Linux Kernel Development

Greg Kroah-Hartman gregkh@linuxfoudation.org

github.com/gregkh/kernel-development



39,700 files 15,870,000 lines

2,497 developers 406 companies

10,700 lines added 7,600 lines removed 2,340 lines modified

10,700 lines added 7,600 lines removed 2,340 lines modified

every day

6.64 changes per hour

How we stay sane

Time based releases Incremental changes







"Longterm kernels"

One picked per year Maintained for two years

3.0 and 3.4

developer developer developer developer



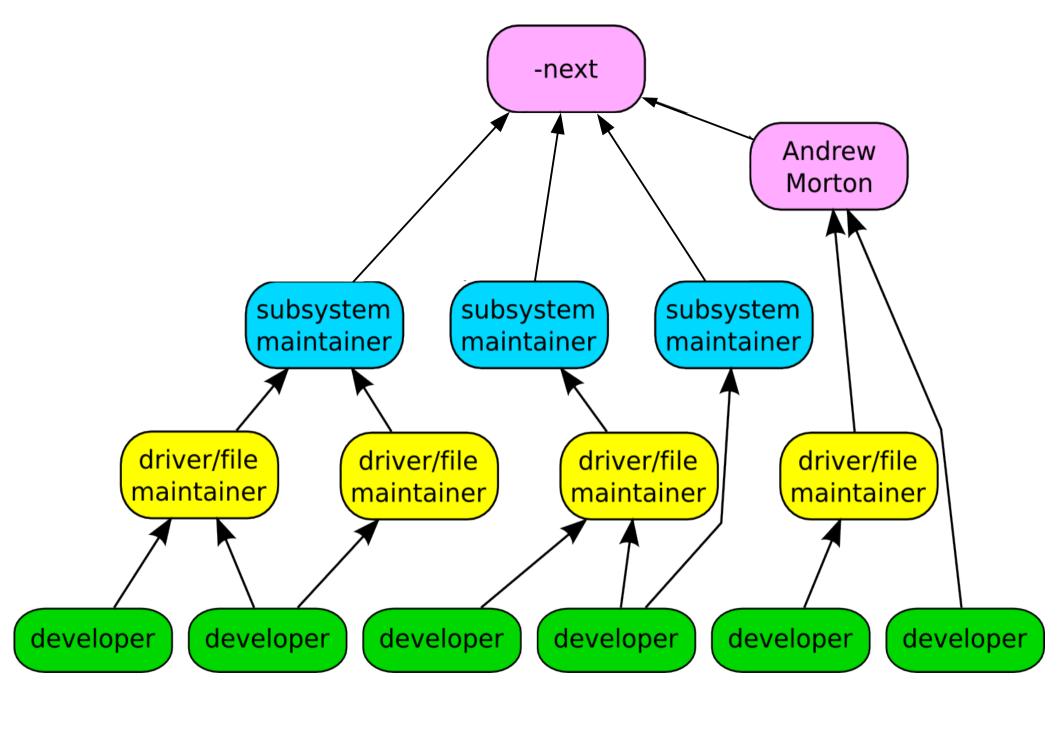
```
commit ecf85e481a716cfe07406439fdc7ba9526bbfaeb
Author: Robert Jarzmik <robert.jarzmik@free.fr>
AuthorDate: Tue Apr 21 20:33:10 2009 -0700
Commit: Greg Kroah-Hartman <gregkh@suse.de>
CommitDate: Thu Apr 23 14:15:31 2009 -0700
    USB: otg: Fix bug on remove path without transceiver
    In the case where a gadget driver is removed while no
    transceiver was found at probe time, a bug in
    otg_put_transceiver() will trigger.
    Signed-off-by: Robert Jarzmik <robert.jarzmik@free.fr>
    Acked-by: David Brownell <dbrownell@users.sourceforge.net>
    Signed-off-by: Greg Kroah-Hartman <gregkh@suse.de>
--- a/drivers/usb/otg/otg.c
+++ b/drivers/usb/otg/otg.c
@@ -43,7 +43,8 @@ EXPORT_SYMBOL(otg_get_transceiver);
void otg_put_transceiver(struct otg_transceiver *x)
        put_device(x->dev);
        if (x)
                put_device(x->dev);
```

Developer's Certificate of Origin

- (a) I created this change; or
- (b) Based this on a previous work with a compatible license; or
- (c) Provided to me by (a), (b), or (c) and not modified

(d) This contribution is public.







Top developers by quantity
Mark Brown
932 **Axel Lin** H. Hartley Sweeten AlViro 589 Johannes Berg 460 Mauro Chehab Greg Kroah-Hartman 429 Russell King 398 David Miller 390 **Eric Dumazet**

Top Signed-off-by: Greg Kroah-Hartman 4179 David S. Miller John Linville 2227 Mauro Carvalho Chehab 2041 Mark Brown 2155 Linus Torvalds 1593 **Andrew Morton** 1360 AlViro 796 David Airlie Axel Lin

Who is funding this work?

1. "Amateurs" 13.8%

2. Red Hat 10.6%

3. Intel 9.4%

4. Unknown Individuals 4.0%

5. Texas Instruments 3.7%

6. Novell 3.6%

7. Linaro 3.4%

8. IBM 2.8%

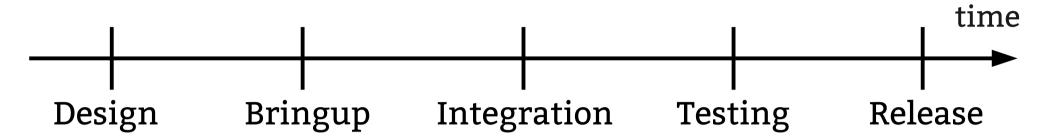
9. Google 2.3%

10. Consultants 2.2%

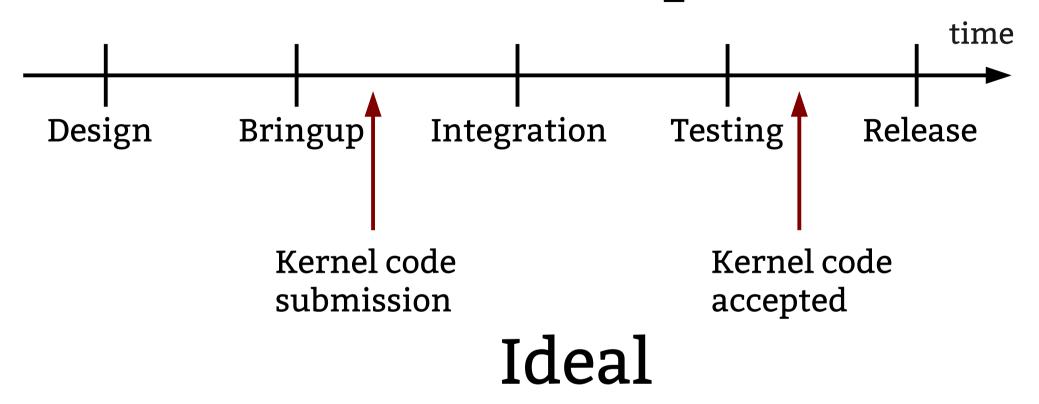
Who is funding this work?

11. Wolfson Micro	2.2%
12. Samsung	1.9%
13. Ingics Technology	1.7%
14. Oracle	1.7%
15. Qualcomm	1.6%
16. Vision Engraving	1.4%
17. NVidia	1.3%
18. Broadcom	1.3%
19. Linux Foundation	1.2%
20. Renesas Electronics	1.2%

Product Development



Product Development



"Working upstream saves time and money"

Dan Frye – VP Open Systems, IBM Dirk Hohndel – Chief Technologist, Intel



github.com/gregkh/kernel-development

Linux Kernel Development

Greg Kroah-Hartman gregkh@linuxfoudation.org

github.com/gregkh/kernel-development



I'm going to discuss the how fast the kernel is moving, how we do it all, and how you can get involved.

39,700 files 15,870,000 lines

Kernel release 3.6.0

This was for the 3.6 kernel release, which happened September 30, 2012.

2,497 developers 406 companies

Kernel releases 3.2.0 – 3.6.0 October 2011 – September 2012

This makes the Linux kernel the largest contributed body of software out there that we know of.

This is just the number of companies that we know about, there are more that we do not, and as the responses to our inquiries come in, this number will go up.

First one year timespan that we have surpassed 400 companies.

10,700 lines added 7,600 lines removed 2,340 lines modified

Kernel releases 3.2.0 – 3.6.0 October 2011 – September 2012

10,700 lines added 7,600 lines removed 2,340 lines modified

every day

Kernel releases 3.2.0 – 3.6.0 October 2011 – September 2012

6.64 changes per hour

Kernel releases 3.2.0 – 3.6.0 October 2011 – September 2012

This is 24 hours a day, 7 days a week, for a full year.

We went this fast the year before this as well, this is an amazing rate of change.

Interesting note, all of these changes are all through the whole kernel.

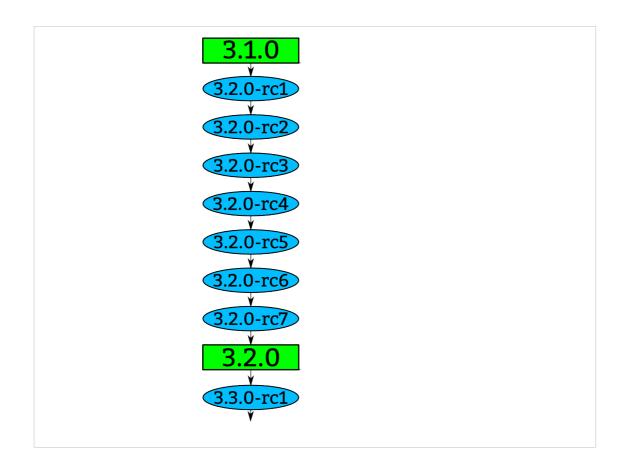
For example, the core kernel is only 5% of the code, and 5% of the change was to the core kernel. Drivers are 55%, and 55% was done to them, it's completely proportional all across the whole kernel.

How we stay sane

Time based releases Incremental changes



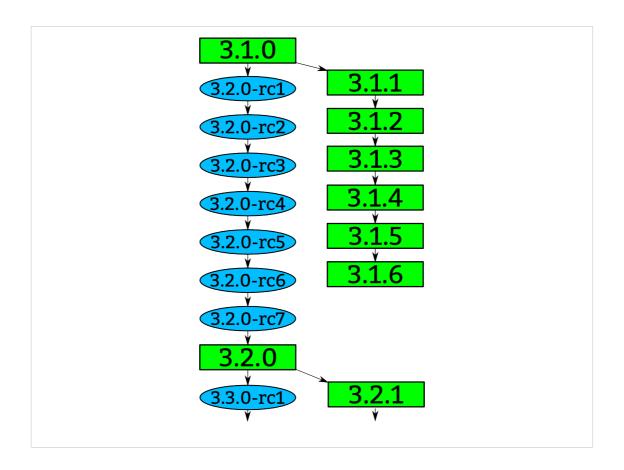
84 days to be exact, very regular experience.



How a kernel is developed. Linus releases a stable kernel

- 2 week merge window from subsystem maintainers
- rc1 is released
- bugfixes only now
- 2 weeks later, rc2
- bugfixes and regressions
- 2 weeks later,rc3

And so on until all major bugfixes and regressions are resolved and then the cycle starts over again.



Greg takes the stable releases from Linus, and does stable releases with them, applying only fixes that are already in Linus's tree.

Requiring fixes to be in Linus's tree first ensures that there is no divergence in the development model.

After Linus releases a new stable release, the old stable series is dropped.

With the exception of "longterm" stable releases, those are special, the stick around for much longer...

"Longterm kernels"

One picked per year Maintained for two years

3.0 and 3.4

I pick one kernel release per year to maintain for longer than one release cycle. This kernel I will maintain for at least 2 years.

This means there are 2 longterm kernels being maintained at the same time.

3.0 and 3.4 are the longterm kernel releases I am maintaining.

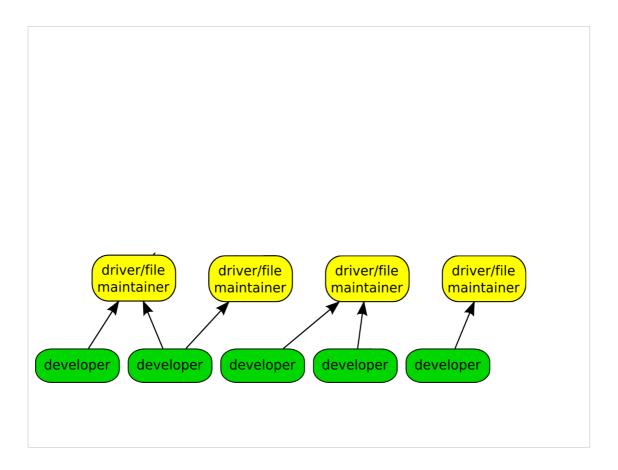
Ben Hutchings is maintaining the 3.2 kernel as a longterm kernel for the Debian project.

The LTSI project is based on the longterm kernels.



Like mentioned before, we have almost 2900 individual contributors. They all create a patch, a single change to the Linux kernel. This change could be something small, like a spelling correction, or something larger, like a whole new driver.

Every patch that is created only does one thing, and it can not break the build, complex changes to the kernel get broken up into smaller pieces.



The developers send their patch to the maintainer of the file(s) that they have modified.

We have about 700 different driver/file/subsystem maintainers

```
commit ecf85e481a716cfe07406439fdc7ba9526bbfaeb
           Robert Jarzmik <robert.jarzmik@free.fr>
AuthorDate: Tue Apr 21 20:33:10 2009 -0700
        Greg Kroah-Hartman <gregkh@suse.de>
CommitDate: Thu Apr 23 14:15:31 2009 -0700
   USB: otg: Fix bug on remove path without transceiver
    In the case where a gadget driver is removed while no
   transceiver was found at probe time, a bug in
   otg_put_transceiver() will trigger.
   Signed-off-by: Robert Jarzmik <robert.jarzmik@free.fr>
   Acked-by: David Brownell <dbrownell@users.sourceforge.net>
   Signed-off-by: Greg Kroah-Hartman <gregkh@suse.de>
--- a/drivers/usb/otg/otg.c
+++ b/drivers/usb/otg/otg.c
@@ -43,7 +43,8 @@ EXPORT_SYMBOL(otg_get_transceiver);
void otg_put_transceiver(struct otg_transceiver *x)
       put_device(x->dev);
+
       if (x)
               put_device(x->dev);
}
```

This is an example of a patch.

It came from Robert, was acked by David, the maintainer at the time of the usb on-the-go subsystem, and then signed off by by me before it was committed to the kernel tree.

The change did one thing, it checked the value of the pointer before it was dereferenced, fixing a bug that would have crashed the kernel if it had been hit.

This is also a "blame" trail, showing who changed each line in the kernel, and who agreed with that change.

If a problem is found, these are the developers that you can ask about it.

Because of this, every line in the Linux kernel can be traced back to at least two developers who are responsible for it.

This is better than any other body of code.

Developer's Certificate of Origin

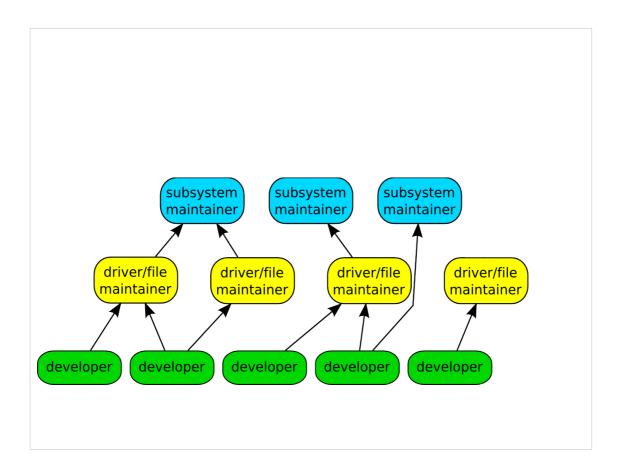
- (a) I created this change; or
- (b) Based this on a previous work with a compatible license; or
- (c) Provided to me by (a), (b), or (c) and not modified
- (d) This contribution is public.

This is what "Signed-off-by:" means. All contributions to the Linux kernel have to agree to this, and every single patch has at least one signed-off-by line, usually all have at least two.

This is also a "blame" trail, showing who changed each line in the kernel, and who agreed with that change.

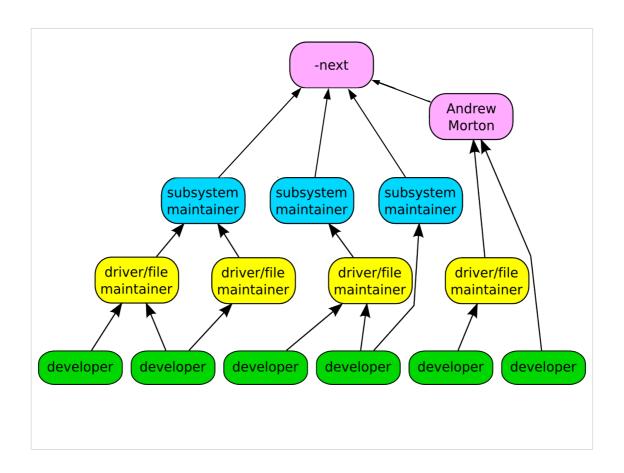
If a problem is found, this is the developers that you can ask about it.

Because of this, every line in the Linux kernel can be traced back to at least two developers who are responsible for it. This is better than any other body of code.



After reviewing the code, and adding their own signed-off-by to the patch, the file/driver maintainer sends the patch to the subsystem maintainer responsible for that portion of the kernel.

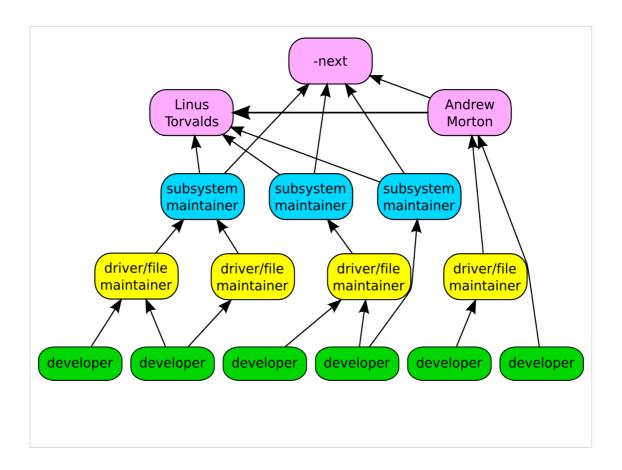
We have around 150 subsystem maintainers



Linux-next gets created every night from all of the different subsystem trees and build tested on a wide range of different platforms.

We have about 150 different trees in the linux-next release.

Andrew Morton picks up patches that cross subsystems, or are missed by others, and releases his -mm kernels every few weeks. This includes the linux-next release at that time.



Every 3 months, when the merge window opens up, everything gets sent to Linus from the subsystem maintainers and Andrew Morton.

The merge window is 2 weeks long, and thousands of patches get merged in that short time.

All of the patches merged to Linus should have been in the linux-next release, but that isn't always the case for various reasons.

Linux-next can not just be sent to Linus as there are things in there that sometimes are not good enough to be merged just yet, it is up to the individual subsystem maintainer to decide what to merge.

Top developers by community and the second s	Luantity 932 on Intel 55%
9 Ns May 30 Martin Medigeto (2.4K) Medical Mainline plans 10 N May 30 Mathias Kachle (2.4K) [PATCH] drivers/block/ub.c: use li 11 N May 30 Mataxehllin [PATCH] merge dst_discard in & out	ist_for each entry() t int 722 moved a duplicat
13 N May 30 Zoltan Boszorme (8.6%) MCP55 NCD problem? 14 N May 30 Se H. n Hartley Sweeten imeout in the second s	in ql=611wn_timeout
17 N May 30 Bil Nottingham (5.1K) [PATCH] drivers/video: Fix comparis 18 N May 30 Bil Nottingham (5.8K) [PATCH] drivers/net: fix comparisons of un	isons 589d < 0
20 N May 30 Bil Nottingham (1.9K) [PATCH] mm: fix comparisons agains 21 N May 30 Bil ohannes Bergfix comparison of ur May 30 Talonannes	st unsign 460
23 N May 30 Jiri Slaby (1.4K) 24 N May 30 St Mauro Chehab lieee1394: eth1394 25 N May 30 01 Mauro Chehab paround is not treather.	4: bri 432 a parent device ated if 132
27 N May 29 Ro Greg Kroah-Hartman	te aga 429 motherboard re
30 N May 29 Ron t Hancock 110KF 17CH -mm] 0/2: PCI MMCONFIG-rela 31 N May 29 YIRUSSEII4KKINg aacraid broken 32 N May 29 Andrew 110KB 12 KKING	398
33 N May 30 Saluzyn, Mark (9.4K) 34 N May 29 Yi David Miller [9.1] serial: convert early 35 N May 29 Yi David Miller [9.1] serial: set DTR in uar	uart 390 con for 8250 ct fo 390 cerial console
36 N May 29 Yinghai Lu (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yi Eric Duma Zet console: console hands as N May 29 Yi Eric Duma Zet console: console hands as N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x86: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x80: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x80: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x80: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) [PATCH 3/5] x80: initial fixmap st. 37 N May 29 Yinghai Lucian (2.3K) x80: initial fixmap st. 37 N May 29	upport. Diver 337 red console index for daying probe
40 N May 30 Wang Zhenyu (19K) [resend] [AGPGART] intel_agp: clea 41 N May 30 Dave Airlie (2.0K) [git pull] drm fixes for 2.5.22-ro 42 N Mau 29 Matt Helsleu (8.2K) [RFC][PATCH] Replacing the /broc/4	Kernel releases 3.2.0 - 3.6.0

Mark – embedded sound Axel – janitorial Hartley – comedi Al – vfs and filesystem Mauro – v4l Russell – ARM Johannes – intel wireless David – networking Eric - networking Greg – USB, staging, tty, etc.

2 N May 30 Robert P. J. Da (1.4K) should "create_proc_read_ent 3 N May 30 Jules Colding" (3.8K) SI MegaRAID problems 4 N May 30 The Comp (SISING Comp of the colding of the coldinary of the coldinar	ry" enforce read-only semantics? I fit
Greg Kroah-Hartman	work with 82 of RAM on Intel 965W
11 N May David S. Miller problem?	& out into 03338 a duplicate
14 N May 30 Bill Nottingham (1.00) [PATCH RFC] gla2xxx: fix tim 15 N May John Linville 16 N May John Linville 17 N May 30 Bill Nottingham (5.1%) [ROTTH] drivers/infiniband:	eout in q1a2x00 down timeout fix comparsio 2227 nsigned and
Mauro Carvalho Cheha	b unsigned 2041
21 N May Marki Brown	2155
24 N May 30 Stefan Richter (1.7k) [PATCH 2.6.21.3] ieee1394: e 25 N May Linus Torvalds ies wraparound is no 26 N May Linus Torvalds mp] 2/2: PCI: disable to the control of	th1394: bring back a parent device t treated in 1593ats e decode of INTS 1881 and 18
28 N May Andrew Morton	1360
31 N May A live (0.4K) kexec and aacraid broken 32 N May A live (2.9K) (2.9K) (3.3 N May 30 Salyzyn, Mark (9.4K)	796
34 N May David Airlie ATCH 4/5] serial: convert 35 N May David Airlie ATCH 3/5] serial: set DTR 886: initial fix	early_uart to earlycop for 8250 in uart for kerne fer al console map support
38 N May AxeliLin (2.2K) [PATCH 1/5] console: more but 39 N May AxeliLin (2.2K) [resend] [AGPGART] intel_agp (40 N May 30 Wang Zhenyu (19K) [resend] [AGPGART] intel_agp (2.0K) [git pull] drm fixes for 2.6 (42 N May 29 Matt Helsley (8.2K) [RFC][PATCH] Replacing the /	f for index pars 728 : use table for 728 : clean First releases 3.2.0 - 3.6.0 proc/(pid self)/exe symlink code

Greg – driver core, usb, staging
David – networking
John – wireless networking
Mauro - v4l
Mark – embedded sound
Linus - everything
Andrew – everything
James – SCSI
David - graphics
Axel - janitorial

Who is funding this work?		
1. "Amateurs"	13.8%	
2. Red Hat	10.6%	
3. Intel	9.4%	
4. Unknown Individuals	4.0%	
5. Texas Instruments	3.7%	
6. Novell	3.6%	
7. Linaro	3.4%	
8. IBM	2.8%	
9. Google	2.3%	
10. Consultants	2.2%	
U U	Kernel releases 3.2.0 – 3.6.0	

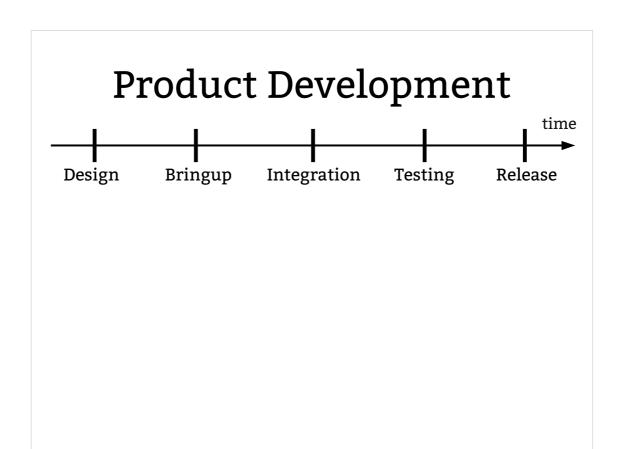
So you can view this as either 20% is done by non-affiliated people, or 80% is done by companies.

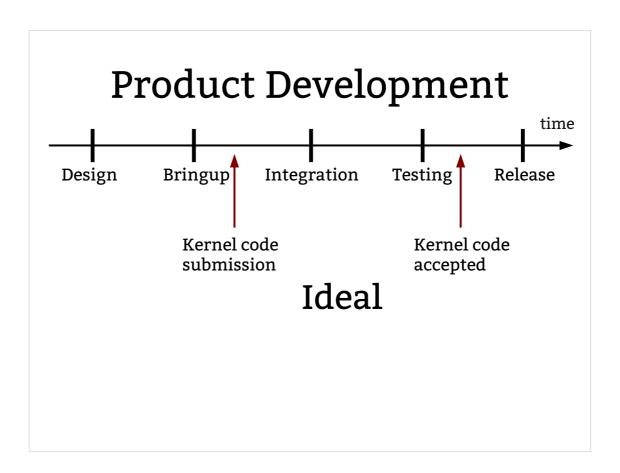
Now to be fair, if you show any skill in kernel development you are instantly hired.

Why this all matters: If your company relies on Linux, and it depends on the future of Linux supporting your needs, then you either trust these other companies are developing Linux in ways that will benefit you, or you need to get involved to make sure Linux works properly for your workloads and needs.

Who is funding this work? 11. Wolfson Micro 2.2% 1.9% 12. Samsung 13. Ingics Technology 1.7% 14. Oracle 1.7% 15. Qualcomm 1.6% 16. Vision Engraving 1.4% 17. NVidia 1.3% 18. Broadcom 1.3% 19. Linux Foundation 1.2% 20. Renesas Electronics 1.2% Kernel releases 3.2.0 - 3.6.0

Samsung 1047 patches LF – 501 patches Qualcomm 707 patches





"Working upstream saves time and money"

Dan Frye – VP Open Systems, IBM Dirk Hohndel – Chief Technologist, Intel



Obligatory Penguin Picture

