The kernel report

(ELC 2012 edition)

Jonathan Corbet
LWN.net
corbet@lwn.net
The Plan

Look at a year's worth of kernel work
...with an eye toward the future
Starting off 2011

2.6.37 released - January 4, 2011
11,446 changes, 1,276 developers

VFS scalability work (inode_lock removal)
Block I/O bandwidth controller
PPTP support
Basic pNFS support
Wakeup sources
What have we done since then?

Since 2.6.37:
- Five kernel releases have been made
- 59,000 changes have been merged
- 3069 developers have contributed to the kernel
- 416 companies have supported kernel development
February
As you can see in these posts, Ralink is sending patches for the upstream rt2x00 driver for their new chipsets, and not just dumping a huge, stand-alone tarball driver on the community, as they have done in the past. This shows a huge willingness to learn how to deal with the kernel community, and they should be strongly encouraged and praised for this major change in attitude.
– Greg Kroah-Hartman, February 9
# Employer contributions 2.6.38-3.2

<table>
<thead>
<tr>
<th>Company</th>
<th>Contributions</th>
<th>Company</th>
<th>Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volunteers</td>
<td>13.9%</td>
<td>Wolfson Micro</td>
<td>1.7%</td>
</tr>
<tr>
<td>Red Hat</td>
<td>10.9%</td>
<td>Samsung</td>
<td>1.6%</td>
</tr>
<tr>
<td>Intel</td>
<td>7.3%</td>
<td>Google</td>
<td>1.6%</td>
</tr>
<tr>
<td>unknown</td>
<td>6.9%</td>
<td>Oracle</td>
<td>1.5%</td>
</tr>
<tr>
<td>Novell</td>
<td>4.0%</td>
<td>Microsoft</td>
<td>1.4%</td>
</tr>
<tr>
<td>IBM</td>
<td>3.6%</td>
<td>AMD</td>
<td>1.3%</td>
</tr>
<tr>
<td>TI</td>
<td>3.4%</td>
<td>Freescale</td>
<td>1.3%</td>
</tr>
<tr>
<td>Broadcom</td>
<td>3.1%</td>
<td>Fujitsu</td>
<td>1.1%</td>
</tr>
<tr>
<td>consultants</td>
<td>2.2%</td>
<td>Atheros</td>
<td>1.1%</td>
</tr>
<tr>
<td>Nokia</td>
<td>1.8%</td>
<td>Wind River</td>
<td>1.0%</td>
</tr>
</tbody>
</table>
Kernel changeset contributions by employer

- Red Hat
- Intel
- Volunteers
- Novell
- IBM
- TI
- Samsung

Graph showing contributions from different employers over time.
Also in February

Red Hat stops releasing individual kernel patches
March
2.6.38 released – March 14, 2011
(9,577 changes from 1198 developers)

Per-session group scheduling
dcache scalability patch set
Transmit packet steering
Transparent huge pages
Hierarchical block I/O bandwidth controller
Somebody needs to get a grip in the ARM community. I do want to do these merges, just to see how screwed up things are, but guys, this is just ridiculous. The pure amount of crazy churn is annoying in itself, but when I then get these "independent" pull requests from four different people, and they touch the same files, that indicates that something is wrong.
– Linus Torvalds, March 17
What is the “ARM problem”?  

Wildly varying hardware
“Embedded” mindset
Little high-level oversight or communications

Results
- Lots of little subtrees
- Lots of duplicated code
- A big ugly mess in general
Why is this happening

For years we have asked embedded vendors to contribute back to the kernel.
Why is this happening

For years we have asked embedded vendors to contribute back to the kernel.

...now they are doing it!
Cleaning up the mess

More high-level oversight
  Arnd Bergmann's arm-soc tree

More cleanup work
  GPIO consolidation
  Pinmux subsystem
  Common clock framework

Move toward device tree
  Eliminate lots of “board files”
  Someday: one ARM kernel for all systems
April
Native Linux KVM Tool

A simple QEMU replacement
Aimed at kernel developers

The sticking point:
The desire to add it to the kernel tree
I think it's only a matter of time until someone takes the Linux kernel, integrates klibc and a toolchain into it with some good initial userspace and goes wild with that concept, as a single, sane, 100% self-hosting and self-sufficient OSS project, tracking the release schedule of the Linux kernel.

– Ingo Molnar, April 5
User-space code in the kernel tree?

Advantages
Wider visibility of the code

Develop ABI and users together

Encourage thinking across the boundary

Better integration
# User-space code in the kernel tree?

**Advantages**
- Wider visibility of the code
- Develop ABI and users together
- Encourage thinking across the boundary
- Better integration

**Disadvantages**
- Kernel tree bloat
- ABI stability problems
- Other projects are disadvantaged
- Where does it end?
The mobile space is about proprietary drivers – Mark Charlebios, Qualcomm Innovation Center
May
Seccomp - sandboxing for Chrome
   A simple bitmask to limit available system calls

“How not make it more powerful?”
   Various filtering schemes proposed
   Perhaps use tracepoints as enforcement points?

The end result
   Nothing merged
Yet another kernel release

2.6.39, May 18, 2011
(10,269 changesets, 1,258 developers)

Directed yield
IPset
Transcendent memory core
User namespace support
Media controller subsystem
BIG KERNEL LOCK
1996-2011
WE THOUGHT YOU WERE WITH US FOREVER.
During the 2.6.40 merge window

The voices in my head also tell me that the numbers are getting too big. I may just call the thing 2.8.0. And I almost guarantee that this PS is going to result in more discussion than the rest, but when the voices tell me to do things, I listen.
– Linus Torvalds, May 23, 2011
During the 2.6.40 merge window

The voices in my head also tell me that the numbers are getting too big. I may just call the thing 2.8.0. And I almost guarantee that this PS is going to result in more discussion than the rest, but when the voices tell me to do things, I listen.
– Linus Torvalds, May 23, 2011

If you do this, I will buy you a bottle of whatever whiskey you want that I can get my hands on in Tokyo next week.
– Greg Kroah-Hartman
June
Ext4 snapshots posted

Save copies of a running ext4 filesystem

Useful for
  System rollbacks
  Backups
  Factory reset
...

Why put all this effort into shoehorning in such a big an invasive feature to ext4 when btrfs does this all already? …

The wonderful thing about ext4 is its a nice basic fs. If we're going to start doing lots of crazy things, why not do them to the fs that isn't yet in wide use and can afford to have crazy things done to it without screwing a bunch of users who already depend on ext4's stability?
– Josef Bacik
What's up with ext4?

“Bigalloc”
Allocate blocks in units >4096 bytes
Makes operations on large files much faster
Merged for 3.2

In the works
Snapshots
Inline data for small files
Secure erase support
Metadata checksumming
...
In other words

Ext4 will continue to develop and grow for a while yet.

I'm actually finding that ext4 has found a second life as a server file system in large cloud data centers. It turns out that if you don't need the fancy-shmancy features that Copy-on-Write file systems give you, they aren't free.

– Ted Ts'o
UEFI secure boot

The objective:
Only give control of the system to a “trusted” boot loader

This concept has value
Thwart bootloader rootkits
Ensure the system is running what you think it is

There is only one problem:
Who is “trusted”?  

The owner of the computer?  
The hardware vendor?  
The software vendor?  
The entertainment industry?
UEFI secure boot could easily be a mechanism by which we lose control of our computers.
Where things stand

Lots of work to call attention to the problem

Some concessions gained
  All [x86] systems can be put into “setup mode”
  It will be possible to install a signing key
Where things stand

Lots of work to call attention to the problem

Some concessions gained
   All [x86] systems can be put into “setup mode”
   It will be possible to install a signing key

But:
   Installing that key may not be easy
   No provision for booting from CD
   ARM systems can be totally locked down
July
3.0-rc7-rt

The first new realtime patch set since March
Users had been stuck on 2.6.33
The state of realtime

Nice determinism on good hardware

May have a solution on per-CPU data
  ...but involves scary locking assumptions

Plan is to merge most of it in the next year
  ...time will tell...
Open issues in realtime

Deadline scheduling
CPU isolation
The 3.0 release is delayed

Nasty bug in the dcache scalability patches

The debugging crew:
   Linus Torvalds
   Al Viro
   Hugh Dickins

...it still took them several days to figure it out
Some parts of the kernel have reached a truly scary level of complexity.
3.0 kernel released, July 21
(9,153 changes from 1,131 developers)

New POSIX clocks
BPF JIT compiler
sendmmsmsg() system call
ICMP sockets (unprivileged ping)
Namespace file descriptors
Cleancache
好 Pizza 不吃嗎？

比薩斜塔，就是一盒又一盒的 pizza 斜斜的疊起來，pizza 山已經在 BoF 等著你，快來挑戰你吃pizza 的極限吧！

August
x32

64-bit mode is great, but:
  64-bit data is rarely needed
  Programs get larger, slower

The best of both worlds: the x32 ABI
  Run in full 64-bit mode
  Use 32-bit data and pointers

Mostly a user-space problem
  But kernel support is needed
20 years of Linux
Kernel.org compromised

What is known:
   Attackers had been on the system for some time
   Stolen credentials used; trojaned SSH installed
   Numerous associated machines compromised
   No attempts to corrupt software distribution

The immediate result:
   kernel.org is down for almost two months
   The 3.1 kernel release is delayed
What has been done

A new kernel.org infrastructure has been built
  Lots of machines to separate functions
  New staff hired

Access has been restricted considerably
  “Maybe 450 shell accounts is a bad idea...”

A new kernel web of trust has been built

Vast support from the Linux Foundation
We do not take the security problem seriously enough.
September
Oracle to use Btrfs by default

...sometime really soon now
Btrfs

Some new development work happening
  Lots of internal work
  Scrub feature

Stability is the biggest concern
Still missing

Btrfsck
  a hard problem, seemingly

Still under development.

Meanwhile
  Root block history array
  Read-only data recovery tool

Also missing: RAID 5/6 support
  Patches exist
October
2011 Kernel Summit
Two pivotal summit outcomes

1) Maintainers should say “no” more often

2) Widely-used code should be merged even if it is not up to normal technical standards
A slow moment at the Summit

The 3.1 kernel
  October 24, 2011 (8,693 changesets, 1,168 developers)

A 95 day cycle (average is 76)

Dynamic writeback throttling
OpenRISC architecture
PTRACE_SEIZE
lseek() hole finding

...
Embedded long-term support initiative

Two-year stable kernel maintenance
  One kernel/year
  Starting with 3.0

A separate tree for products
  Backports and such

A staging tree for upstreaming
November
Per-group TCP buffer limits

Limit kernel memory used by TCP buffers
   Accepted for 3.3

The first overt limit on kernel memory use
   Wanted for containers and such
   Lots more to come
Control groups

A simple mechanism for grouping processes
...that everybody hates

The real problem is the controllers
  Memory usage
    (Now kernel memory usage too)
  Block I/O bandwidth
  Scheduling
  CPU affinity
  ...

Expect a lot of cleanup work in this area
LTTng pulled into staging

A comprehensive tracing toolkit
Widely used in some areas

Intended for merging into 3.3
Two pivotal summit outcomes

1) Maintainers should say “no” more often

2) Widely-used code should be merged even if it is not up to normal technical standards
The outcome

LTTng loses
December
The Android mainlining project

An effort to get the Android kernel code merged

Includes

- **Binder** - interprocess communication
- **Logger** - user-space logging system
- **Low-memory killer**
- **Pmem** - contiguous memory allocation
- **RAM console**
- **Timed GPIO**
- **Ashmem** - shared object storage
The Android mainlining project

An effort to get the Android kernel code merged

Includes

**Binder** - interprocess communication
**Logger** - user-space logging system
**Low-memory killer**
**Pmem**—contiguous memory allocation
**RAM console**
**Timed GPIO**
**Ashmem** - shared object storage
January
Happy New Year

3.2 released, January 4, 2012
(11,828 changesets from 1,309 developers)

Proportional rate reduction
Extended verification module
CPU scheduler bandwidth controller
Cross-memory attach
Hexagon architecture
Btrfs recovery
I/O-less dirty throttling
3.3 merge window

“team” network device
Network priority control group
TCP buffer size controller
Byte queue limits
Open vSwitch

ARM LPAE support
The Android drivers return
DMA buffer sharing API

Expect 3.3 sometime in March
February
Greg KH joins the Linux Foundation
Stuff not covered

- Writeback
- Transcendent memory
- Barriers
- Preemption disable
- Perf and ftrace
- Pin controller
- RAID x 4
- Opportunistic suspend
- Power domains
- Common clocks
- Signed tags

- Compaction stalls
- GPL violations
- GPL termination
- Patch review
- Testing tools
- SCSI targets
- Bufferbloat
- Power-aware sched.
- Solid-state storage
- IIO
- ...

...
Stuff not covered

- Writeback
- Transcendental memory
- Barriers
- Preemption disable
- Perf and ftrace
- Pin controller
- RAID x 4
- Opportunistic suspend
- Power domains
- Common clocks
- Signed tags

Questions?

- Compaction stalls
- GPL violations
- GPL termination
- Patch review
- Testing tools
- SCSI targets
- Bufferbloat
- Power-aware sched.
- Solid-state storage
- IIO
- ...