Introduction

- Welcome to the magical world of Linux 2.5 (or 3.0?)
- Just my personal selection of kernel features
- … or kernel gossip?
- some of it already in late 2.4
- Sources:
  - yes, sources
  - LKML
  - *http://kt.zork.net/kernel-traffic/*
  - *http://www.linuxsymposium.org/2003/proceedings.php*
- feature frozen (at Halloween)
- 2.5-bk(9.11.2002): 13549 files, 5502454 lines
- 2.4.19: 10811 files, 4585372 lines

Behold, times have changed!

- kernel summits (2002 San Jose, 2003 Ottawa)
- Linus uses BitKeeper (but not mandatory)
- real changelogs!
- "Trusted developers" – "convince me!"
- Trivial Patch Monkey (aka Rusty Russell)
  - *http://www.kernel.org/pub/linux/kernel/people/rusty/trivial*
- chaos with merging of 2.4.x stuff
- Kernel Janitor project (*http://kerneljanitor.org/*):
- Stanford Checker (checking for locks on return paths, user space pointers, large stack vars, …)
- multiple trees (-ac, -aa, -dj …)
- documentation for functions

Building the kernel

- relatively recent toolchain required
- beware of `procps` (2 competing versions arose)
- config rewrite by Eric Raymond: *scratched*
- `kconfig` by Roman Zippel
  - moved from procedural to declarative configuration
  - single back-end, multiple front-ends
- Makefile rewrite (`kbuild 2.5`; multiple obj trees) by Keith Owens: *not included*
- rather changing the Makefiles in small steps

Processes & threading

- New scheduler (finally!)
  - old scheduler was linear, new one is "$O(1)$"
  - runs separately on each CPU, steals others' processes
  - optimized for a small number of ready tasks
  - good interactive performance under heavy load
  - fairness
  - priorities & niceness
  - realtime scheduling
  - keeping on current CPU if possible (caches!)
  - batch scheduling for nice +19 jobs (almost `SCHED_IDLE`)
  - setting affinity of processes (and also interrupts)
- $HZ = 1000$ on i386
  - low-latency work
  - … unexpected result: preemptive kernel (spinlock trick)
- NPTL (Native Posix Thread Library – nested acronym :))
  - "threads don't exist" (`clone`) … almost: PID vs. TID, some signal obscurities
  - 1:1 model, `CLONE_THREAD`
  - per-thread storage: use `gs` (via dynamic per-CPU GDT)
  - 100000 threads (somewhat silly)
  - `exit_group`
- futexes, basic ideas:
  - userspace counter modified by atomic instructions
  - kernel helper functions solving races and handling waits
  - nothing is allocated
  - kernel uses hashed wait queues according to phys. address

MM

- again rewritten :)
- reverse mapping
- huge TLB (new syscalls)
- NUMA [Non-Uniform Memory Architecture] support (beginning)
- new overcommit code (3 modes: reasonable (default), full and none)
- `/proc/sys/vm/swappiness` (0 to 100)

Architectures

- x86-64 (AMD & SuSE)
  - seemingly the most lunatic CPU architecture (still extending i386!)
  - ... but it's probably the only chance to make it marketable :(
  - long mode: removed useless instructions → prefixes → more registers
  - vsyscalls (`gettimeofday` &c.; living in special parts of address space)
- UML (with hostfs & virtual network card, ideal for debugging & profiling)
- sparc32 no longer maintained :(
- i386 has subarchs (IBM Summit, SGI VisWS, ...)

Networking

- VLANs
- NFS: v3, v4, NFS over TCP
- new network driver API (use polling if available)
- HTB scheduler by Martin Devera
- IPv6 merges (USAGI) & IPsec infrastructure
- SCTP (RFC 2960: Stream Control Transport Protocol)
  - addresses shortcomings of TCP
  - originally developed for transport of PSTN signalling
  - packet-based, packets are *partially* ordered
  - takes care of retransmits, congestion avoidance, fragmenting & coalescing
  - multihoming
- netfilter: bridging & ipv6

Drivers

- Great Unified Architecture ("been there"), rationale:
  - problems with naming of devices (must do in user space)
  - hotplugging (partially in user space)
  - power management
  - bus & device trees
- sysfs
- hotplugging PCI cards
- hotplugging CPU's
- PCMCIA still in discord, hopefully somebody will fix it
- more work on I^2C, but sensors still missing

Block devices

- IDE rewrite scratched :(
  - IDE maintainership seems to be dangerous for mental health :)
  - Andre Hedrick strikes back; Bartlomiej Zolnierkiewicz?
- IDE Tagged Command Queueing
- new I/O scheduler with deadlines
- 64-bit `sector_t` (`CONFIG_LBD` for >2TB disks)
- EDD: how to determine boot disk?
- Raw driver (not cache-coherent; data needs to be aligned)
- LDM (new Windows partitions)

Subsystems

- Input drivers (unifications of keyboards, joysticks, mice &c.; event-driven)
- Bluetooth
- ALSA by Jaroslav Kysela

Power management

- ACPI (uh!)
- swsuspend (Pavel Machek)
- cpufreq (setting of range & policy in `/proc/cpufreq`)

Security

- good news: sec by obsc constantly refused
- security model completely changes – LSM
- default model:
  - The Almighty Root is gone
  - based on capabilities
  - setuid → set all caps
  - … unless `SECURE_NONROOT` (global secure bits; nobody sets them yet :))
  - how to do setcap?
  - poor separation problems … user exec server?
- 32-bit uids (&libc5 problems)
- crypto infrastructure (IPsec, encrypted loop &c.)

VFS

- multiple mounts, binding mounts
- `CLONE_NEWNS` (get a copy of current namespace)
- new quota with byte accounting (needed for reiser)

- async IO – WIP
  - don't know what's been merged to the kernel – confusing, but promising
  - traditional UNIX I/O model and its limits
  - attempts to make select scale better: `poll`, `/dev/poll` &c.
  - ... but they don't solve problems with files → threads → overhead → async IO
- `epoll` (originally `/dev/epoll`, but Linus said: "syscall!")
- `sys_readahead`
- `O_DIRECT` (needs fs support, ext[23], JFS and NFS have it)

Filesystems

- EXT3: essentially EXT2 with journaling (why and why not)
- EXT{2,3} new features (only partially merged):
  - htrees: btree with key=hash, leaves are pages of std directory; compatible
  - extended attributes
  - ACL's
- NTFS 2.x (maybe we'll get write support some day)
- AFS (trivial read-only version)
- BEFS
- JFS
- XFS
- CIFS (post-Samba)
- Gone: ext, xiafs (probably will be resurrected)

Varia

- using C99 (at least initializers)
- Intel compiler
- oprofile

Guts

- PPP rewrite (generic part, sync, async, OE)
- uninlining (saves a good lot)
- RB trees (e.g., VMA's)
- radix trees for page array(s)
- BH's removed
- `cli()` gone
- One Lock to Rule Them All ... almost gone
- debugging switches

To be merged (maybe)

- Reiserfs 4
- ext2 online resize
- fbdev rewrite
- console rewrite
- LVM2 or EVMS (LVM1 is gone)
- new module loader with unified parameter passing
- crash dumping
- posix hi-res timers
- uclinux
- initramfs (cpio &c.)
- new mount API
- `CONFIG_TINY`
- kexec
- dynamic `NGROUPS`

Other interesting stuff

- tmpfs
- `MODULE_LICENSE`, `EXPORT_SYMBOL_GPLONLY` &c.
- `if (likely(...))`
- init sections of various types
- brlock
- hyperthreading (2.4)
- slab allocator
- `vfork` (another oldie)
- khttpd gone (no more needed)
- ADEOS nanokernel

Most controversial stuff

- BitKeeper & its licensing
- KiB, MiB, GiB
- LTT &al.