



Working with the Linux Community

CELF Jamboree
February 2008

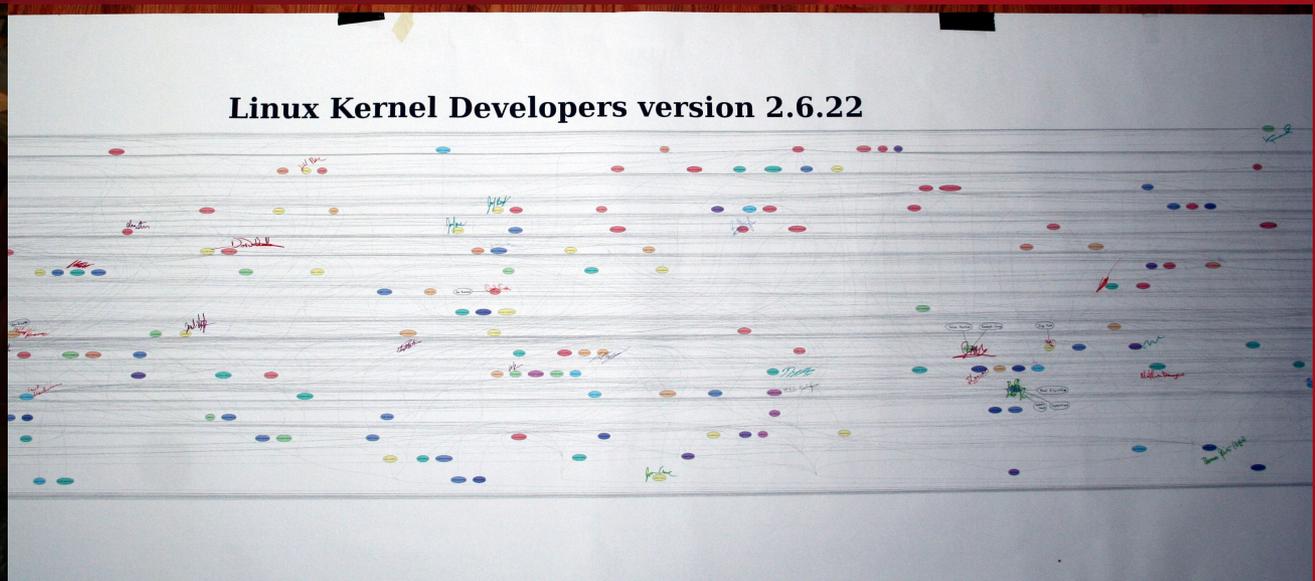


Overview

- **The Linux community and development process**
- **Benefits of working with the community**
- **How best to work with the community**
- **The GIT version control system**
- **Other resources**



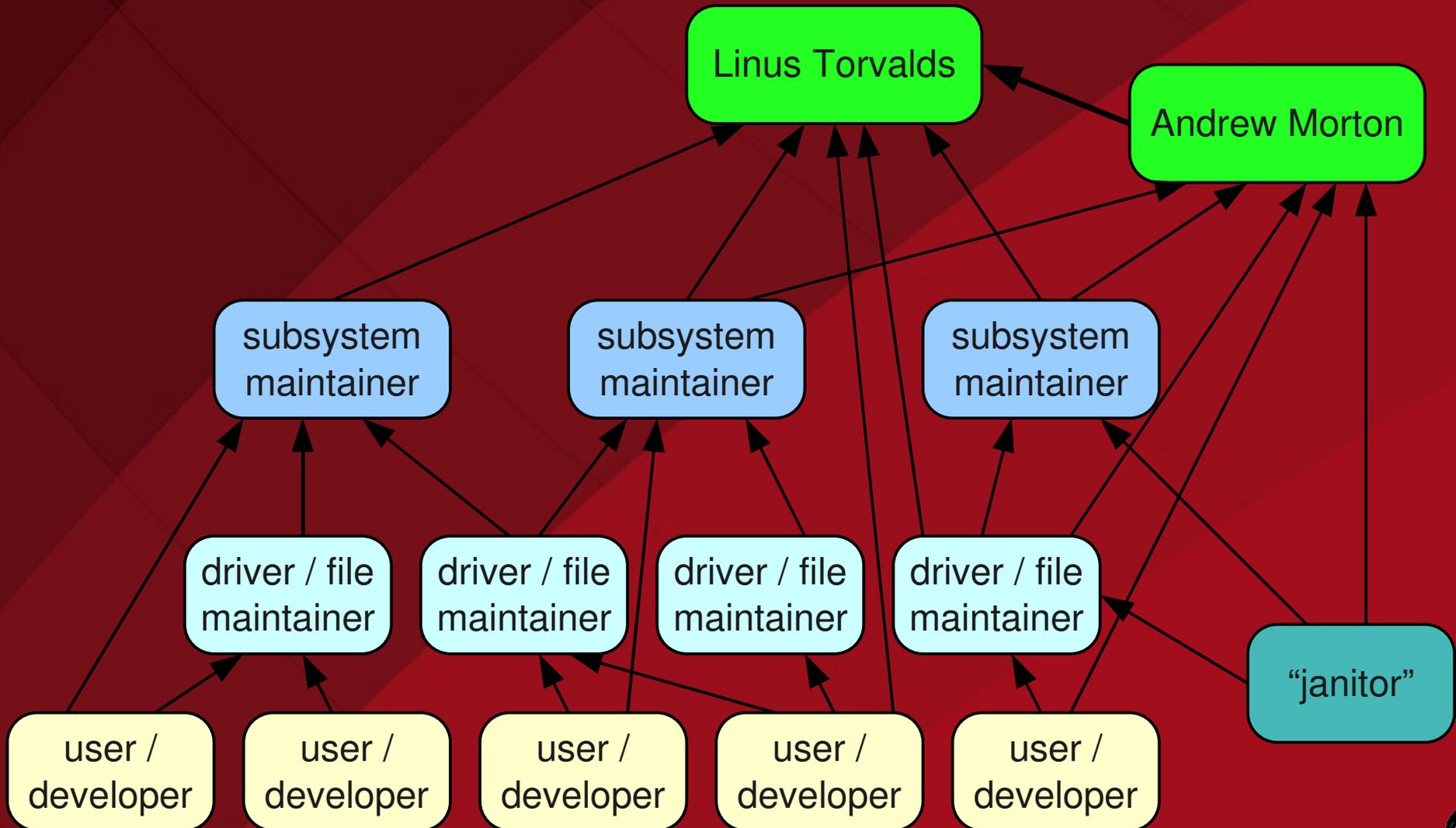
Who is the “Linux Community”?



- 2.6.22: 6840 commits from 911 developers
- 2.6.23: 7075 commits from 946 developers
- 2.6.24: 10353 commits from 1014 developers

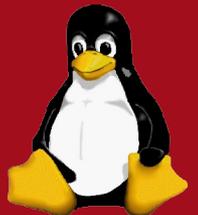


“Structure” of the Linux Community

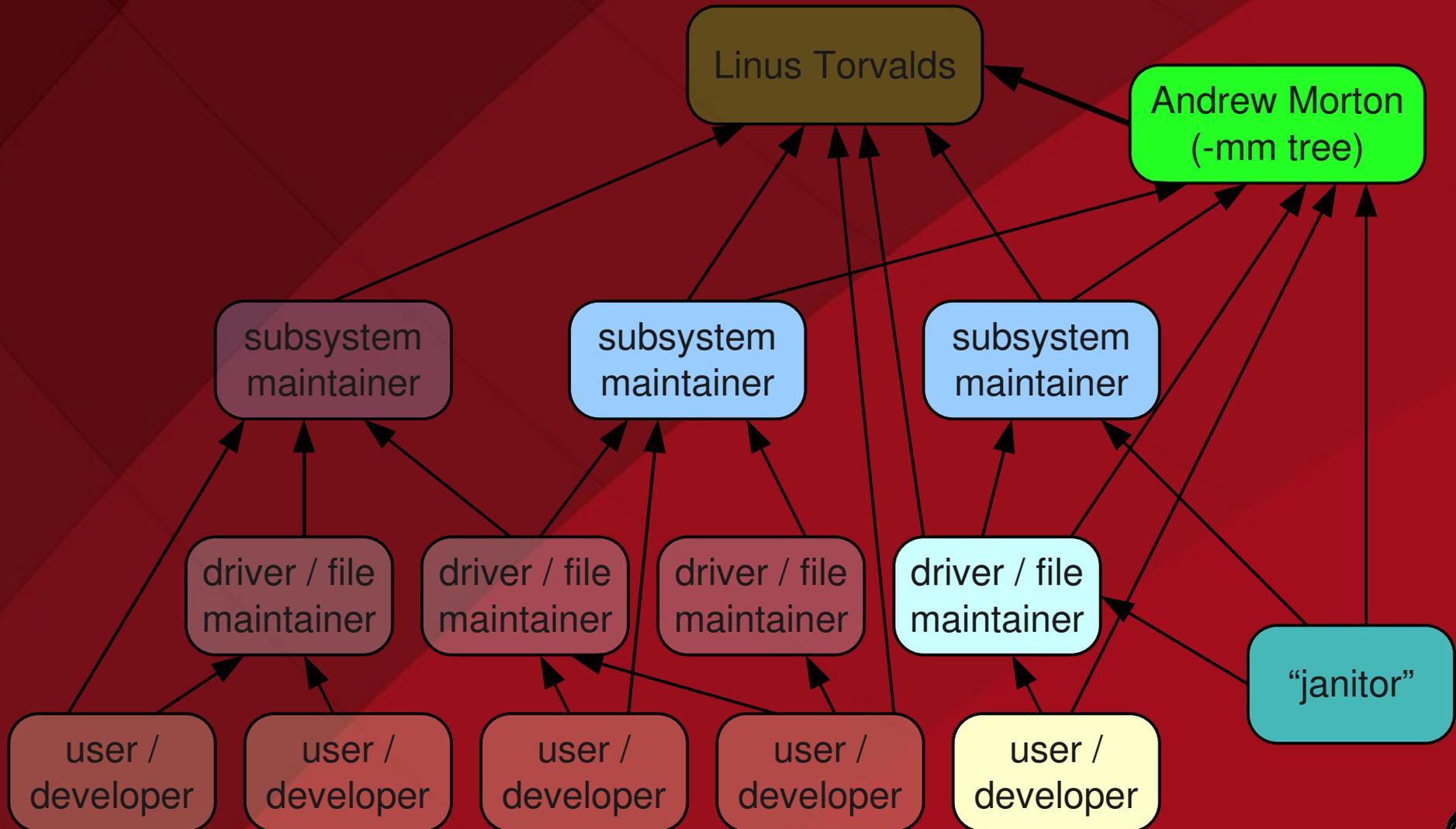


Andrew Morton's “-mm” tree

- Holding area for features to “mature”
- Subsystem git trees
- Large intrusive patch sets
- Small fixes and cleanups
- ... and everything else



Andrew Morton's “-mm” tree



Benefits of being close to “upstream”

- Ease of importing bug fixes
- Ease of merging new features
- Collaboration with others
- Testing



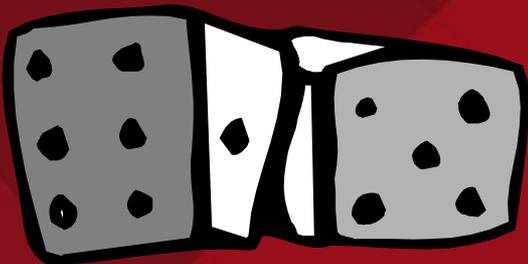
Benefits of sending code “upstream”

- External code review
- Avoidance of duplication
- Compile coverage testing
- Automatic bug checking (Coverity / “Stanford checker”)
- Code updated for changing APIs
- “Kernel Janitors”



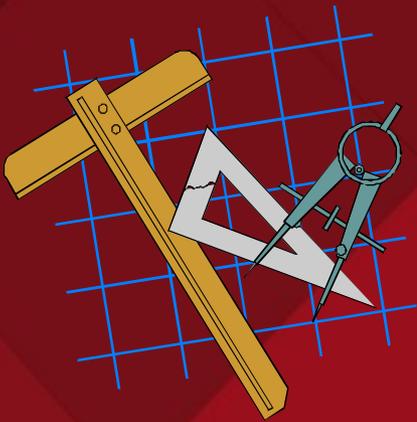
Improving your chances of acceptance

- Design in public
- Involve interested parties
- Work against current kernel
- Release early, release often
- Respond to feedback from reviewers
- Have patience



Approaching design

- Use existing infrastructure
- Avoid local hacks – *CHANGE* existing infrastructure
- Locking
- User API
- Avoid overengineering



Approaching design

“Linux is evolution, not intelligent design”

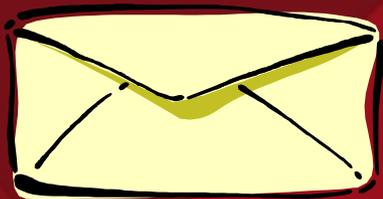
Linus Torvalds



Communication

- **Avoid SourceForge and web sites**
- **Use mailing lists**
- **Publish patches / GIT trees**
- **RESPOND TO FEEDBACK!**

<http://vger.kernel.org/vger-lists.html>

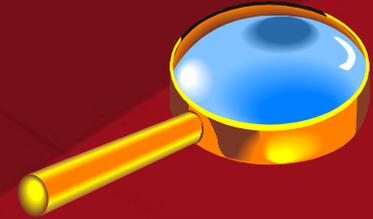


Coding Style

- **Avoid StudlyCaps**
- **Avoid typedefs**
- **No Hungarian Notation**
- **Short functions**
- **See `Documentation/CodingStyle`**



Submitting Patches



- Read the *patch* before you send it

```
--- a/drivers/net/wireless/libertas/assoc.c
+++ b/drivers/net/wireless/libertas/assoc.c
@@ -13,6 +13,7 @@

static const u8 bssid_any[ETH_ALEN] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF };
static const u8 bssid_off[ETH_ALEN] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char
*str);

static void print_assoc_req(const char * extra, struct assoc_request * assoc_req)
{
@@ -211,9 +212,17 @@ static int assoc_helper_channel(wlan_private *priv,
{
    wlan_adapter *adapter = priv->adapter;
    int ret = 0;
+   int restart_mesh = 0;

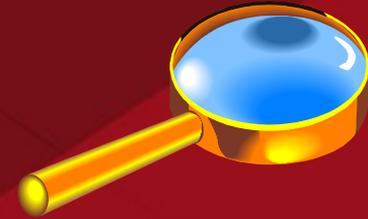
    lbs_deb_enter(LBS_DEB_ASSOC);

+   if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
+   {
+       libertas_mesh_config(priv,0,adapter->curbssparams.channel,NULL);
+       restart_mesh = 1;
+   }
+
    ret = update_channel(priv);
    if (ret < 0) {
        lbs_deb_assoc("ASSOC: channel: error getting channel.");
@@ -225,11 +234,13 @@ static int assoc_helper_channel(wlan_private *priv,
    lbs_deb_assoc("ASSOC: channel: %d -> %d\n",
        adapter->curbssparams.channel, assoc_req->channel);

+
    ret = libertas_prepare_and_send_command(priv, CMD_802_11_RF_CHANNEL,
        CMD_OPT_802_11_RF_CHANNEL_SET,
```



Submitting Patches



- Read the *patch* before you send it

```
--- a/drivers/net/wireless/libertas/assoc.c
+++ b/drivers/net/wireless/libertas/assoc.c
@@ -13,6 +13,7 @@

static const u8 bssid_any[ETH_ALEN] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF };
static const u8 bssid_off[ETH_ALEN] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
+extern int libertas_mesh_config(wlan_private * priv, int action, int channel, char
+str);

static void print_assoc_req(const char * extra, struct assoc_request * assoc_req)
{
@@ -211,9 +212,17 @@ static int assoc_helper_channel(wlan_private *priv,
{
    wlan_adapter *adapter = priv->adapter;
    int ret = 0;
+   int restart_mesh = 0;

    lbs_deb_enter(LBS_DEB_ASSOC);

+   if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
+   {
+       libertas_mesh_config(priv, 1, adapter->curbssparams.channel, NULL);
+       restart_mesh = 1;
+   }
+
    ret = update_channel(priv);
    if (ret < 0) {
        lbs_deb_assoc("ASSOC: channel: error getting channel.");
@@ -225,11 +234,13 @@ static int assoc_helper_channel(wlan_private *priv,
    lbs_deb_assoc("ASSOC: channel: %d -> %d\n",
        adapter->curbssparams.channel, assoc_req->channel);

+
    ret = libertas_prepare_and_send_command(priv, CMD_802_11_RF_CHANNEL,
        CMD_OPT_802_11_RF_CHANNEL_SET,
```



Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`

```
$ scripts/checkpatch.pl 0002-MESH-START-STOP_IOCTLs.patch
WARNING: line over 80 characters
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);

ERROR: "foo * bar" should be "foo *bar"
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);

ERROR: need space after that ',' (ctx:VxV)
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);
                                     ^

WARNING: externs should be avoided in .c files
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);

ERROR: That open brace { should be on the previous line
#19: FILE: drivers/net/wireless/libertas/assoc.c:219:
+     if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
+     {
ERROR: need a space before the open parenthesis '('
#19: FILE: drivers/net/wireless/libertas/assoc.c:219:
+     if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
```



Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`
- Send patches to yourself first

```
$ patch -p1 < saved-email.txt
patching file drivers/net/wireless/libertas/assoc.c
Hunk #1 FAILED at 13.
Hunk #2 FAILED at 212.
Hunk #3 FAILED at 234.
patch: **** malformed patch at line 71: adapter->meshid);
```



Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`
- Send patches to yourself first
- Read `Documentation/SubmittingPatches`

<http://www.linux.or.jp/JF/JFdocs/kernel-docs-2.6/SubmittingPatches.html>

Linux カーネルに変更を加えるための Howto です



Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`
- Send patches to yourself first
- Read `Documentation/SubmittingPatches`
- Include appropriate description

```
commit 4ac9137858e08a19f29feac4e1f4df7c268b0ba5
Author: Jan Blunck <jblunck@suse.de>
Date: Thu Feb 14 19:34:32 2008 -0800
```

```
Embed a struct path into struct nameidata instead of nd->{dentry,mnt}
```

This is the central patch of a cleanup series. In most cases there is no good reason why someone would want to use a dentry for itself. This series reflects that fact and embeds a struct path into nameidata.

Together with the other patches of this series

- it enforced the correct order of getting/releasing the reference count on `<dentry,vfsmount>` pairs
- it prepares the VFS for stacking support since it is essential to have a struct path in every place where the stack can be traversed



The GIT version control system

● Merge branch 'for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux-2.6	Linus Torvalds <torvalds@woody.us>	2008-02-15
● HID: add USB IDs for MacBook 3rd generation	Tobias Mueller <Tobias_Mueller@t-online.de>	2008-02-14
● HID: add LCSPEC from VERNIER to quirk list	Jiri Kosina <jkosina@suse.cz>	2008-02-11
● HID: fix processing of event quirks	Jiri Kosina <jkosina@suse.cz>	2008-02-08
● HID: Blacklist new GTCO CalComp USB device PIDs	Jeremy Roberson <jroberson@gtccorp.com>	2008-02-01
● Merge branch 'release' of git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux-2.6	Linus Torvalds <torvalds@woody.us>	2008-02-15
● Merge branches 'release', 'dmi', 'idle' and 'misc'	Linus Torvalds <torvalds@woody.us>	2008-02-15
● pnpacpi: __initdata is not an identifier	Roel Kluin <12o3l@tiscali.nl>	2008-02-12
● ACPI: static acpi_chain_head	Adrian Bunk <bunk@kernel.org>	2008-02-14
● ACPI: static acpi_find_dsdt_initrd()	Adrian Bunk <bunk@kernel.org>	2008-02-14
● ACPI: static acpi_no_initrd_override_setup()	Adrian Bunk <bunk@kernel.org>	2008-02-14
● thinkpad_acpi: static	Adrian Bunk <bunk@kernel.org>	2008-02-14
● ACPI suspend: Execute _WAK with the right arguments	Rafael J. Wysocki <rjw@sisk.pl>	2008-02-13
● ACPI: fix suspend regression due to idle update	Venkatesh Pallipadi <venkatesh.pallipadi@intel.com>	2008-02-12
● cpuidle: Add Documentation	Venkatesh Pallipadi <venkatesh.pallipadi@intel.com>	2008-02-14
● ACPI, cpuidle: Clarify C-state description in sysfs	Venkatesh Pallipadi <venkatesh.pallipadi@intel.com>	2008-02-12
● ACPI: DMI: quirk for FSC ESPRIMO Mobile V5	Len Brown <len.brown@intel.com>	2008-02-14
● ACPI: DMI blacklist updates	Len Brown <len.brown@intel.com>	2008-02-11
● Merge git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux-2.6	Linus Torvalds <torvalds@woody.us>	2008-02-15
● sh: Fix multiple UTLB hit on UP SH-4.	Hideo Saito <saito@densan.co.jp>	2008-02-14
● sh: fix pci io access for r2d boards	Magnus Damm <magnus.damm@denx.de>	2008-02-14
● sh: fix ioreadN_rep and iowriteN_rep	Magnus Damm <magnus.damm@denx.de>	2008-02-14
● sh: use ctrl_in/out for on chip pci access	Magnus Damm <magnus.damm@denx.de>	2008-02-14
● sh: Kill off more dead symbols.	Paul Mundt <lethal@linux-sh.org>	2008-02-13
● sh: __uncached_start only on sh32.	Paul Mundt <lethal@linux-sh.org>	2008-02-13
● sh: asm/irq.h needs asm/cpu/irq.h.	Paul Mundt <lethal@linux-sh.org>	2008-02-13
● serial: sh-sci: Fix up SH-5 build.	Paul Mundt <lethal@linux-sh.org>	2008-02-13
● sh: Get SH-5 caches working again post-unification	Paul Mundt <lethal@linux-sh.org>	2008-02-13



Basic GIT usage

- Clone
- Add
- Commit
- Push
- Pull / Merge

<http://www.kernel.org/pub/software/scm/git/docs/tutorial.html>



GIT Bisect

```
$ git-bisect start
$ git-bisect good 834f4ee6
$ git-bisect bad HEAD
Bisecting: 28 revisions left to test after this
[3174069fbb1e4b894ad0] OLPC: rework olpc_ec_cmd debugging output
$ git-bisect good
Bisecting: 14 revisions left to test after this
[e7911f321b8710557a26] config: enable es1371 as a module
$ git-bisect visualize
```



GIT Bisect

●	master	—	origin	—	bisect/bad	[JFFS2] Remove stray	David Woodhouse <dwmw2@infra	2007-10-13
●						[JFFS2] Handle dirents on the flash with embedded zero	David Woodhouse <dwmw2@infra	2007-10-13
●						[JFFS2] Check for creation of dirents with embedded zero	David Woodhouse <dwmw2@infra	2007-10-13
●						[JFFS2] Don't count all 'very dirty' blocks except in debug	David Woodhouse <dwmw2@infra	2007-10-13
●						[JFFS2] Check whether garbage-collection actually observed	David Woodhouse <dwmw2@infra	2007-10-13
●						[JFFS2] Relax threshold for triggering GC due to dirty blocks	David Woodhouse <dwmw2@infra	2007-10-13
●						[JFFS2] Trigger garbage collection when very_dirty_list is non-empty	David Woodhouse <dwmw2@infra	2007-10-07
●						[MTD] [NAND] Avoid deadlock in erase callback; release lock	David Woodhouse <dwmw2@infra	2007-10-07
●						bisect [MTD] [NAND] Restore correct timing register	David Woodhouse <dwmw2@infra	2007-10-07
●						[MTD] [NAND] Fix PCI ident table for CAFÉ NAND controller	David Woodhouse <dwmw2@infra	2007-10-07
●						Fix 'GPLv1 or later' and other copyright text in OLPC DC	David Woodhouse <dwmw2@infra	2007-10-06
●						cafe_ccic: default to allocating DMA buffers at probe time	Andres Salomon <dilinger@debian	2007-09-19
●						[PATCH] libertas: properly end commands on hardware	Pierre Ossman <drzeus-list@drzeu	2007-08-21
●						timekeeping: Prevent time going backwards on resume	Thomas Gleixner <tglx@linutroni	2007-09-16
●						OLPC: pm: only support STR on B2 and later	Andres Salomon <dilinger@debian	2007-09-10
○						bisect/good-e7911f321b8710557a26bed1a5ea16a632be2	Andres Salomon <dilinger@debian	2007-09-10



GIT Bisect

```
$ git-bisect bad
Bisecting: 1 revisions left to test after this
[4d03e22a47286441832e] [JFFS2] Trigger garbage collection when very_dirty_list size
becomes excessive
$ git-bisect good
Bisecting: 0 revisions left to test after this
[1e0c0ef8a8ddd84156b3] [JFFS2] Relax threshold for triggering GC due to dirty blocks.
$ git-bisect good
76b7506f1fd77698547e9f22b5f3fa3fde2a12f7 is first bad commit
commit 76b7506f1fd77698547e9f22b5f3fa3fde2a12f7
Author: David Woodhouse <dwmw2@infradead.org>
Date: Sat Oct 13 11:31:23 2007 +0100
```

[JFFS2] Check whether garbage-collection actually obsoleted its victim.

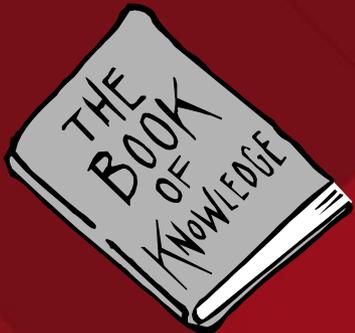
In OLPC trac #4184 we found a case where a corrupted node didn't actually get obsoleted when we tried to garbage-collect it. So we wrote out many million copies of it, in repeated attempts to obsolete it, until the flash became full. Don't Do That.

Signed-off-by: David Woodhouse <dwmw2@infradead.org>



Kernel Newbies

- Documentation
- Mailing lists
- IRC: *irc.oftc.net #kernelnewbies*
- Mentoring





Kernel Newbies

FrontPage - Kernel Newbies Japan - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://jp.kernelnewbies.org/

Google

FrontPage Login

Kernel Newbies Japan
最初のページ
カーネルハッキング
Webリソース
Mailing List

Wiki bits
Recent Changes
Help Contents
Find Page
Navigation Menu
WikiWall Terms of Service
Hosted by WikiWall

Web This site
Google Search

Immutable Page Info Attachments More A

FrontPage

KernelNewbies Japan

Web から得られる情報

たくさんのカーネル開発に関する情報が Web 上に既に公開されています。
[webresources](#) を参照してください。

カーネル2.6に含まれるドキュメントの日本語版

- [HOWTO Linux](#) カーネル開発のやり方の重要な事柄を網羅したドキュメントです
- [SubmittingPatches](#) Linuxカーネルに変更を加えるためのHowto です
- [SubmitChecklist](#) 開発者がパッチをより早く取り込んでもらいたい場合に、実践すべき基本的な事柄を紹介しています。
- [stable_api_nonsense.txt](#) この文書は、なぜ Linux ではバイナリカーネルインターフェースが定義されていないのか、またはなぜ不安のカーネルインターフェースとすることを説明しています
- [ManagementStyle](#) Linux カーネルのための好ましい (人によっては勝手に作った物だと言うかもしれません) マネージメントスタイルについて書かれた文書
- [Codingstyle](#) Linux カーネルのソースコードの好ましい書き方を説明しています
- [The perfect patch](#) Andrew Morton氏の The perfect patch の翻訳です

オープンソースやコミュニティの理解

- [Myths, Lies and Truths about the Linux Kernel](#) Ottawa Linux Symposium 2006におけるGreg Kroah-Hartmanによる基調講演の翻訳です
- [Linux 若と仕事をしてボクが学んだこと](#) Junio C Hamano氏の資料@第1 @OSDL-Japan Linux Symposium
- [To make OSS developers](#) 高橋浩和氏の資料@第1 @OSDL-Japan Linux Symposium
- [USAGI Project Experience](#) 吉藤英明氏の資料@第1 @OSDL-Japan Linux Symposium
- [Linux Kernel Development Getting Started](#) Randy Dunlap氏の資料@第4 @ The Linux Foundation Japan Symposium

カーネルハッキング

- [KernelHacking](#) ではカーネル開発をする上での参考になる書籍やプレゼンテーション資料を紹介しています。

[Linux プログラムを高速化](#) [組み込みLinux 講座](#) [コンテンツ ネットワーク](#) [アイダUL-400](#)
数回のマウス操作で、問題となる デジタル家電市場拡大で注目される Google の大規模な広告ネットワー 超高精度塑性成形機

Done



GPL licensing issues

These requirements apply to the modified work as a whole. ***If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works.*** But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.



GPL licensing issues

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. ***But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.***

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative ***or collective works*** based on the Program.





Working with the Linux Community

CELF Jamboree
February 2008

Questions / Discussion

