# NetBSD - Now, 4.0 and Future

Japan NetBSD Users Group
Internet Initiative Japan, Inc.
Masao Uebayashi
uebayasi@gmail.com

# Contents

- Enquete
- Releases
- Notable changes in 2006-2007
- Branches

# Enquete

- Purpose
  - Desktop / PDA
  - Server
  - Embedded (network) / Embedded (non-network)
- Business / Hobby

# Releases -- List

- netbsd-2
- netbsd-3
- netbsd-4
- netbsd-5 (HEAD)

# Releases -- netbsd-2

- Native pthread
- I18N (Citrus)

# Releases -- netbsd-3

- pf(4), tap(4), EtherIP, IPsec NAT-T, ipsec-tools's racoon
- Xen 2.0
- vndcompress(8)

# Releases -- netbsd-4 (1)

- agr(4), tun(4) IPv6, carp(4), hostapd(4)
- bluetooth(4), ieee1394(4), gpio(4), onewire(4), firmload(4), iscsi-target(8), ipmi(4), spi(4)
- kauth(9), fileassoc(9), paxctl(8), libssp(3)
- LFS improvement, tmpfs

# Releases -- netbsd-4 (2)

- Xen 3.0 dom0, landisk, ews4800mips
- proplib(3)
- todr(9), timecounter
- gcc 4.1
- ...

# Releases -- netbsd-5

- newlock2, better MP stability, 1:1 thread model
- puffs
- secmodel
- wcurses, ls(1) I18N
- gcc 4.1.x

# Releases -- Comparison

- netbsd-3 ... stability
- netbsd-4 ... many new features
- netbsd-5 ... better MP support

# Wanted (``mada-'') list -- Kernel framework

- dconf
- m17n/i18n wscons (uwscons?)
- MI power management framework (envsys?)
- devfs
- compat32 on amd64

# Wanted (``mada-'') list -- Filesystem

- LFS syscall?
- Journaling
- HFS
- NTFS write
- resize_ffs for UFS2

# Wanted (``mada-'') list -- Filesystem

- filesystem pathname m17n/i18n
- mount_msdos case option
- UDF

# Wanted (``mada-'') list -- Network

- pfsync

# Wanted (``mada-'') list -- Device support

- sparc wscons
- in-kernel audio mixing
- ehci, ieee1394, ...

# Wanted (``mada-'') list -- machine-dependent

- mips SMP, sparc64 SMP
- mips64
- i386 reorg?
  - Drop ``i386'' support
  - ``acpix86''? (no-ISA architecture)
  - pc98

# Wanted (``mada-'') list -- library

- library thread-safe

# Wanted (``mada-'') list -- pkgsrc, userland

- sysinst upgrade
- syspkg
- pkg_install deep-dependency
- Samba3 Japanese support
- mplus

# Wanted (``mada-'') list -- Documentation

- Japanese manual pages

# Wanted (``mada-'') list -- People

- Young people
- JNUG website

# Branches (1)

- newlock2
- vmlocking
- thorpej-atomic
- ad-audiomp
- wrstuden-fixsa

# Branches (1)

- yamt-idlelwp, yamt-km, yamt-pdpolicy, yamt-readahead, yamt-splraiseipl, yamt-uio_vmspace, yamt-vop
- yamt-lazymbuf

# Branches (3)

- kent-audio2
- itohy-usb1
- ppcoea-renovation

# tech-kern -- 2006/7-2006/9

- FFS journal (61)
- Power management and related concerns (42)
- Throttling IO Requests via Congestion Control (31)
- SE Linux vs SE NetBSD!! (36)
- Status report: sysmon_cpufreq(9) + powerctl(8) (31)
- Moving scheduler semantics to kern_synch.c (33)
- Proposal for changes to todr interface (33)

# tech-kern -- 2006/10-2006/12

- Magic symlinks: uid keyword translation (51)
- Veriexec by default (Re: CVS commit: src/sys/arch) (33)
- Re: CVS commit: src/sys/secmodel/bsd44 (57)
- Poolifying fileassoc (40)
- p_flag in struct proc: int -> uint64_t (43)
- New kpi proposal, sysdisk(9) (78)

# tech-kern -- 2007/1-2007/3

- timedwork (34)
- bce(4) and memory > 1GB problem (34)
- Interrupt, interrupt threads, continuations, and kernel lwps (64)
- Please Revert newlock2 (115)
- USB stack needs early review (Re: Someone should fix USB stack...) (65)
- Patches for EST and SMP (44)
- Atomic ops API (57)
- Belkin Bluetooth vs aue vs ubt (41)

# tech-kern -- 2007/4-2007/6

- envsys version 2 API (38)
- Kernel driver support for Dell 5/iR raid controller (30)
- FYI: ENVSYS 2 ready (56)
- proplib changes (123)
- GPT support still needed? (was: RE: Recursive partition) (61)

# newlock2 (1) - Goals

- Implement new locking primitives on most-used ports
- Minor locking improvements
- Remove spinlockmgr

# newlock2 (2) - Primitives

- mutex(9) - mutual exclusion primitives
- condvar(9) - condition variables
- mb(9) - memory barriers
- rwlock(9) - reader / writer lock primitives

# newlock2 (3) - mutex(9)

- Mutexes are used in the kernel to implement mutual exclusion among LWPs (lightweight processes) and interrupt handlers.
- API:
  - void mutex_init(kmutex_t *mtx, kmutex_type_t type, int ipl);
  - void mutex_destroy(kmutex_t *mtx);
  - void mutex_enter(kmutex_t *mtx);
  - void mutex_exit(kmutex_t *mtx);
  - void mutex_tryenter(kmutex_t *mtx);
  - int mutex_owned(kmutex_t *mtx);

# newlock2 (4) - condvar(9)

- Condition variables (CVs) are used in the kernel to synchronize access to resources that are limited (for example, memory) and to wait for pending I/O operations to complete.

# newlock2 (5) - condvar(9)

- API:
  - void cv_init(kcondvar_t *cv, const char *wmesg);
  - void cv_destroy(kcondvar_t *cv);
  - void cv_wait(kcondvar_t *cv, kmutex_t *mtx);
  - void cv_wait_sig(kcondvar_t *cv, kmutex_t *mtx);
  - int cv_timedwait(kcondvar_t *cv, kmutex_t *mtx, int ticks);
  - int cv_timedwait_sig(kcondvar_t *cv, kmutex_t *mtx, int ticks);
  - void cv_signal(kcondvar_t *cv);
  - void cv_broadcast(kcondvar_t *cv);
  - bool cv_has_waiters(kcondvar_t *cv);

# newlock2 (5) - mb(9)

- Many types of processor can execute instructions in a different order than issued by the compiler or assembler. On a uniprocessor system, out of order execution is transparent to the programmer, operating system and applications, as the processor must ensure that it is self consistent.
- On multiprocessor systems, out of order execution can present a problem where locks are not used to guarantee atomicity of access, because loads and stores issued by any given processor can appear on the system bus (and thus appear to other processors) in an unpredictable order.

# newlock2 (6) - mb(9)

- API:
  - void mb_memory(void);
  - void mb_read(void);
  - void mb_write(void);

# newlock2 (7) - rwlock(9)

- Reader / writer locks (RW locks) are used in the kernel to synchronize access to an object among LWPs (lightweight processes).
- In addition to the capabilities provided by mutexes, RW locks distinguish between read (shared) and write (exclusive) access. RW locks are intended to provide protection for kernel data or objects that are read much more frequently than updated. For objects that are updated as frequently as they are read, mutexes should be used to guarantee atomic access.

# newlock2 (8) - rwlock(9)

- API:
  - void rw_init(krwlock_t *rw);
  - void rw_destroy(krwlock_t *rw);
  - void rw_enter(krwlock_t *rw, krwlock_op_t op);
  - void rw_exit(krwlock_t *rw);
  - int rw_tryenter(krwlock_t *rw, krwlock_op_t op);
  - int rw_tryupgrade(krwlock_t *rw);
  - void rw_downgrade(krwlock_t *rw);
  - int rw_read_held(krwlock_t *rw);
  - int rw_write_held(krwlock_t *rw);
  - int rw_lock_held(krwlock_t *rw);

# newlock2 (9) - example

```
static void
workqueue_run(struct workqueue *wq)
{
    struct workqueue_queue *q = &wq->wq_queue;

    for (;;) {
        struct workqhead tmp;

        mutex_enter(&q->q_mutex);
        while (SIMPLEQ_EMPTY(&q->q_queue))
            cv_wait(&q->q_cv, &q->q_mutex);
        tmp.sqh_first = q->q_queue.sqh_first; /* XXX */
        SIMPLEQ_INIT(&q->q_queue);
        mutex_exit(&q->q_mutex);

        workqueue_runlist(wq, &tmp);
    }
}
```

# proplib (1)

- ``Property'' list, plist
- XML
- Mac OS X
- Users
  - envsys2
  - bootprops

# proplib (2) -- Example: envsys2

```
dispatch_dev_power: 1 events available
dispatch_dev_power: event type 0
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
        <key>driver-name</key>
        <string>acpi0</string>
        <key>power_type</key>
        <string>pswitch</string>
        <key>powerd-event-name</key>
        <string>pressed</string>
        <key>powerd-script-name</key>
        <string>power_button</string>
</dict>
</plist>
running script: /etc/powerd/scripts/power_button acpi0 pressed
```

# proplib (3) -- TODO

- plist editor?
- binary format?

# Flash support

- NetBSD i386 _native_ browser + linux binary
- How to use:
  - pkgsrc/multimedia/ns-flash
  - pkgsrc/multimedia/libflashsupport
  - pkgsrc/www/nspluginwrapper
- Google for "netbsd flash abs"

# TNF servers

- ftp.netbsd.org stability?
- releng infrastructure
- pkgsrc bulkbuild