Load_Char'
ld: /usr/X11R7/lib/libfreetype.so.19: error adding symbols: DSO
missing from command line
```

#### Hardware:

Apple Mac mini M1 2020 A2348  
 USB SSD 512GB BUFFALO SSD-PGM2.0U3-BC

| Filesystem | 1M-blocks | Used   | Avail   | %Cap | Mounted on |
|------------|-----------|--------|---------|------|------------|
| /dev/dk6   | 955474    | 131445 | 776255  | 15%  | /          |
| /dev/dk5   | 43        | 43     | 0       | 100% | /boot      |
| ptyfs      | 0         | 0      | 0       | 100% | /dev/pts   |
| procfs     | 0         | 0      | 0       | 100% | /proc      |
| tmpfs      | 1912      | 3      | 1909    | 1%   | /var/shm   |
| /dev/sd1a  | 1938192   | 523211 | 1318071 | 29%  | /media     |

#### ToDo:

- audio
- Wireless LAN

#### pkgsrc:

- pkgsrc packages:  
<http://cdn.netbsd.org/pub/NetBSD/misc/jun/aarch64/aarch64/2025/pkgin-support>

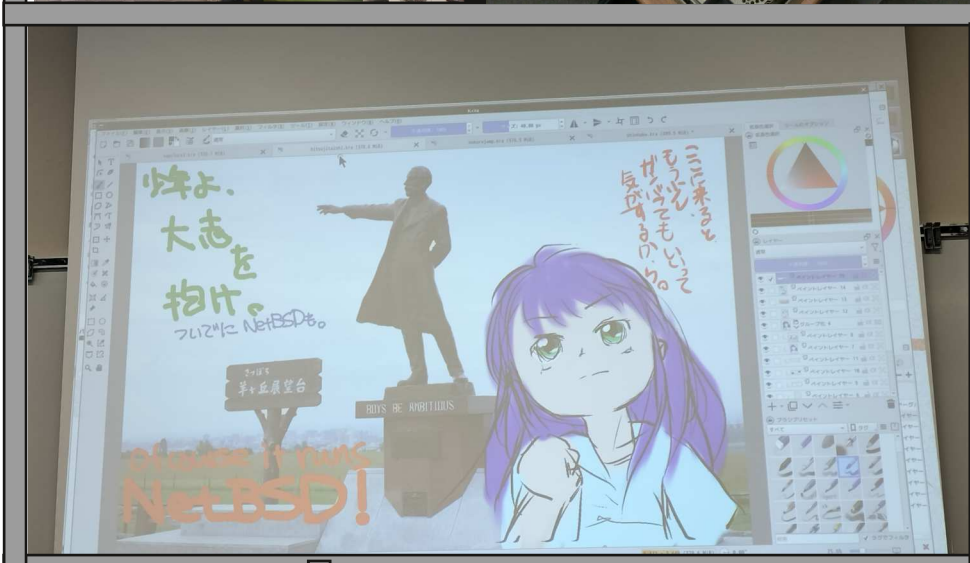
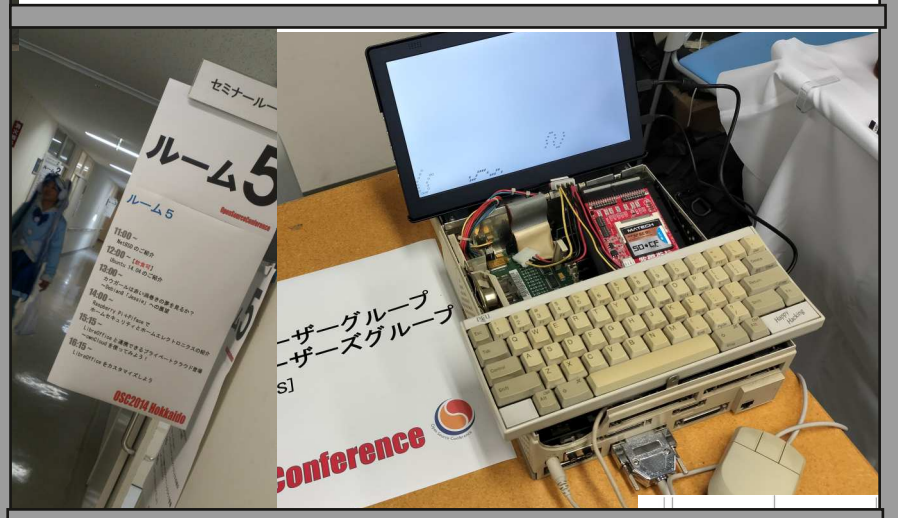
#### System Updates:

##### sysinfo:

```
bind-9.20.11 g++-14.3.0 gcc-14.3.0 libc-12.224 libssh-50.0 libstdc++-9.2
NetBSD-11.99.4 openssh-10.2 openssl-3.5.1
sh-20220122-20251201093144Z tcsh-6.24.16 unbound-1.23.1
userland-NetBSD-11.99.4/evbarm
```

##### pkgsrc:

```
GConf-3.2.5nb2 ORBit2-2.14.19nb7 OpenJPH-0.23.1 SDL-1.2.15nb47 SDL2-2.32.10nb2
SDL2_mixer-2.8.1nb1 abseil-20250814.1 ap24-php82-8.2.29nb13 apache-2.4.65nb2
appstream-1.1.1nb1 apr-1.7.6 argp-1.3nb3 at-spi2-core-2.58.1 atkmm-2.28.4nb3
atril-1.26.2nb20 automake-1.18 avahi-0.8nb8 babl-0.1.114nb2 bash-5.3.3nb1
boost-headers-1.89.0nb1 boost-jam-1.89.0 boost-libs-1.89.0nb1 cairo-1.18.4nb2
cairomm-1.14.5nb4 caja-1.26.4nb11 caja-dropbox-1.26.0nb14
caja-extensions-1.26.1nb23 cmake-4.1.2 cmake-1.1.8 consolekit-1.2.6nb1
curl-8.17.0 dbus-1.16.2nb2 dbus-glib-0.114nb1 dbus-python-common-1.3.2nb2
dconf-0.40.0nb7 desktop-file-utils-0.27nb1 dialog-1.3.20240619
djvulibre-lib-3.5.29nb1 dvipdfmx-20250205nb1 dvipsk-2025.1 emacs29-29.4nb20
enchant2-2.8.12nb1 eom-1.26.1nb19 exempi-2.5.2nb10 exiv2-0.28.7
fastfetch-2.54.0 ffmpeg4-4.4.6nb1 ffmpeg7-7.1.2nb2 firefox-145.0
firefox-l10n-145.0 fluidsynth-2.5.1 fmlib-12.1.0 fontconfig-2.17.1nb1
fossil-2.27 freetype2-2.14.1 game-music-emu-0.6.4 gawk-5.3.2 gcc12-12.5.0
gcc14-libjit-14.3.0 gcr-3.40.0nb7 gd-2.3.3nb17 gdbm-1.26 gdk-pixbuf2-2.42.12nb5
gdk-pixbuf2-xlib-2.40.2nb7 gegl-0.4.62nb5 gettext-tools-0.22.5nb1
gexiv2-0.14.3nb2 ghostscript-9.05nb45 ghostscript-agpl-10.06.0nb1 gimp-3.0.6nb1
git-base-2.52.0 glib-networking-2.80.0nb7 glib2-2.84.4nb1 glibmm-2.66.7nb1
glm-1.0.2 gnome-keyring-3.36.0nb16 gnutls-3.8.10
gobject-introspection-1.84.0nb1 gsettings-desktop-schemas-40.0nb4
gspell-1.8.4nb12 gssdp-1.6.2nb10 gst-plugins1-bad-1.26.7
gst-plugins1-base-1.26.7 gst-plugins1-good-1.26.7 gstreameer1-1.26.7
gtk2+-2.24.33nb34 gtk2-engines-2.20.2nb42 gtk2-engines-murrine-0.98.2nb25
gtk3+-3.24.51 gtkmm3-3.24.10nb1 gtksourceview3-3.24.11nb31
gtksourceview4-4.8.4nb18 gumbo-parser-0.13.2 gupnp-1.6.9nb2 gvfs-1.6.7nb76
harfbuzz-12.2.0 harfbuzz-icu-12.2.0 icewm-3.9.0 imath-3.2.2 imlib2-1.12.5
inih-62 inkscape-1.4.2nb7 jasper-4.2.8 json-glib-1.10.6nb2 latexmk-486
lcms2-2.16nb2 lerc-4.0.0 libIDL-0.8.14nb7 libabw-0.1.3nb27 libao-3.12.1
libappindicator-12.10.0nb30 libass-0.17.4nb1 libatomic_ops-7.8.4
libavif-1.3.0nb2 libcanberra-0.30nb21 libcdr-0.1.7nb20 libclucene-2.3.3.4nb28
libcms-0.6.2nb5 libcroco-0.6.13nb15 libcups-2.4.14nb1
libdbusmenu-glib-16.04.0nb3 libdbusmenu-gtk3-16.04.0nb11 libdeflate-1.25
libe-book-0.1.3nb37 libepubgen-0.1.1nb11 libetonyek-0.1.10nb21 libffi-3.5.2
libfreehand-0.1.2nb28 libfyam-0.9 libgcrypt-1.11.2 libgnome-keyring-3.12.0nb3
libgpg-error-1.56 libgtop-2.28.4nb22 libgxs-0.2.2nb35 libheif-1.20.2
```



Issue: 228  
2026/6/27

contact: jun@soum.co.jp twitter: @ebijun  
backnumber: github.com/ebijun/osc-demo/  
facebook.com/NetBSD.jp

