

# 30 entrevistas a desarrolladores del kernel de Linux



**Victorhck**  
**in the free world**

*omnia sunt communia!*



## Índice de entrevistas

Introducción.....	3
1.- Linus Torvalds.....	4
2.- Thomas Gleixner.....	6
3.- Sara Sharp.....	8
4.- Jean Delvare.....	11
5.- Greg Kroah-Hartman.....	14
6.- Dave Jones.....	16
7.- Paul Mundt.....	18
8.- Alan Cox.....	21
9.- Arnd Bergmann.....	24
10.- John W. Linville.....	26
11.- Johannes Berg.....	28
12.- Martin K. Petersen .....	30
13.- Julia Lawall.....	31
14.- Ben Hutchings.....	34
15.- Mauro Carvalho Chehab.....	36
16.- Jiří Slabý.....	40
17.- Laurent Pinchart.....	42
18.- Jiří Kosina.....	45
19.- Chuck Lever.....	48
20.- H. Peter Anvin.....	50
21.- Steven Rostedt.....	53
22.- Stephen Hemminger.....	57
23.- Glauber Costa .....	59
24.- Jonathan Corbet .....	62
25.- John Stultz.....	64
26.- James Bottomley.....	66
27.- Chris Mason.....	68
28.- Herbert Xu.....	70
29.- Frédéric Weisbecker.....	71
30.- Paul McKenney.....	73

## Introducción

Hace unos meses, en el 2012 la página Linux.com puso en marcha una serie de entrevistas semanales, a desarrolladores del kernel de Linux. Se trataba de una buena iniciativa para dar a conocer a las personas que están detrás del código.

Desde mi blog (<http://victorhckinthefreeworld.wordpress.com/>) has podido seguir puntualmente todas las semanas, todas las entrevistas traducidas, desde que empezaron hace ya 8 meses!! Empecé a traducirlas desde el inglés en su página original para difundirlas y darlas a conocer. Me parecieron interesantes, para conocer de primera mano cómo se trabaja, y cómo se desarrolla el kernel de Linux.

Espero que os hayan gustado. He tratado de hacer las traducciones lo mejor posible, y lo mejor que sabía. Algunas han sido más fáciles y otras más difíciles debido al lenguaje técnico. Siempre he estado abierto a mejoras de la traducción si veáis algún fallo. Siempre que sea posible, si en algún término dudais o no está claramente traducido acudid al original.

Si os han gustado y os han parecido interesantes me alegro, habrá sido un tiempo y un esfuerzo bien empleado. Y si os han servido de inspiración para profundizar y tratar vosotros mismo de meteros un poco más en el mundo del desarrollo del kernel entonces ya estoy más que satisfecho!

Si alguna vez os hacen una entrevista como estas mencionad que os picó el gusanillo a través de unas entrevistas que traducía un tipo en su blog.

Si quieres ver el original en inglés visita la página original **Linux.com**. Escrito por **Jennifer Cloer** a ella y la página original pertenecen los derechos de autor. La serie completa de entrevistas en inglés la puedes ver en este enlace:

<https://www.linux.com/news/special-feature/linux-developers>

Por supuesto una vez más dar las **GRACIAS** a Jennifer por realizar las entrevistas y a **Linux.com** por permitir la difusión de su trabajo.

Hace unos meses un habitual de mi blog, Xman, me propuso recopilar las entrevistas en formato PDF para distribuirlas a aquellos que quizás tienen más restringido el uso de internet, el estaba pensando en sus compatriotas cubanos.

Me pareció una idea interesante, así que me propuse recopilarlas y difundirlas en formato PDF de libre descarga y uso. Este trabajo está licenciado con licencia CC-by-sa-nc.

Puedes compartirlo pero siempre respetando los términos de la licencia, citando expresamente la fuente original del artículo en inglés, y este blog como creador de la traducción.



Gracias a los que habéis pasado por el blog y habéis mostrado interés, espero que esta nueva entrega recopilatoria sea de vuestro agrado, y por supuesto a los protagonistas de las entrevistas, incluso alguno me mandó un correo y se puso en contacto conmigo.

Empezamos...

# 1.- Linus Torvalds

**¿Cual es tu nombre?**

Linus Torvalds

**¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?**

Soy el supervisor último del kernel, y no estoy encargado de ningún subsistema en particular directamente; si bien, ocasionalmente me involucro directamente con la capa VFS (VFS layer) (y muy de vez en cuando algunas de las discusiones VM).



**¿De quien recibes tu nómina?**

The Linux Foundation.

**¿En qué parte del mundo vives, y porque allí?**

Portland, Oregon, y al porqué allí, es sobre todo porque es mucho más tranquilo y más habitable que Silicon Valley, donde vivimos durante varios años antes de mudarnos aquí. El clima puede que no sea tan genial, pero es una zona mucho mejor para que crezcan los niños, creo. Y pudimos comprar una casa más grande en un buen distrito escolar.

**¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Realmente sólo utilizo un navegador web (tanto para correos, como para pasar el rato) y también varias terminales en las que uso git. En ocasiones uso una ventana gitk, para ver el historial git. La mayoría de mi tiempo lo empleo en leer (y responder) correos electrónicos, fusionando diferentes ramas de software y comprobando los resultados.

La otra herramienta que utilizo es “perf” una herramienta que realiza perfiles de rendimiento para las cargas de las que me ocupo (que principalmente son compilar el kernel y otros trabajos en git).

**¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Je. No tenía el suficiente sentido común y el conocimiento para saber que escribir tu propio sistema operativo era una gran cantidad de trabajo.

**¿Qué es lo que hace que sigas interesado en esto?**

Todavía me gusta el bricolaje, y además la parte técnica de la misma. El hecho de que actualmente, en realidad es bastante social, y que pueda llamar a las personas por su nombre, es además un extra.

**¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Creo que mi parte favorita es cuando alguien hace algo completamente loco utilizando Linux. Cosas que simplemente no tienen mucho sentido, pero que son muy impresionantes desde el punto de vista técnico (y aún más impresionantes desde el punto de vista de “¿han pasado varios meses haciendo \*eso\*?”)

Por ejemplo cuando Alan Cox estuvo trabajando en exportar Linux al procesador 8086. O el tipo que hizo su propio ordenador usando un microcontrolador de 8 bits que el mismo cableó a la memoria RAM y a una tarjeta SD, y escribió un emulador ARM para él, y consiguió arrancar Linux (de manera muy, muy, muy lenta) en su “ordenador”.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Empezar despacio. Ni siquiera tiene por que ser Linux, hay un montón de proyectos de código abierto que necesitan ayuda, y deseas aprender cómo participar. Y una vez que te des cuenta de que el carácter de los programadores es algo tímido, y quieres involucrarte con la programación del kernel, no trate de revolucionar algo esencial del código del kernel – trate de encontrar una cierta preocupación persistente muy pequeña y solucione esa pequeña cosa. Tal vez un controlador de hardware al que tienes acceso pero que no funciona tan bien como debería, cosas así.

Lleva un tiempo aprender los entresijos, y lo que realmente ayuda es si la gente puede ver que has hecho otras cosas antes de empezar a mandar parches de mayor categoría.

Pero lo más importante es el “buen gusto”. Es difícil de describir, pero es algo que personalmente busco. Las personas que hacen las cosas de la “manera correcta” – y no me refiero a que debes seguir todas las reglas que han surgido con los años (aunque se debe hacer eso, también) -, si no que estoy hablando de esa rara cualidad de escribir código que tiene sentido obvio y hace lo correcto sin un montón de casos especiales o de manera compleja, pero también sin ser innecesariamente abstracto y de propósito general. “Haz una cosa, y hazla bien.”

### **¿Qué escuchas mientras programas?**

Oh, me gusta que la oficina este totalmente en calma. Escucho música cuando salgo con el coche a llevar a mis hijos de excursión, etc. Pero cuando estoy trabajando no me gusta oír nada. No sólo música, ni tampoco el ruido de los ventiladores de mis ordenadores. Sólo el silencio.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

No entro en ningún canal IRCo ni uso ningún otra herramienta de comunicación en tiempo real. Hago todo por correo electrónico. Sigo las listas de correo de kernel y git, pero incluso en esto tengo activado el modo “auto-archivar”, así que sólo veo los temas que busco expresamente, o en los que estoy con copia (cc) o los dirigidos directamente a mí.

En cuanto a las conferencias, simplemente la cumbre de Linux Kernel. Trato de ir a LinuxConf Australia la mayor parte de los años. Me gusta como conferencia y es en Australia durante su verano. Pero esta conferencia es una de esas cosas que funciona sola, así que es probable que vaya sólo cada 2 años o más.

Hay algunas otras conferencias a las que voy, normalmente porque se celebran en sitios interesantes y así de paso aprovecho para realizar algo de buceo en el mismo viaje.

## 2.- Thomas Gleixner

### Nombre.

Thomas Gleixner, nick: tglx

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Algunos me consideran “un viejo gruñón”. Debido a mi edad, y a esta edad ya hace que uno no tenga que estar aguantando chorradas (N.T: revisa el original y propón en los comentarios una traducción mejor)

Como mantenedor soy el responsable del núcleo de infraestructura de los “timers”, “timekeeping” y el manejo de las interrupciones. Soy parte del equipo de mantenedores de la arquitectura x86, y el mantenedor y principal desarrollador del parche del kernel “real time preemption”. Además de todo esto tengo una especial afinidad por la misión de limpieza de código imposible, y organización de la rama principal del kernel.



### ¿De dónde recibes tu nómina?

De mi propia empresa, que obtiene parte de una devolución por los contratos con Red Hat y otras personas interesadas en mi trabajo.

### ¿En qué parte del mundo vives, y porque allí?

En Alemania. Es mi patria, ¿por qué vivir en otro sitio?

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Herramientas de la línea de comandos. No trates de liarme en discusiones tipo Emacs vs. VI y no me preguntes sobre herramientas con interfaz gráfica

### ¿Que es lo que tienes funcionando en tu PC?

Cambio de sabores de las distribuciones de Linux. Mis requisitos de escritorio son más bien bajos: Administrar un “montonazo” de terminales, utilizar un navegador web gráfico y, en ocasiones algunas aplicaciones GUI inevitables.

Estoy tratando desesperadamente de evitar la nueva aplicación tan de moda impulsados por los equipos de “escritorio”, que insisten en saber mejor que yo cómo administrar mi flujo de trabajo de una manera eficiente.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Curiosidad

### ¿Qué es lo que hace que sigas interesado en esto?

La diversión que conlleva. Trabajar con gente inteligente de todo el mundo.

### ¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?

Esa es una pregunta difícil. Yo tengo mis favoritos de todas las categorías, pero en lo que se refiere

a tonterías, esta es mi favorita:

```
+ d->core_internal_state__do_not_mess_with_it |= SOME_CONSTANT;
```

Ver este enlace: <http://www.spinics.net/lists/linux-tip-commits/msg11099.html>

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Encuentra el área que más te interese y empezar a resolver los problemas que te afectan.

**¿Qué escuchas mientras programas?**

Los pensamientos a la deriva a través de mi cerebro.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Lista de correo: Principalmente LKML (Linux Kernel Mailing List)

Canales IRC : Mi nick es único

Conferencias : Muchas y variadas”

### 3.- Sara Sharp

**Nombre:**

Sara Sharp

**¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?**

Soy responsable de mantener los driver xHCI del kernel Linux. Me encargo del soporte para el USB 3.0Y mando mis parches a Greg Kroah-Hartman, que es el mantenedor del subsistema USB

**¿De dónde recibes tu nómina?**

Trabajo en el centro tecnológico de código abierto de Intel, con unos cuantos buenos desarrolladores del kernel de Linux

**¿En qué parte del mundo vives, y porque allí?**

Vivo en Portland, Oregon. Hay una razón por la que decimos “Manténga Portland extraña”. Tenemos de todo, desde Donuts Voodoo hasta carreras de bicicleta “Zoo bombers” o Powell’s, que es la librería más grande de América. Portland también está volcada con el desarrollo del software de código abierto.



Es la casa de OSCON (Open Source CONvention), y hay una gran compenetración entre los usuarios de Linux y la comunidad de bicicletas. Los constructores de electrónica tambien estan muy unidos a Portland. No olvides visitar el encuentro Dorkbot cuando estés por la ciudad, o pasa un rato con las mujeres más tecnológicas de Code N Splode.

**¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Um, soy más bien de usar vim y mutt. El desarrollo de software para mí es todo texto. Y sobre los que ejecuto en mi Pc de escritorio, es Linux, por supuesto. Utilizo Debian o Ubuntu, y trato de hacer todo mi trabajo en Linux.

**¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Greg Kroah-Hartman habló con mi profesor de informática, Bart Massey, y le dijo que estaba buscando un estudiante para desarrollar un proyecto en Linux con USB. Bart pensó que yo sería perfecta, fue mi tutor en la facultad para un proyecto que hice para “usbfs2”. Trabajé en usbfs2 en la universidad del estado de Portland, y después en un proyecto pagado por el programa de investigación para no graduados de Intel.

Estaba muy nerviosa al enviar mi primer parche, pero mi novio de entonces, y ahora marido, me animó ha hacerlo y enviarlo. La comunidad USB de Linux era un buen lugar para empezar a desarrollar el kernel, y la gente de la lista de correo fue muy paciente con mis preguntas de novata.

Un par de meses antes de que me graduara, me propuse hablar de mi proyecto con usbfs2 en OSCON 2007. Kristen Accardi estaba en el comité de selección de OSCON, y me recordaba de algunas reuniones de Linux en Portland a las que yo había asistido. Ella sabía que el Centro



Tecnológico de Código Abierto de Intel estaba buscando algún desarrollador de USB en Linux, y me consiguió una entrevista. He estado trabajando en este CTCIAI en los últimos 5 años, hasta hoy que continuo mi trabajo en el subsistema de USB Linux.

Yo no sería una desarrolladora del kernel de Linux hoy en día sin los contactos que he creado en red con otros desarrolladores en conferencias y eventos de tecnología.

### **¿Qué es lo que hace que sigas interesada en esto?**

La gente hace que me siga gustando. Me encanta aprender y discutir nuevas ideas con la comunidad del kernel de Linux, y con mis compañeros de Intel. Por supuesto que hay discusiones subidas de tono de vez en cuando, pero la mayoría de los desarrolladores, son lo suficientemente amables para tomarse un tiempo y responder a algunas preguntas que hago.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Me divertí un montón con Greg KH con las descripciones de mis peticiones (pull request descriptions). Aquí un extracto de una respuesta que me envió:

*Date: Thu, 26 May 2011 00:04:50 -0700*  
*From: Greg KH*  
*To: Sarah Sharp*  
*Cc: linux-usb@vger.kernel.org*  
*Subject: Re: [RFC 0/3] xhci: Quitar depuración inútil*

*On Wed, May 25, 2011 at 04:28:51PM -0700, Sarah Sharp escribió:*

- > El driver xHCI que tiene ya? dos años? Es hora de ponerle*
- > sus pantalones de chica mayor y quitar toda esa basura inútil de información de desarrollo.*

*Ahora dame una toallita húmeda para limpiar de mi portátil el café que acabo de resoplar por la nariz y que lo ha puesto todo perdido.*

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Encuentra un proyecto mediano, dentro de la comunidad del kernel de Linux, que tenga lista de correo. No pierdas tu tiempo en un montón de parches arreglar ortografía. Un par de parches de errores revisión debe ser suficientes para iniciarse en el proceso de envío de parches, pero en algún momento tiene que seguir adelante y empezar a hacer contribuciones más útiles y más grandes.

Encuentra un mentor. No tiene por que ser alguien que sea un mantenedor de algún subsistema de Linux. Simplemente vale con que sea alguien que sepa principios básicos de git y pueda revisar tu código, y ayudarte a configurar tu cliente de correo, para enviar tus parches. Te puedes unir con un amigo para revisar el código en común y tratar de averiguar un subsistema del kernel Linux juntos.

### **¿Qué escuchas mientras programas?**

No me puedo concentrar mientras programo cuando oigo voces, así que me inclino a escuchar música electrónica de ritmos lentos, música clásica o bandas sonoras. Suelo escuchar Daft Punk, Justice, Hans Zimmer, and Klaus Badelt.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Estoy en la lista de correo linux-usb@vger.kernel.org. Y en cuanto a conferencias, asisto a OSCON, Open Source Bridge, LinuxCon America, Linux Plumbers Conf, Linux Kernel Summit,

and Linux Conf Australia. Este año también estaré en AdaCamp D.C, representando a las mujeres dentro de código abierto.

## 4.- Jean Delvare

### Nombre:

Jean Delvare

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

En la actualidad soy co-mantenedor de 2 subsistemas del kernel de Linux: I2C (soporte del núcleo de I2C, SMBus y protocolos relacionados, más una docena de drivers controladores SMBus para máquinas x86) y hwmon (monitoreo de la salud del hardware, por ejemplo la velocidad de los ventiladores o las temperaturas.)



En un principio estaba manteniendo ambos todo por mi mismo, pero la carga de trabajo ha ido creciendo con los años, más de lo que era posible de manejar, así que tuve que encontrar co-mantenedores. Ahora todo va bastante bien.

También contribuyo en otras áreas, como por ejemplo, drivers de tarjetas gráficas V4L o DVB

### ¿De dónde recibes tu nómina?

Técnicamente, de Novell SARL, Francia. Pero desde una perspectiva lógica, de SUSE, unidad de negocio del grupo Attachmate.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en Francia, más concretamente en Mérignac, cerca de Burdeos, en el sur oeste de Francia.

Francia, simplemente porque nacía allí. Mérignac porque... bueno es una relativamente larga historia, pero interesante. Cuando la que sería mi esposa y yo dejamos la escuela secundaria en 2001-2002, pasamos unos tiempos difíciles buscando trabajo. La época que siguió al 11 de Septiembre, no era buenos momentos para los que estábamos empezando, porque muchas empresas tecnológicas pararon sus contrataciones durante ese tiempo. A pesar de que originalmente no queríamos vivir en París, después de un año sin encontrar trabajo, tuvimos que cambiar de ideas y aceptar que tendríamos que empezar desde allí.

Así que terminamos pasando 3 años en París, desde 2003 al 2009, trabajando para varias empresas de consultoría tecnológica. La mayoría de los encargos no tenían interés ninguno, y pasamos mucho tiempo en transportes públicos, además de la cantidad de dinero gastado en alquileres. Pero por lo menos ganamos experiencia. A finales de 2005, mandé de forma espontánea una aplicación tanto a Red Hat como a SUSE. Y tuve una respuesta positiva por parte de SUSE, y empecé a trabajar para ellos en Abril de 2006, desde casa.

Mientras tanto mi esposa negoció un proyecto en “cualquier sitio” en la mitad sur de Francia. Ese “cualquier sitio” terminó siendo Burdeos. Como yo llevaba ya 4 meses trabajando desde casa sin problemas, me pude permitir el seguirla hasta allí. Elegimos Mérignac porque está el aeropuerto de Burdeos, muy cómodo cuando tengo que viajar a Nuremberg o Praga para reunirme con mis compañeros de trabajo.

Desde entonces, hemos tenido 2 hijos. Mi esposa interrumpió su carrera, para criarlos. Compramos un piso de tres plantas más arriba en el que originalmente estábamos de alquiler. Es un buen lugar para criar a unos hijos, y no tenemos intención de mudarnos.

### **¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Soy un hombre de gustos simples. Un terminal, y GCC me hacen feliz. Quilt para la gestión de parches y Git o Subversion para la gestión de código fuente. Utilizo Nedit como editor de texto, que es realmente una elección poco frecuente.

### **¿Qué es lo que tienes ejecutando en tu PC de escritorio?**

Como sistemas operativos: Suse Linux Enterprise Desktop para el trabajo y openSUSE en mi ordenador personal.

Como entornos de escritorio: He cambiado muchas veces. Di una oportunidad a KDE 3 cuando me uní a SUSE, pero cuando migré a KDE 4 dejaron de funcionar mis atajos de teclado, así que cambié a Gnome 2. En mi ordenador personal tuve Xfce durante un tiempo, hace poco dí una oportunidad a Gnome3, pero estoy convencido de que regresaré a Xfce pronto porque Gnome 3 consume muchos más recursos de lo que mi pobre tarjeta gráfica puede soportar.

Y como aplicaciones, Firefox como navegador web, Claws como mi cliente de correo para desarrollo y gestiono mi música con MPD.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Cuando me cambié de Windows a Linux en el 2001, Linux no me decía la temperatura de la CPU ni la velocidad del ventilador de la CPU. Yo quería saber esos datos, así que me uní al proyecto lm-sensors y ofrecí mi ayuda para testarlo. Esos tipos eran muy agradables, así que me quedé con ellos. En 2005 (IIIRC) nuestro código se unió al kernel, tomando la forma subsistemas “i2c y hwmon” ayudé en esa tarea. En ese tiempo los miembros legendarios de proyecto lm-sensors se marcharon, así que me convertí en el nuevo líder “de-facto”. Y en eso estoy hoy en día.

### **¿Qué es lo que hace que sigas interesada en esto?**

Dos aspectos: El reto técnico, y el ayudar a gente real a solucionar problemas reales. Un nuevo reto técnico aparece cada mes. Un nuevo componente al que dar soporte, posiblemente con alguna característica inesperada. Programar un nuevo estándar en el kernel, con el que el código existente tiene que convivir. Una nueva API con la que podamos reducir cientos de drivers a sólo algunos para ganar en rendimiento o tamaño, etc. Esto nunca acaba

Del mismo modo, la gente viene a nosotros cada semana con un problema que resolver. A veces muy simple, a veces muy complejo. Siempre se siente uno bien cuando puedes ayudar. Ah, y por supuesto, ahora que me están pagando por eso, esta es una razón más, muy buena para seguir haciéndolo.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Con el tiempo, he aprendido a alejarme de las discusiones extremas. Tengo maneras más divertidas de pasar mi tiempo cuando lo deseo.

Peticiones de código tontas, he visto pocas, nada comparado con lo que tuve que sufrir en mi trabajo anterior en el que trabajaba con código fuente cerrado. Esto es lo verdaderamente increíble

en comparación con los proyectos de código abierto. No tienes porque estar de acuerdo con todas las propuestas, pero mur rara vez son tontas.

Los mayores logros no se dan en nuestra área de trabajo. No puedo recordar una sola cosa que por ella misma me quedara impresionado a primera vista. Esto es debido a que no somos gente de marketing. En vez de eso somos como hormigas incansables. Lo increíble no es lo que hice hoy, si no lo que he hecho en estos 10 años hasta ahora, y lo que haré en los próximos 10, mientras sigan requiriendo de mi trabajo. Lo que es increíble es que hace 10 años no abandonamos cuando Linux se encontraba tan lejos de sus competidores, en muchos campos. El progreso es lento, pero es constante.

Me recuerda un texto de una carta de un juego de magia, que decía así:

“Las plantas dijeron: Lucharemos contra las piedras con las raíces, el tallo y las semillas. Somos pacientes. Ganaremos”

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Ser paciente y escuchar. No lo era al principio, y estoy satisfecho de que Greg KH (Kroh-Hartman) me enseñara.

### **¿Qué escuchas mientras programas?**

Tengo gustos musicales relativamente amplios. Depende de la combinación de mi estado de ánimo y lo que actualmente estoy tratando de lograr. Escribir código nuevo es diferente de la depuración o la revisión de código de otros.

Se extiende desde la música clásica (Bach, Mendelssohn, Saint-Saëns) hasta varios tipos diferentes de metal (Dream Theater, Ayreon, Nightwish). Pero también me gusta escuchar más común internacional (Radiohead, Muse, The Corrs) o el pop/rock francés (Goldman, Souchon, Voulzy.) E incluso puede sumergirse en cosas más clásicas, (Bob Dylan), o disfrutar de descubrir nuevos talentos (recientemente:Nikki Yanofsky)

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Solía estar muy presente en canales IRC, pero ahora los frecuento menos, desde que me dí cuenta que me podían distraer fácilmente. Trabajando en casa con niños cerca, no necesito otra fuente de distracción en mi trabajo. Cuando me conecto a los canales IRC, estoy en freenode en las salas: #linux-sensors, #v4l y #linuxtv. A veces en #ffmpeg o en #x264 cuando tengo problemas con códecs de vídeo.

No soy un asiduo a las conferencias, depende de la oportunidad y del lugar. Fui a Sucon en el 2004, Guadec en el 2005, LGM y OLS en 2006, Kernel Summit en 2008 y 2011, y a las conferencias privadas Suse Labs muchas veces.

## 5.- Greg Kroah-Hartman

### Nombre:

Greg Kroah-Hartman, pero comunmente conocido como “Greg K-H” porque nadie sabe cómo pronunciar mi nombre.

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Actualmente gestiono los lanzamientos de la rama estable del kernel, y soy mantenedor de subsistemas de USB, el núcleo de los controladores, lanzamiento de los drivers, tty, char, y algún otro menor.

### ¿De dónde recibes tu nómina?

Trabajo para Linux Foundation.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en Norte América, en la parte noroeste del país, justo a las afueras de Seattle, Wasintong. Vivo allí porque es un gran lugar para vivir, buen clima, y un gran lugar para formar una familia



### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Convivo con mi cliente de mail (mutt), y mi editor (vim), y no podría sobrevivir sin ellos. Otras herramientas que utilizo diariamente son Git y Quilt para el desarrollo del kernel, Chrome y Firefox para navegar por la web e irssi para las charlas en salas IRC. Normalmente tengo GNOME3 en mi PC de escritorio, pero a veces termino frustrado con él y regreso a OpenBox o i3m. De vez en cuando también pruebo KDE para asegurarme de que no me estoy perdiendo nada.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Yo era un desarrollador de software embebido, probando el dispositivo en el que estaba trabajando (un escaner de código de barras) con diferentes sistemas operativos para asegurarme de que tenía el firmware USB correcto. Linux tenía poco soporte USB en aquel tiempo, y me dí cuenta de que podía ayudar y contribuir para hacer que funcionara mejor. Una cosa llevó a otra y de pronto me encontré con un trabajo a tiempo completo desarrollando el kernel de Linux, de eso hace ya 10 años, y nunca miré atrás.

### ¿Qué es lo que hace que sigas interesado en esto?

Asegurarme que Linux funcionará de manera correcta en todos los nuevos dispositivos que se creen. Como siempre habrá nuevos dispositivos y plataformas, no creo que nunca me aburra.

### ¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?

Creo que la cosa más asombrosa es que nunca sabrás cuando te vas a encontrar con alguien en persona con la que has tenido trato sólo mediante correo electrónico. Un buen ejemplo de esto fue hace un año en la República Checa, en una conferencia de Linux (LinuxCon europe 2011). Un grupo de desarrolladores fuimos juntos a un gimnasio de escalada una tarde, y de repente me encontré a mi mismo escalando junto con otro desarrollador del kernel que trabajaba para otra compañía diferente, alguien a quien había rechazado en el pasado por diferentes razones, y que

eventualmente aceptó después de un número de diferentes interacciones. Así que después de ese incidente siempre pienso “ser siempre amable por correo electrónico, ya que nunca sabes cuando, la persona del otro lado de correo, te va a sujetar por una cuerda tratando de mantener tu seguridad”

La otra cosa maravillosa es relativa al proceso, que este se centra en las personas, no en la compañía para la que trabaja. La gente cambia de trabajos constantemente, pero aún así seguimos trabajando juntos, en las mismas cosas, y encontrándonos en todo el mundo en diversas partes, sin importar en la compañía para la que trabajes

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Simplemente empieza con algo pequeño y encuentra algo que te interese. Tenemos un montón de áreas que necesitan ayuda, y no faltan áreas interesantes que lleven a Linux al futuro.

### **¿Qué escuchas mientras programas?**

Mi radio local en [www.kexp.org](http://www.kexp.org), es lo que normalmente suelo escuchar mientras reviso código, o hago mantenimiento del kernel estable. A veces necesito hacer tareas en las que hay “que pensar más” y normalmente en esos casos escucho una de las mejores músicas para programadores([musicForProgramming\(\)](http://musicforprogramming.net/)), una mezcla que puedes encontrar en <http://musicforprogramming.net/>

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Estoy en las listas de correo de [linux-usb](mailto:linux-usb@kernel.org) y [linux-kernel](mailto:linux-kernel@kernel.org), junto con otras listas de correo relacionadas con Linux, y estoy tratando de dejar los canales IRC, ya que son una distracción enorme. Me puedes encontrar en casi todas las conferencias de Linux Foundation, y también en otras conferencias que se celebran durante todo el año (CUSEC, Linux Plumbers, etc.)

## 6.- Dave Jones

### Nombre:

Dave Jones

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Soy el líder del equipo del kernel Fedora. Parte de esa responsabilidad consiste en tratar con los errores del kernel con los usuarios, lo que hace que al final tenga que estar encima de todo el kernel. Hace poco renuncié a mantener el subsistema de cpufreq. Tratar con los errores que los usuarios de Fedora encuentran (y otros que encuentro yo mismo) consume todo mi tiempo disponible.

### ¿De dónde recibes tu nómina?

Red Hat.

### ¿En qué parte del mundo vives, y porque allí?

Boston, MA, EEUU. Principalmente porque la ingeniería de Red Hat está afincada aquí.



### ¿Cual es tu herramienta favorita para el desarrollo de software?

Nada especial. Probablemente las mismas herramientas que la mayoría que los otros desarrolladores usan. Unos cuantos scripts de consola para automatizar un montón de tareas aburridas que son parte de mi trabajo, como la interacción con bugzilla, etc. He estado trabajando en una herramienta para encontrar errores en el kernel de una manera mucho más rápida (que parece que ha tenido poco éxito) <http://codemonkey.org.uk/projects/trinity/>

### ¿Y que tienes instalado en tu PC?

Xfce

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Necesité programar mi propio kernel, porque ninguna de las distros que probé tenían lo que necesitaba. Y la característica que necesitaba estaba sólo disponible en la rama que se encontraba en desarrollo en ese momento (que era en la época del kernel 2.1.X). No recuerdo qué es lo que era, pero tenía algo que ver con VFAT. Las cosas no eran siempre estables, así que hice el hábito de actualizarlo regularmente (reportando los últimos tarball en un disco zip desde la universidad a casa). Empecé a mandar parches allí donde veía cosas que creía que podría mejorar. Estoy tratando de recordar mi primer logro. Puede que haya sido mejorando AFFS durante la serie 2.1.X. Había en aquel entonces un buen puñado de cosas menores por mejorar.

### ¿Qué es lo que hace que sigas interesado en esto?

Una aparente fuente infinita de nuevos errores.

### ¿Cual sería tu consejo para los desarrolladores que quieran implicarse?

Centrarse en un aspecto particular del kernel que te interese, y meterse de lleno en él. Empieza poco a poco, pero vaya tirando hacia arriba. No aprenderás mucho resolviendo errores de ortografía



o realizando transformaciones repetitivas guiado por lo que te dice el “checkpatch”. Encuentra un problema, entiéndelo, y trata de arreglarlo. Incluso aunque propongas soluciones equivocadas, la respuesta que recibas será una lección valiosa en tu experiencia. Limpiar y repetir.

Creo firmemente en la necesidad constante de hacer mejores herramientas. Parece que no hay una escasez de contribuyentes al kernel, pero siento que muchas de las herramientas alrededor del kernel (especialmente herramientas como Sparse) podrían ayudar mucho más. Texto sin traducir: Compiler/Toolchain people takes a certain rare mindset though it seems. (Si tienes una opción de traducción ponlo en un comentario)

Otra área donde siempre se puede ayudar es testeando. Añadiendo nuevos test a los ya sugeridos, como xfstests, ltp, etc., podría ser una contribución muy útil de la que muchos se beneficiarían.

### **¿Qué escuchas mientras programas?**

Varias cosas dependiendo del estado de ánimo <http://www.lastfm.es/user/kernelslacker>

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Trato últimamente de reducir los viajes. La conferencia Linux Plumbers Conference y los eventos de North American Fedora son todo lo que hago por ahora.

## 7.- Paul Mundt

### Nombre:

Paul Mundt

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Primordialmente cuido de la arquitectura SuperH (y por proxy, parte esencial de plataformas para móviles ARM-based SH/R), pero esto lleva a tener que ocuparse de otras cosas. En el pasado era sobre todo un esfuerzo por mantener diferenciado el código genérico para no romper mis plataformas (un esfuerzo que persiste hoy día), pero como el código embebido (N.d.T: o incrustado) está cayendo en desuso convirtiéndose en un ciudadano de segunda clase en el kernel, es más fácil centrar el tiempo en la reutilización de las infraestructuras existentes para uso de casos adicionales.

Además del papel de mantenedor de mis arquitecturas, mi tiempo principalmente lo paso entre tareas como la gestión de memoria (especialmente MMU-less systems, soporte NUMA, etc), funciones de ahorro de tiempo del reloj, y gestión de las interrupciones IRQ. Últimamente he estado trabajando en hacer más útiles los dominios de IRQ (irqdomains) (particularmente para los non-DT) y extendiendo los eventos del reloj para utilizar los canales no usados del temporizado de una manera más efectiva.



En el pasado también me encargué del subsistema framebuffer durante un tiempo, pero ese trabajo ha cambiado de manos, y me era imposible dedicarle el tiempo que requería para mantenerlo.

### ¿De dónde recibes tu nómina?

Renesas Electronics

### ¿En qué parte del mundo vives, y porque allí?

En Tokyo, Japón. Me desilusioné con Silicon Valley hace ya tiempo y no veo una razón para tener que regresar. Después de probar en varios países durante un tiempo, Japón se ha convertido en mi hogar desde hace más de 6 años, y me va muy bien. El hecho de que la mayor parte del país sea montañoso también me ofrece suficientes vías para mantenerme ocupado fuera del trabajo.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Supongo que sería una combinación de fbcon , vim y mutt. Es de la manera que he trabajado desde siempre. (aunque sin fbcon en los primeros días del kernel 2.1.) Nunca he sido capaz de trabajar en serio con un entorno de escritorio, así que los evito todo lo que puedo. Dicho esto, en las ocasiones en las que por ejemplo necesito navegar por algún sitio no soportado por lynx , o leer algún documento presentado en un formato sin sentido, o dándome cuenta de que paso una gran cantidad de tiempo con entradas en japonés, a regañadientes utilizo fvwm2. El formato de texto siempre ha sido mi método de trabajo preferido.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Llegué a Linux un poco tarde, alrededor de 1996 más o menos. Empecé de manera muy general, una pila de las cartas random ISA que era soportada, no funcionaba. El primer proyecto fue conseguir hacer funcionar DMA en un 3c501, o algo parecido, eso en un kernel 2.0. Fue hace mucho tiempo y no recuerdo muy bien los detalles, pero no fue fácil. Cuando surgió el subsistema framebuffer en los últimos kernels 2.1.x empecé a pasar mi tiempo y gradualmente pasó a ser una arquitectura funcional. (MIPS al principio, y después SuperH vía ingeniería inversa de Dreamcast)

### **¿Qué es lo que hace que sigas interesado en esto?**

La evolución constante. Evolucionando código existente para que soporte nuevos requerimientos mientras debe coexistir con lo antiguo. Ser capaz de revisar código que escribiste hace décadas para tratar un problema particular después de encontrar que de pronto tienes una manera de resolverlo de una manera más simple y eficiente. (eso asumiendo que te acuerdas de lo que estabas pensando), etc.

Es también interesante ver como la gente utiliza de diversas maneras el kernel, en formas que tu antes no habías considerado, particularmente te ofrece un punto de vista totalmente diferente de una infraestructura ya existente.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Encontrarás que hay un montón de gente interesada en algún aspecto concreto de un área que puede llegar a ser muy territoriales, a los que a menudo puedes manipular para tu propio beneficio: mediante la publicación de un parche que soluciona un problema particular, y mientras al mismo tiempo ofender su sensibilidad suficientemente como para que está impulsado en forma inmediata a la acción para resolverlo de una manera mucho más eficaz para ti.

Estaba trabajando en un problema en particular, cuando llegué a una limitación de la API de mapas de bits, donde mi tamaño deseado de mapa de bits excedí el número de bits, un caso que había sido diseñado para provocar un error con una nota muy útil que todo el que llegara por primera vez llegara lo codificara.

Yo estaba trabajando en un problema particular, cuando llegué a una limitación de la API de mapas de bits, donde mi tamaño deseado de mapa de bits superado el número de bits, un caso que había sido diseñado para provocar un error con una nota muy útil para ver quien era el primero que conseguía escribir el código. Me puse a hacer frente a esto con un algoritmo de crudo para el seguimiento de estos errores (después llamado la extensión Mundt multipalabra), durante un vuelo de larga distancia desde Seúl. El acceso a Internet era irregular en el mejor de los casos, pasé más tiempo durmiendo y cuidando a mi bebida que monitoreando activamente el tráfico de la lista, pero en el momento en que aterricé el código ya había sido reescrito, optimizado, y enviados a Linus para la fusión con el kernel.

### **¿Cuál sería tu consejo para los desarrolladores que quieran implicarse?**

No se quede atrapado en los jardines amurallados.

El núcleo y las personas que trabajan en él han resistido muchos vendedores con sus propias agendas, y seguirá haciéndolo en el futuro previsible. A menos que sobre todo quiera trabajar para un proveedor determinado, no se distraiga en el corto plazo y deje atrapar en un mismo proveedor por ser más fácil en términos de disponibilidad de hardware (esto también se aplica a los foros de la industria que alegan tener un interés por resolver los problemas de forma genérica, sin tratar de colaborar aguas arriba durante el desarrollo). Como muchas empresas tienen intereses superpuestos

para el kernel, los hackers del kernel gozan de un cierto nivel de autonomía – algo que no vale la pena ceder para lo que será otra de una larga serie de iniciativas abandonadas en unos pocos años.

Mientras que la barrera de entrada para apoyar el nuevo hardware puede ser muy alto, es bastante sencillo de encontrar un área que te interesa y encontrar lo que te interesa cambiar. En última instancia todo se reduce a curiosidad, que es algo que es poco probable que experimente con el que hacer trivial o mecánico. Usted tendrá más que suficiente ayuda en el camino, siempre y cuando esté dispuesto a poner el esfuerzo y tratando de hacer cambios significativos. Más allá de eso, todo el mundo es rechazado de vez en cuando, independientemente de que ha estado trabajando en el núcleo durante 15 minutos o 15 años.

Ningún hacker del kernel que se me ocurra ha comenzado realizando cambios de espacios en blanco o de ortografía, y parece poco probable que esto cambie. Si usted está tratando de hacerse un nombre en la comunidad del kernel, lo ideal queremos que sea un hecho positivo y no peyorativo.

### **¿Qué escuchas mientras programas?**

Eso varía. Generalmente prefiero algo tranquilo, pero en un entorno de oficina esto no es siempre posible, así que cualquier cosa que haga de filtro del ruido de fondo está bien.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Para la lista de correo lo normal vger, linux-kernel, linux-arch y para comentarios pertenecientes a mi arquitectura linux-sh.

Trato de evitar todo menos la cumbre del kernel, pero normalmente doy una charla una vez al año o así dependiendo de cómo vaya en lo que estoy trabajando. Si hago alguna conferencia, normalmente me salto las charlas y las sigo con discusiones por los pasillos.

## 8.- Alan Cox

### Nombre:

estooooo paso... ¿son todas las preguntas así de difíciles?

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

He trabajado en partes grandes del kernel en distintos momentos. En este momento estoy tratando de finalizar el remodelado de la capa terminal, tratando de resolver errores en esta área. Es básicamente un trabajo desagradable, horrible técnicamente que no es excitante o no hay mucha más gente que se preocupe de ello.



Otra cosa en la que ando actualmente es tratando de conseguir que sea estable es soporte para gráficos 2D para varios basados en Imagination, o dispositivos basados en gráficos creados por Intel, y la misión de un hombre de limpiar la colección de errores bugzilla del kernel

### ¿De dónde recibes tu nómina?

Después de un período de diez años en Red Hat ahora me he cambiado a Intel hace un par de años, y me estoy divirtiendo por aquí.

### ¿En qué parte del mundo vives, y porque allí?

Swansea, al Sur de Gales. La primera vez que acabé por aquí fue cuando fui a la universidad. Encontré a mi mujer aquí y aquí hemos estado desde entonces. Esto lo encuentro mucho más relajante y humano que las grandes ciudades. Es bonito poder mirar por la ventana y contemplar las colinas y el mar.

También tenemos el mejor equipo de rugby aquí que el de Inglaterra

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

El cerebro, la experiencia y a veces una pizca de suerte. Mis herramientas técnicas son las más tradicionales, pero con una con un mejor gusto en editores. Así utilizo bash shell, editor joe, gcc, make, git

La mayoría de mis PC's utilizan Fedora, pero mi escritorio varía según el sistema y la configuración y también al azar, cuando tengo ganas de revolver las cosas y de cambiar o intentar cosas. Ahora mismo estoy jugando con algunas de las características de GNOME 3 en parte debido a que en Fedora 17 están apareciendo errores en mi driver de gráficos. Mi otra elección habitual suele ser Xfce.

Utilizo una máquina virtual con Ubuntu corriendo con Unity para incluir mis trabajos en Android,

que parece un pobre “remake” de Windowmaker pero sin su estilo

También estoy experimentando con otras distribuciones de GNU/Linux ya que encuentro a Fedora (aunque es técnicamente muy buena y con un montón de cosas de vanguardia bien integradas) cada vez con más fallos.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Principalmente por accidente. Había estado añadiendo facilidades a Unix para incluir email e identificaciones de usuarios a mi Amiga, pero realmente necesitaba un mejor sistema operativo para hacer correr AberMUD. Windows por aquel entonces era un chiste el utilizarlo. 386BSD necesitaba FPU y algo más, Linux simplemente funcionaba.

Habiendo descubierto que Linux simplemente funcionaba, pronto descubrí que no funcionaba del todo bien, y terminó teniendo varios errores, por aquel entonces la pila de red. Con el tiempo terminé manteniendolo junto con los lanzamientos estables. Durante ese tiempo en algún momento coincidí trabajando con Thomas Radke para producir el soporte Linux SMP (Thomas desafortunadamente siempre parece olvidarlo) y tuve un par de incursiones en otros proyectos: Linux Mac68K y mini Linux en 8086.

### **¿Qué es lo que hace que sigas interesada en esto?**

En parte el hecho de que está siempre en constante cambio, cambia el software, cambian las necesidades. Así que siempre te enfrentas a nuevos retos.

El otro aspecto de esto es que es capaz de cambiar parte del mundo. Directamente, pone a los ordenadores en lugares en los que nunca se podrían pagar licencias propietarias. Indirectamente, forma parte de la primera ola de todas las reclamaciones de cultura y producción para el pueblo. Esto es algo que los que hacen cultura están extendiendo en el mundo físico.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

No creo que pudiera elegir un momento en concreto. Estos tienden a ser breves pero en mucha cantidad. No es esto un sitio para aburrirse. Estoy comprobando las cifras de Android con asombro. A este ritmo Linux podría superar en número de copias a los Beatles dentro de poco.

### **¿Cuál sería tu consejo para los desarrolladores que quieran implicarse?**

Nunca admitas que no sabes nada sobre bases de datos, y nunca te ofrezcas para ayudar a configurar bugzilla.

Para cualquier proyecto, hacer cosas que te importen y hacerlas bien. Hacerlo bien es un término ambiguo pero si eres feliz con una parte de código y eso lo haces de la manera que crees, debería salir bien, entonces eso es probablemente “hacerlo bien.” Cinco años después puedes echarle un vistazo y echarle a temblar, pero eso es el aprendizaje.

Tampoco te centres en el kernel porque es famoso y algo grande. En muchos aspectos eso lo hace que sea difícil trabajar en ello. Si rompes el kernel hay un montón de gente que se preocupará (y muchos más que se añaden sólo por Android.)

Hay mucha más apertura y flexibilidad en muchos de los proyectos y tecnologías actuales. Gente haciendo cosas asombrosas con impresoras 3D, HTML5 y WebGL están probablemente más cerca de la naturaleza del Linux de los primeros días cuando cualquier cosa era posible y no había nada

demasiado loco para intentar porque nadie sabía que funcionaría.

### **¿Qué escuchas mientras programas?**

Si estoy trabajando con código complicado, nada. Para el trabajo duro depende de cómo sea de tediosa la tarea. Una amplia gama de bandas de folk/rock desde Show of Hands hasta ZZ Top o New Model Army. Para trabajo tedioso y repetitivo de código no hay nada como Mötörhead o quizás otro poco de Ministry o Dead Kennedys.

También otros menos conocidos y apartados de los más conocidos en la música como The Fishermans Friends, Machinae Supremacy y Hörstreich. Está bien ver a los músicos finalmente liberarse de las cadenas de la industria de la música.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Un buen número de listas relacionadas con el kernel, pero no muchas conferencias. No soy muy fan de los largos viajes y aborrezco y detesto volar. Ya estaba bastante mal antes de todo este teatro de la seguridad, ahora es simplemente horroroso.

Sin embargo tuve la idea de hacer de Swansea a Praga en tren, es probablemente más probable que me encuentres en una exposición de maquetas de trenes o el patrimonio ferroviario que en una conferencia de Linux. Linux puede ser divertido, pero aún así es “trabajo”.

## 9.- Arnd Bergmann

**Nombre:**

Arnd Bergmann

**¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?**

Soy co-mantenedor junto con Olof Johansson de la rama arm-soc del kernel. Recogemos los parches que nos mandan docenas de mantenedores de subarquitecturas de ARM y consolidamos los parches unidos para enviárselos a Linus Torvalds. En la actualidad esto supone alrededor de 1000 parches en cada petición de envío.

También reviso todas las nuevas arquitecturas de puertos que se incluyen en Linux, y también vigilo los directorios de el kernel drivers/char, drivers/misc e include/asm-generic.



**¿De dónde recibes tu nómina?**

Trabajo para IBM investigación y desarrollo en Böblingen, que me permite trabajar a tiempo completo en el proyecto sin ánimo de lucro Linaro a través de un acuerdo con el Centro de Tecnología Linux de IBM e IBM Microelectrónica.

**¿En qué parte del mundo vives, y porque allí?**

Me mudé a suroeste de Alemania por mi trabajo en IBM hará poco más de 10 años, este fue uno de los pocos lugares en mi país de origen que me permitía trabajar en el kernel de Linux

**¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Realizo casi todo mi trabajo usando git y vim. También utilizo una estación de trabajo rápida X86 para realizar kernels ARM multi-estructuras. Utilizo Kubuntu en mi PC de escritorio, pero he estado pensando cambiarme a XFCE4 una vez que encuentre una forma de migrar las configuraciones de mi correo de kmail 1.12.

**¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Empecé como un simple usuario de Linux con un montón de hardware poco común en los últimos años de la década de los 90, en los que se requería hacer modificaciones en el kernel para que funcionara. Estuve mandando sobre todo soluciones a errores durante mi tiempo de universitario, hasta que aprendí que podrían pagarme haciendo las mismas cosas que antes hacía por afición.

**¿Qué es lo que hace que sigas interesada en esto?**

No me puedo imaginar haciendo otra cosa después de haber estado revisando alguna vez la mayoría de subsistemas del kernel durante los últimos 15 años.

Una de las mejores cosas de mi trabajo es que todas las cosas buenas de mi trabajo permanecerán ahí, incluso si yo cambio de trabajo o incluso algún proyecto se viene abajo.

**¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo**



**colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Una vez revisando un nuevo código que fue propuesto para ser incluido para una nueva arquitectura . Le dije al autor que aquel código necesitaba un montón de ciclos de revisiones durante meses e hice un listado de cientos de cosas que estaban mal. Una semana después el desarrollador regresó con una nueva versión que resolvía todos los errores listados y el código fue aceptado en la siguiente petición de envío de código al kernel.

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Leer un montón de código y revisiones de código de otra gente que este haciendo un buen trabajo. Cuando escribas tu propio código o revises parches piensa cómo lo harían ellos.

Como primera contribución, empieza con algo útil y pequeño, como un error obvio. Se pueden encontrar un buen montón de errores usando las herramientas automatizadas que se listan en la documentación del kernel.

**¿Qué escuchas mientras programas?**

Suelo escuchar last.fm pero no funciona bien con los cortafuegos de las empresas, así que actualmente utilizo radioparadise.com que reproduzco en squeezebox.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Asisto a los tres o cuatro eventos de Linaro Connect cada año, y también LinuxCon y ELC cuando tengo tiempo. Mis canales IRC por el momento son #linaro-kernel y #armlinux en freenode.net

## 10.- John W. Linville

### Nombre:

John W. Linville



### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Desde el 2006 soy el mantenedor del subsistema de redes inalámbricas dentro del kernel Linux. Me gustaría que esto significara que he escrito un montón de nuevo código, pero lo que significa en gran parte es que reviso código, recopilo y unifico parches, etc. Creo que esto me ha permitido ser un “jugador más neutral” en términos de equilibrar la influencia de los diversos colaboradores. Generalmente me mantengo alejado de los desarrolladores principales de las redes inalámbricas para mantener buenas relaciones.

Recientemente, los desarrolladores trabajan en Bluetooth y Comunicaciones de campo cercano (Near Field Communications, NFC) también han estado mandando sus parches a través de mi rama del kernel. No soy un experto en esas tecnologías. Pero creo que el papel que juego ayuda a liberar de parte de trabajo a Dave Miller para que el código sea más fluido cuando se le manda a Linus.

### ¿De dónde recibes tu nómina?

Soy empleado de Red Hat inc.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en una zona un poco rural, a unos 45 minutos al oeste de Raleigh, Carolina del Norte. La zona en la que crecí era similar a esta, y creo que este es un buen lugar para criar una familia. Suelo trabajar fuera de mi casa, pero a menudo voy a la oficina de Red Hat en Raleigh con el fin de conseguir ver cara a cara a algunos de mis compañeros de trabajo.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Yo soy un simple hombre de línea de comandos. La pantalla de mi escritorio suele estar lleno de ventanas de terminal, y me siento feliz usando Bash, sed y awk cuando necesito de automatización.

En cuanto a los equipos de sobremesa ... Yo era un usuario de KDE 3.x, pero nunca me adapté a KDE 4.x Entonces fui usuario de GNOME 2, pero... bueno, ahora soy un refugiado de nuevo.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Hace casi 20 años, que tenía que hacer algo de ingenierías para conseguir que mi tarjeta de red para funcionase con cualquier kernel que estaba vigente por entonces. Pero, no me tomé en serio el desarrollo de Linux hasta finales de 1998. En ese momento yo estaba tratando de utilizar Linux como sistema operativo para una plataforma de demostración de hardware que mis jefes proveían a sus fabricantes. El hardware de demostración era un procesador de red, y yo estaba usando Linux para proporcionar la pila de red para hacer que el procesador de red hiciera algo que valiera la pena.

Por supuesto, eso fue antes de que Linux dominara el mercado de sistemas embebidos. Así que cuando conseguí que funcionara, la gestión de esa empresa me dio una palmadita en la cabeza y luego me preguntó si yo podría hacer todo el trabajo en la parte de VxWorks. Afortunadamente,

otros empresarios más tarde se interesaron más en mis conocimientos de Linux!

**¿Qué es lo que hace que sigas interesada en esto?**

Linux y el código abierto creo que es una buena baza para mí. He usado otros sistemas basados en Unix cuando estaba en la universidad, y siempre he encontrado la “filosofía Unix” una forma atractiva para mí hacer un buen uso de las computadoras. Así que Linux es la natural y moderna elección de continuar esa tradición.

Más allá de mi faceta de ingeniero, quiero ser capaz de investigar “debajo del caparazón” para arreglar ,o mejorar aquello que desee. El kernel de Linux y los otros bits de código abierto de una distribución Linux satisfacen mi necesidad de ser capaz de hacer que el equipo se ajuste a mis necesidades, en lugar de tener que adaptar mis necesidades a lo que alguien ofrece.

**¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

No soy muy bueno “llevando la cuenta” de estas cosas. Pero, por lo general me río bastante cuando alguien decide que ha encontrado el único y verdadero camino para hacer algo y luego tratan de convencer a los demás en la comunidad que todos tenemos que cumplir con sus demandas. Probablemente el mejor ejemplo que me viene a la mente sobre esto es de los debates recurrentes sobre lo que permite código C ++ en el núcleo. C ++ es mejor que C, ¿no?

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Lo mejor es rascarse lo que a uno mismo le pica! Si has experimentado un problema (quizás un error, o simplemente un bajo rendimiento) entonces trabaja sobre ello y trata de arreglarlo. Deberás leer un montón de código, intercambiar algunos mails, estudiar y buscar información, y hacer mucho trabajo. Pero aprenderás un montón, y probablemente también será divertido!

**¿Qué escuchas mientras programas?**

Soy aburrido. Escucho las emisoras de noticias y entrevistas por la radio.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Dirijo las siguientes listas de correo: linux-wireless@vger.kernel.org, netdev@vger.kernel.org y linux-kernel@vger.kernel.org y también algunas otras. Tampoco soy muy difícil de encontrar si alguien quiere mandarme un correo electrónico privado. Y en canales IRC, mi nick es “linville” en freenode, OFTC ay algunas otras redes. En cuanto a conferencias he ido a LinuxCon North America y la Linux Plumber’s Conference en años anteriores. Trataré de ir a LinuxCon Europe o algún otro evento de la Linux Foudation tan a menudo como pueda encontrar a alguien que pague mis billetes. También he estado en FUDCon y en algún otro evento en el pasado.

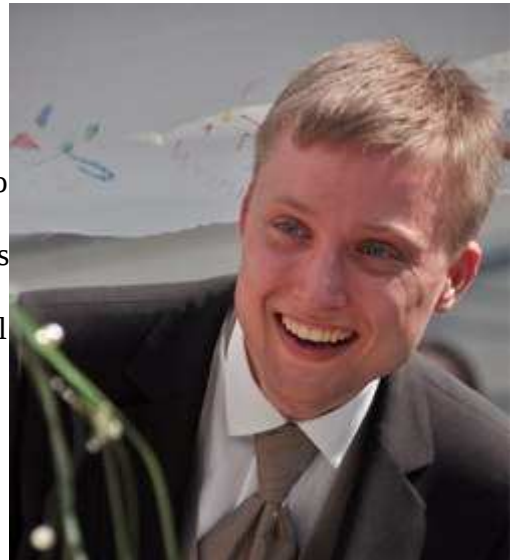
## 11.- Johannes Berg

**Nombre:**

Johannes Berg

**¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?**

La mayor parte de mi “carrera en Linux” la he pasado en el desarrollo de las redes inalámbricas (802.11) y mantengo la mayoría del código genérico del kernel de redes inalámbricas, y además el driver iwlmwifi. A veces hago otros trabajos relacionados en otros subsistemas que interactúan con las redes inalámbricas o cuando algo me molesta. En el pasado también he hecho algo en lo referente a la alimentación eléctrica de los ordenadores (simplemente como afición), por ejemplo implementé la hibernación en 64-bit. (Pero no me preguntes si esto todavía funciona!)



**¿De dónde recibes tu nómina?**

Intel

**¿En qué parte del mundo vives, y porque allí?**

En Detmold, Alemania. Crecí por aquí y me gusta esta zona, pero la razón más pragmática que puedo elegir es que mi mujer trabaja aquí.

**¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Voy a mantener esta respuesta sobre el escritorio en secreto, para no ser tratado como un loco por el resto de la comunidad en torno al kernel, pero lo que más utilizo es un navegador web, un cliente de correo, y un montón de terminales para todo lo demás. Y para herramientas de producción, no puedo vivir sin mi editor favorito “joe” y no utilizo mucho más que las herramientas normales para el desarrollo del kernel de linux: el editor, git, sparse, smatch and spatch; para mi trabajo con las redes inalámbricas necesito wireshark; y en los últimos años me ha encantado trace-cmd, porque permite a los usuarios recolectar un montón de fallos a pie de campo que después, más tarde puedo analizar estando desconectado.

**¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Es una larga historia que se puede resumir en que tenía hardware que no funcionaba. En mi caso particular era una tarjeta inalámbrica Broadcom en un modelo 2005-model PowerBook.

**¿Qué es lo que hace que sigas interesada en esto?**

Hay un nuevo reto cada día y las soluciones no siempre son obvias, lo que satisface mi lado creativo. Además, me encanta hacer un trabajo que en realidad afecta a mucha gente. Lo cierto es que, he oído más sobre ello cuando hay un impacto negativo, pero estoy seguro de que no puede ser del todo malo.

**¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Alguna de las bromas de parches enviado el día del April Fools', he visto algunos muy ingeniosos!

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Tratar de seguir los asuntos por un tiempo y tratar de tener una idea de cómo las interactúan las personas unas con otras. Si alguien te grita, pero todos los demás parecen ignorarlo, entonces quizás también debieras ignorarlos. Pero si es el mantenedor del subsistema del que dependes quizás no debieras hacerlo! Hay una estructura social en la comunidad que puede ser difícil de adivinar, pero si lo intentas probablemente seras más feliz.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

He dejado de suscribirme a muchas listas de correo, así que sólo sigo de cerca la lista de wireless del kernel. Hay un canal llamado #linux-wireless por el que me conecto, y otros que no sigo. Las conferencias son más o menos aleatorias. Estaré hablando en LinuxCon North America, pero me tendré que perder LinuxCon Europe a pesar de que este año iba a haber una cumbre sobre redes inalámbricas.

## 12.- Martin K. Petersen

### Nombre:

Martin K. Petersen

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Principalmente trabajo añadiendo soporte para nuevas características de almacenamiento, así que mis áreas de trabajo son los bloques y las capas SCSI, y de vez en cuando un poco de ATA.



### ¿De dónde recibes tu nómina?

Trabajo in Oracle's Linux organization

### ¿En qué parte del mundo vives, y porque allí?

Crecí en Dinamarca, pero actualmente vivo en Montreal, Quebec. Me mudé a Canadá para unirme a Linux en la época del punto-com

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

En mi PC de escritorio tengo Fedora, pero es esencialmente sólo una terminal X. Todo el trabajo real lo hago con un puñado de máquinas remotas. Las herramientas que utilizo raramente cambian: xterm, emacs, perl, midnight commander y git. Hace tiempo que perdí la paciencia usando entornos gráficos de escritorio y ahora utilizo el gestor de ventanas i3 porque mepermite organizarme las terminales como más me gusta.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Yo era un estudiante allá por los primeros años de los 90. Las terminales y las estaciones de trabajo eran escasas en la universidad, así que ser capaz de hacer funcionar algo parecido a Unix en mi casa era muy conveniente. Linux era nuevo y excitante y la elección obvia. Empecé toqueteando en el kernel y el servidor X para que funcionase bien con el hardware que tenía por entonces en mi PC. Desde entonces he experimentado con plataformas más exóticas como PA-RISC o Itanium.

### ¿Qué es lo que hace que sigas interesada en esto?

Me gusta hacer que funcione el hardware; eso es lo que en realidad me motiva. Además de la adrenalina de finalmente encontrar un bug que he estado persiguiendo durante semanas.

### ¿Cual sería tu consejo para los desarrolladores que quieran implicarse?

Encuentra algo que esté roto y arréglalo. Errores actuales, no errores de tipografía o formato. Al principio puede parecer una locura, tener que invertir horas/días/semanas en arreglar algo. Pero es el proceso para comprender plenamente el problema de que es la parte más importante, sin abrir inmediatamente un editor para hacer el cambio de código. Leer el código que otros han escrito puede ser duro al principio, pero es algo fundamental para trabajar en un proyecto comunitario. Puedes estudiar cómo ser un desarrollador del kernel, pero la única manera de aprender a fondo es poniéndose manos a la obra.

### ¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?

Linux-scsi. No me conecto a los canales IRC más, demasiada distracción. Trato de ir a una o dos conferencias de Linux al año aparte de la Linux Storage & Filesystem Summit.

## 13.- Julia Lawall

### Nombre:

Julia Lawall

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Desarrollo la adecuación y transformación de la herramienta Coccinelle. Coccinelle puede ser aplicado a cualquier código en “C”, pero yo principalmente lo aplico al Kernel de Linux.

### ¿De dónde recibes tu nómina?

Inria. Dentro de Inria, participo en IRILL, un centro de investigación de software libre y código abierto.

### ¿En qué parte del mundo vives, y porque allí?

París. Una ciudad muy bella y muy habitable...

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Emacs, ocaml, xfce.



### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Mi formación es en el análisis del programa, la transformación de los programas y la programación funcional, pero siempre estuve interesada en la comprensión de cómo funcionan los sistemas de computación en todos los niveles. Un colega me sugirió sobre investigar el problema de portar los controladores de dispositivos del núcleo Linux 2.4 para Linux 2.6. Estudié un montón de cambios que se hicieron en la serie 2.5, y hemos diseñado Coccinelle de acuerdo a las necesidades de los tipos de cambios que observé.

### ¿Qué es lo que hace que sigas interesada en esto?

La variedad aparentemente infinita de cosas que pueden estar mal en el código y la alta calidad de la retroalimentación de la comunidad Linux.

### ¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?

Una combinación divertida de un comentario y un poco de código es el siguiente:

```
/* Don't leak any random bits. */  
memset(elfregs, 0, sizeof(elfregs));
```

(N.T: Quien lo entienda que me lo explique)

### ¿Cual sería tu consejo para los desarrolladores que quieran implicarse?

Utilizar las herramientas para encontrar algunos errores simples. O ver los problemas que otros han resuelto, y tratar de encontrar nuevas ideas o soluciones. Cuando encuentras algún error, mira alrededor en el código de cercano. A menudo se pueden encontrar otras cosas interesantes.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

En la lista de correo de Coccinelle. En la lista de correo del Kernel. Principalmente asisto a conferencias académicas sobre sistemas operativos, lenguajes de programación e ingeniería de software. También he asistido a Linux Plumbers Conference alguna vez y he aprendido un montón.





## 14.- Ben Hutchings

### Nombre:

Ben Hutchings

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

En mi tiempo, soy uno de los mantenedores de kernel de Linux en Debian, y mantengo la serie estable de Linux 3.2 que será usada en Debian 7.0. Trato de resolver fallos importantes en todo el kernel que se envía y se revisa para incluirlo en 3.2.y en otras series estables. También soy co-mantenedor de los repositorios de linux-firmware, que recopilan varias funciones que los drivers de Linux podrían necesitar cargar con sus dispositivos.

En mi día a día en el trabajo, cuido del driver sfc network y ethtool, con algunas incursiones en el núcleo de redes y otros subsistemas de los que depende sfc.



### ¿De dónde recibes tu nómina?

Solarflare.

### ¿En qué parte del mundo vives, y porque allí?

Cambridge, Inglaterra. Es donde encontré mi primer trabajoy todavía no he encontrado una razón para tener que dejarla durante mucho tiempo.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Emacs, Evolution, GNOME 3, Iceweasel (aka Firefox).

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Fue una combinación de varias cosas que ocurrieron entre 2007-2008: Un cambio gradual en mi trabajo desde desarrollador de pruebas a mantenimiento de código de producción; tratando de mantener los paquetes Debian de los primeros drivers wifi Ralink; y queriendo resolver los problemas de larga duración (para Debian) del código no libre embebido en el kernel.

### ¿Qué es lo que hace que sigas interesada en esto?

Siempre hay oportunidades de hacerlo mejor en términos de rendimiento, seguridad y usabilidad. Hay un flujo constante de nuevos equipos a los que dar soporte, con nuevas características y nuevos retos. Y, por supuesto, siempre hay errores que resolver (algunos de estos problemas son más interesantes que otros).

### ¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?

No estoy seguro de poder elegir un solo ejemplo, pero esta fue una presentación bastante tonta:  
- <http://lwn.net/Articles/490040/>

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Empiece por rascarse un picor. Entonces si quieres, trabaja tu manera de hacer cirugía cerebral. Lee documentación y registros de cambios.

**¿Qué escuchas mientras programas?**

Una amplia variedad de música, alguna la puedes encontrar en esta lista:

<http://www.last.fm/user/womble2/tracks>

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Estoy suscrito a las listas de correo de netdev y stable y también estoy en varios canales IRC. No hay una conferencia a la que asista todos los años, pero he estado en DebConf, FOSDEM, Chaos Communication Congress, Linux Plumbers Conference varias veces.

## 15.- Mauro Carvalho Chehab

### Nombre:

Mauro Carvalho Chehab. Nick: mchehab

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Soy el mantenedor del subsistema de medios en el Kernel. Esto incluye drivers para mandos a distancia, webcams, captura en directo de audio y video, televisión analógica y digital, receptores y transmisores de AM/FM.

Soy el principal contribuidor del subsistema EDAC. Y como tal estoy trabajando actualmente en una mejor integración de varios subsistemas relacionados con el reporte de errores hardware (EDAC, APEI GHES, MCE) con el fin de trabajar juntos.



También trabajo en lo relacionado con herramientas como xawtv, v4l-utils, edac-utils y algunas otras.

### ¿De dónde recibes tu nómina?

En agosto, celebré cuatro años en Red Hat. Antes de eso, solía mantener el subsistema multimedia en mi tiempo libre, mientras trabajaba en algo que no tenía nada que ver (mi anterior cargo fue para planear e implementar los sistemas de telecomunicaciones y la red de gestión de telecomunicaciones – trabajé como tal en una gran empresa Brasileña)

### ¿En qué parte del mundo vives, y porque allí?

Ahora estoy viviendo en la parte suroeste de Brasil. Antes de eso estuve viviendo en Brasilia. Mi mujer, mis hijos y yo solíamos padecer bastante el clima seco y desértico de Brasilia, así que migramos a una bonita ciudad de la zona interior del estado de São Paulo, en el valle del río Paraíba, una ciudad con buenas infraestructuras, y buenas escuelas para los niños.

Ahora estamos cerca de una gran ciudad como São Paulo (pero no muy cerca como para sufrir los problemas normales de una gran ciudad) así como a algunos lugares agradables en la montaña y en la playa.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Utilizo las herramientas normales: git, gcc, gmake, bash, perl un cliente de correo y un navegador web.

No me preocupa mucho sobre el cliente de correo, así que utilizo Thunderbird la mayor parte del tiempo. Aunque también utilizo pine, claws-mail y/o evolution depende de mi estado de ánimo. Como todas las carpetas están instaladas en un servidor imap local, no importa mucho que cliente de correo utilice.

En general prefiero utilizar herramientas basadas en modo texto. Pero no estoy en contra de las ayudas de las interfaces gráficas. Para la edición de archivos utilizo nano la mayor parte del tiempo. Así cuando necesito realizar alguna tarea compleja Kate es mi opción.

He escrito varios scripts en bash y perl que me ayudan en el manejo de parches.

La distro que utilizo actualmente es Fedora, porque me ofrece todo lo que necesito. Generalmente instalo la última versión estable., me encanta ver las mejoras en mi escritorio. Aunque, no me gusta mucho cuando las características que utilizo en algunas interfaces gráficas se eliminan en una nueva versión de la misma aplicación.

Respecto a las interfaces gráficas, utilizo 3 monitores, ya que mantengo muchas ventanas abierta al mismo tiempo. Desafortunadamente muchas interfaces gráficas no soportan bien más de un monitor. Así que siempre estoy intentando una nueva para ver si esta funciona bien con mi entorno de escritorio (KDE 3 y Gnome 2 es lo que utilizo para trabajar, pero ya no se encuentran en las modernas distros.)

Así que voy cambiando ente xfce, lxce, KDE 4 y GNOME 3. Desde Fedora 17, Gnome 3 casi funciona con varias extensiones habilitadas. Así que actualmente estoy tratando de vivir en paz con Gnome 3 ya que, incluso corriendo xfce o lxce, la mayoría de las aplicaciones de GNOME 3 se carga núcleo de todos modos. Así que los ahorros de memoria con xfce o lxce generalmente no se aprovechan, y algunos pequeños problemas pueden surgir por no tener la shell de GNOME 3 para gestionarlos.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Tengo una larga historia con Linux. La primera versión de Linux que utilicé fue Slackware 0,97 con Kernel 1.0, que instalé para trabajar como un router de Internet (tenía la suerte en ese momento de tener la asombrosa cantidad de 9.600 Kbps de enlace dedicado a Internet en mi escritorio y quería compartirlo con mis amigos en la oficina). Traté primero en utilizar un router TCP / IP de código abierto, llamado KA9Q, pero no era estable en mi sistema, por lo que un amigo me sugirió Linux en su lugar.

Desde entonces, he instalado Linux en mi PC de escritorio y utilizado Linux y en varios lugares con el fin de administrar las oficinas centrales de Plain Old Telephony Service. En ese tiempo usar código abierto y hacer un desarrollo interno dentro de la compañía de telecomunicaciones era considerado una blasfemia, incluso cuando estaba ahorrando diez millones de dólares al año con una solución basada en Linux. Por lo tanto, he experimentado una sensación creciente de que yo debería devolver parte a la comunidad debido a los grandes servicios prestados por el software de código abierto en esa época.

Mis comienzos como estudioso del Kernel fue por accidente. En 2005 estaba pensando en trabajar un título de doctorado de ingeniería. Como las tesis de mi graduación y las 2 anteriores se hicieron en el área de proceso de imágenes, mi idea era trabajar escribiendo algún códec de vídeo. Así que compré una tarjeta capturadora de TV que supuestamente funcionaba en Linux.

Desafortunadamente, el formato estándar utilizado en Brasil (PAL/M) no estaba soportado. Sin embargo, todo para PAL / M parecía estar allí en el código del controlador. Sin embargo, los vídeos se veían en blanco y negro.

Así que decidí arreglarlo por mí mismo, y compartir los logros. Me llevó sobre dos semanas trabajando en ese parche, volviendo a leer viejos libros sobre TV analógica para ver los detalles de los estándares de vídeo y leer documentación sobre cómo programar el dispositivo. Al final del día, los colores no estaban funcionando debido a un desplazamiento de tiempo 0,1% en la parte del driver responsable de la programación del decodificador de vídeo de tiempo de muestreo. El resultado fue de parche de dos líneas de código

Por ese tiempo, Randy Dunlap y Andrew Morton me ayudaron mucho para aportar mi parche, dándome las instrucciones adecuadas de cómo compartirlo en la lista de correo de Linux Kernel. Hoy en día, cuando recibo un parche de un remitente nuevo, me acuerdo de la paciencia que tenía y trato de no ser gruñón con el remitente.

Después de la solución al PAL/M, encontré otros errores que me condujeron a reescribir el módulo dentro del subsistema y varias otras partes en el propio driver. También otros desarrolladores me preguntaron cómo contribuir con sus parches y más tarde me pidieron oficialmente que me encargara del mantenimiento. Lo que no sabía era que iba a significar eso en mi vida, era ingenuo y lo suficientemente valiente para hacerme cargo

Algún tiempo después, me dí cuenta que algo que empezó como una simple solución a un problema se convirtió en un trabajo serio. Así que fue el siguiente paso lógico para mí, dejar atrás mis 20 años de carrera para dedicarme por completo a Linux y al código abierto.

### **¿Qué es lo que hace que sigas interesada en esto?**

Cada nuevo driver que escribo es como un nuevo hijo: Verle dar sus primeros pasos, y llegar a ser útil para alguien más es muy excitante.

También me gusta arreglar diseños que no funcionan, limpiar código y ayudar a otros a contribuir. El gran logro del código abierto y Linux es debido a la colaboración, y me siento orgulloso de ayudar en este éxito y ayudar también a los usuarios a que tengan drivers para sus dispositivos en Linux.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

No hay un hecho aislado que resaltar. En general ocurre cuando veo nueva gente empezando con parches simples y de repente llegan a ser contribuidores activos.

También está muy bien cuando drivers fuera de la rama estable del kernel son incorporados a este. Tuvimos varios casos de esos dentro del subsistema de medios: gente que trabajaba por su cuenta sobre un proyecto finalmente de daba cuenta que el unir su código es bueno no sólo para ellos sino para todos aquellos que quieren utilizar su código. La fusión de estos códigos no es fácil, ya que los desarrolladores en general se quejan del estilo de codificación de Linux. Pero al final, los drivers tienden a ser mejores, a medida que más gente revisa el código y ofrece soluciones.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

No asustarse por aportar parches; si hay algo que no funciona o algo que mejorar por favor escribe un parche y compártelo.

Es muy probable que la gente haga malos comentarios sobre tu primer parche, diciendo que el parche no es correcto o que no sigue el estilo de código de Linux. Así que mi segundo consejo es (citando a Galaxy Quest): “Nunca abandones, nunca te rindas”. Vuelve a hacer el parche hasta que lo consigas. Puede que te lleve algunas intenciones hasta que lo consigas, pero aprenderás un montón durante el proceso.

Y mi tercer consejo: No te sientas intimidado por el lenguaje Inglés! Conozco mucha gente que se retrae cuando tiene que mandar sus parches porque no hablan inglés. Los parches son revisados por la calidad del código no por el vocabulario. De acuerdo que una buena descripción es una parte de

ello, pero si no dominas el lenguaje al 100%, eso es algo que tiene solución. Algunos de los que revisan, incluyéndome a mí, tenemos el hábito de arreglar los comentarios mal escritos cuando los parches no están adecuadamente descritos o cuando se advierte que el escritor del parche tiene problemas con el idioma.

Leer documentación en el directorio /Documentation antes de subir un parche ayuda bastante a realizar mejor el código.

El proyecto de Greg K-H : Linux Drive Project es un buen punto de partida para aquellos que quieren implicarse.

### **¿Qué escuchas mientras programas?**

El sonido del silencio Generalmente prefiero no ser molestado por ningún sonido mientras programo.

Cuando realizo un trabajo más rutinario, a veces escucho buena música. Soy muy ecléctico cuando escucho música: Me gusta la música clásica, rock, country, música internacional, ópera, ópera rock y por supuesto estilos brasileños (Samba, Bossa nova, Chorinho, old guard, new guard, Tropicalia, Sertaneja, rock de Brasilia, rock Brasileño, etc) también escucho nuevas voces como Adele.

Normalmente, al final de un día de trabajo, o cuando estoy muy estresado o si tengo que esperar una compilación que tarda bastante, tomo una guitarra (o un bajo) y trato de tocar una canción. Esa es una buena manera de deshacerse del estrés diario y darle a mi cerebro un poco de tiempo y oxígeno para pensar cómo resolver algún asunto complejo.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

A la lista de correo de “media subsystem” que es: [linux-media@vger.kernel.org](mailto:linux-media@vger.kernel.org). A la lista de correo de EDAC que es: [linux-edac@vger.kernel.org](mailto:linux-edac@vger.kernel.org). Normalmente estoy registrado en freenode en las salas de IRC: #v4l and #linuxtv. También estoy en OFTC en el servidor de IRC en las salas: #kernel-br (una iniciativa de la comunidad de desarrolladores del kernel en Brasil como una manera de ayudar a la gente que habla Portugués a empezar a conocer el kernel.)

En general no soy muy activo en el IRC, porque puede ser una gran distracción. Así que mantengo un bot siempre conectado y de vez en cuando me paso a echar un vistazo a los logs, a veces se discute sobre cómo tareas de desarrollar parches.

En cuanto a las conferencias, es más variable. Desde el año 2008, hacemos al menos una conferencia anual para el subsistema de medios de comunicación. En general suelo acudir. También suelo ir a la Cumbre del Kernel de Linux y sus conferencias.

## 16.- Jiří Slabý

### Nombre:

Mi nombre es Jiří Slabý. Pero apostaría a que a la gente le cuesta leerlo y mucho más pronunciarlo.

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Siempre hago lo que parece que debería hacerse. Antes hice este trabajo por dinero, me ofrecí voluntario para reescribir todos los drivers para evitar que quedaran obsoletos los dispositivos pci\_find.

Un tiempo después, reescribí y unifiqué los drivers para la tarjeta wifi Atheros 5k. Creo que la fusión y otras medidas tomadas por otros desarrolladores de drivers llevaron a Atheros a “abrir” su código. Pero había también una parte oscura: una discusión encendida que yo empecé.

Después reescribí la capa de entrada HID para convertirlo en bus. A pesar que muchos usuarios utilizan esta capa cada día, estoy contento porque nadie parece haber notado el cambio. Eso está bien porque los cambios que no se notan significan que no hay regresiones.

Ahora estoy de nuevo en las capas TTY. Junto con otros, estoy tratando de limpiar esas capas y eliminar y limpiar todo el código antiguo o inútil, que no cuenta con hotplug y otras cosas más modernas.

Al mismo tiempo, estoy tratando de encontrar la forma de cómo permitir a los parches del kernel servicios 24/7/365. Y como todos los demás también he hecho alguna cosilla más como escribir los límites de los procesos, y arreglar y mantener algunos otros drivers que reescribí, etc.

### ¿De dónde recibes tu nómina?

SUSE, desde hace más de cinco años. Siempre como empleado a media jornada.

### ¿En qué parte del mundo vives, y porque allí?

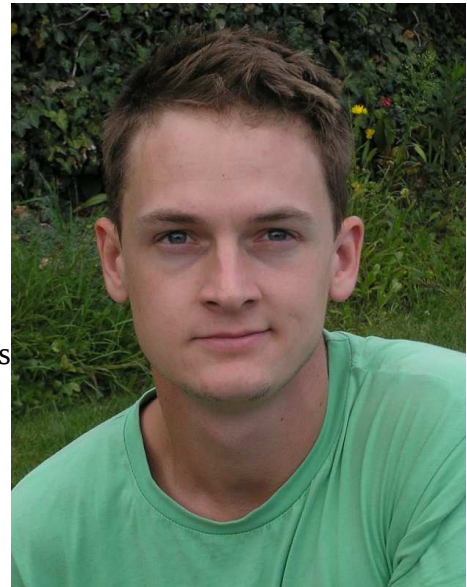
Prostejov, una pequeña ciudad en la parte este de la República Checa. Vivo aquí porque nací aquí, crecí aquí y me encanta este lugar.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Vim es lo mejor. Y también otras herramientas sin las que no podría vivir. Como por ejemplo Git. En mi PC tengo KDE4 con openSUSE. La versión de desarrollo en mi PC de escritorio, y la última estable (conocida como Tumbleweed) en mi portátil.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Porque quería construir un kernel, pero el sistema no funcionaba del todo bien en algunas áreas. Así que arreglé eso y ese fue mi primer parche.





Después hubo un parón porque estuve trabajando en los drivers NetBSD para un proyecto de redes Checo. Pero regresé a Linux, convertí los drivers NetBSD a Linux y finalmente continué trabajando solo en Linux en los proyectos que ya he mencionado más arriba.

### **¿Qué es lo que hace que sigas interesada en esto?**

Porque nadie puede realmente entender el kernel completamente. Es un constante ecosistema en cambio, y uno siempre tiene que estar aprendiendo algo nuevo. Y también por la gente. Ellos me enseñaron realmente a programar.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Siempre menciono un momento cuando finalmente encontré un error de una corrupción silenciosa que había introducido alguien. Era en el proceso de la suspensión a RAM. Me recuerdo siete días y siete noches pensando y persiguiendo el mismo error. Era que siempre aparecía el valor 0xf0 en algún lugar aleatorio de la memoria. Cuando finalmente pude poner mi huella sobre la pieza rota del código, estaba contento de haber derrotado finalmente a ese error.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Sé lo difícil que es empezar por tu cuenta, cuando no tienes a nadie para preguntar sobre cómo una parte en particular del kernel esta diseñada para trabajar. Si quieres empezar a programar con el kernel y tienes cerca a algún conocido que sepa del kernel, no seas tímido y no dudes en pedir ayuda. Estudiando por tu cuenta también es un camino, pero so te llevará mucho más tiempo.

También por esa razón estoy dando lecciones de kernel en mi “alma mater”. Así que revisa en tu universidad si hay folletos que anuncien cursos o cosas así.

### **¿Qué escuchas mientras programas?**

A menudo la gente se equivoca cuando tratan de adivinar eso. Escucho Heavy Metal (Iron Maiden, si tengo que elegir uno) Trash Metal (Sepultura) y quizás algo de Rock (nadie en particular). Pero si a veces bluegrass

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Estoy suscrito a la lista de correo LKML. Desde que soy más estudiante que hacker, cada vez más asisto a conferencias de ciencias de la computación. Pero de vez en cuando aparezco en conferencias sobre Linux también. Como Linux Kernel Summit el pasado año y quizás LinuxCon Europe este año.

## 17.- Laurent Pinchart

### Nombre:

Laurent Pinchart

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Estoy centrado sobre todo en video captura (V4L2) y más recientemente, soporte para display (DRM y FBDEV) para plataformas embebidas. Depende a quien preguntes, la gente me conocerá como “El tipo de los controladores multimedia” (dentro del marco multimedia en el núcleo del kernel) o “El tipo UVC” (USB Videl Class, un estándar USB de webcams).

Además de mantener varios drivers relacionados con el vídeo, estoy implicado en el diseño y desarrollo de controladores multimedia, subsistemas V4L2, DRM y FBDEV, y también escribo documentación para sus API's.

También considero que revisar parches y dar soporte a los usuarios de mi código es una parte importante. Aunque no dedico tanto tiempo a esto como me gustaría .



### ¿De dónde recibes tu nómina?

Renesas Electronics es actualmente mi principal ingreso. También ofrezco servicios de consultoría a clientes más pequeños a través de Ideas on Board, mi compañía de consultoría integrada Linux.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en Bélgica, principalmente porque he pasado toda mi vida aquí. La escena local del desarrollo del kernel es desafortunadamente muy pequeña. El software libre y de código abierto está, sin embargo, bien representado gracias a eventos como el FOSDEM.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Vim siempre está a mano en mi lista de los más usados, pero trato de no ser muy “fanático” con esto. Más recientemente, otro cambio sustancial fue git. Ha incrementado mi productividad y ahora lo utilizo para todos mis proyectos.

Y para mi PC, en este utilizo KDE, quizás por razones históricas. Manías de desarrollador, seré feliz mientras mi sistema me permita la opción de tener el control de la consola sólo mediante el teclado.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

A pesar de que ya había empezado a escribir código del kernel en 2002 en una pequeña empresa Belga, la verdadera diversión empezó en 2005 con un típico problema de “rásate tu propio picor”. Me había comprado una webcam para un proyecto personal sobre robótica y decidí intentar escribir mi propio driver para Linux. Siete años después mantengo el controlador UVC para cámaras que soporta más de 250 modelos.

La impredecible naturaleza del desarrollo del kernel de Linux siempre me sorprende. Trabajar en el controlador UVC me permitió entrar en contacto con Logitech (resultó que un estudiante de informática había planeado escribir el mismo controlador para su tesis de maestría durante una pasantía en Logitech y tuvo que cambiar sus planes cuando publiqué mi código – lo siento por eso Martin) Esto llevó a un primer contrato pagado para un desarrollo embebido de Linux en la plataforma TI DaVinci, contactos con la comunidad del kernel ARM y eventualmente propuestas de contratos con Nokia. Me gustaría aprovechar esta oportunidad para agradecer a Kevin Hilman, no podría haber llegado donde estoy hoy si no me hubiera recomendado entonces.

### **¿Qué es lo que hace que sigas interesada en esto?**

El desarrollo del kernel de Linux me hace sentir útil a la gente, no sólo a empresas. A lo largo de estos años he aprendido a sentirme apasionado por sus cambios técnicos, es sorprendente que los desarrolladores de los que continuamente aprendo, compartan su cultura y su ecosistema dinámico me lleva a inesperados – pero siempre interesantes – viajes.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Fue más sorprendente que divertido, la historia que recuerdo ocurrió a principios de 2008. Un desarrollador de una fábrica de un chipset de una webcam de Taiwan contactó conmigo para preguntarme si había considerado venderles el controlador de Linux UVC por 50.000\$. Después de recuperarme del shock, le pregunté que el controlador no estaba en venta, pero estaba disponible para ellos para utilizar de manera libre bajo licencia GPL.

Un par de correos electrónicos más tarde, no entendían como yo podía pasar mi tiempo desarrollando el controlador y liberar el resultado como código libre, el desarrollador se ofreció a mandarme 250\$ de su propio dinero, diciendo que él no podía pagar más debido a que su hijo iba a la universidad y le resultaba muy caro. No necesito decir que rechacé el dinero, pero recuerdo que esta historia me conmovió profundamente.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

“Ráscate tu propio picor”. Pocos (si es que hay alguno) proyectos de código libre y abierto que tienen éxito empiezan a lo grande. Necesitarás gente interesada en un proyecto que te ayude en los malos momentos.

A los estudiantes, me gustaría darles un consejo para que se involucren mientras están estudiando. No lo creerán, pero tendrán menos tiempo una vez que se gradúen. Las fiestas y las bebidas son divertidas y tentadoras, pero esto no hace un buen Curriculum Vitae.

Finalmente no te tomes las revisiones de tu código de manera personal. Los desarrolladores más experimentados tienden a olvidar que no sentirse ofendido por una crítica no es una habilidad innata. Recuerda que las revisiones son a tu código, no a ti mismo de manera personal (hay excepciones a esta regla, pero esto ocurre muy raramente). Consideralas como un consejo de un experto, intenta aprender de ellas y no te disgustes. Terminarás considerándolas como un proceso natural.

### **¿Qué escuchas mientras programas?**

Esto depende del trabajo y de mi estado de ánimo. Necesito silencio (o a veces incluso una vuelta en bici) para concentrarme en trabajos difíciles, pero me gusta relajarme con un buen té y música de fondo cuando estoy trabajando en problemas menos complicados, o escuchar canciones pegadizas para acelerar mi ritmo de trabajo. Por mencionar algunos: Girls in Hawaii (una banda famosa en

Bélgica de indie pop), The killers, Indochine o Les Cowboys Fringants son habituales en mi lista de escuchados.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Las listas de correo que más sigo son: [linux-media@vger.kernel.org](mailto:linux-media@vger.kernel.org), [linux-sh@vger.kernel.org](mailto:linux-sh@vger.kernel.org) y [dridevel@lists.freedesktop.org](mailto:dridevel@lists.freedesktop.org). La lista de correo [linux-kernal@vger.kernel.org](mailto:linux-kernal@vger.kernel.org) tiene mucho tráfico de correos para estar revisándola todos los días. La mejor manera de asegurarte de que vea tus correos, es poniéndome a mí en copia (CC)

También me paso por las salas de IRC #v4l and #dri-devel en [freenode.net](http://freenode.net). Me puedes “hacer ping”, pero por favor utiliza la lista de correo adecuada para preguntas que puedan ser interesantes para más gente.

La conferencias son mi forma preferida para socializarme con otros desarrolladores del kernel. Las recomiendo; encontrarse con gente cara a cara hace las futuras interacciones en las listas de correo más fáciles. Trato de ir por lo menos a una como Linux Plumbers Conference, Embedded Linux Conference or FOSDEM cada año, y trato de dar una charla sobre el trabajo en el que estoy desarrollando. No me encontrarás siempre en las salas de las conferencias. Paso la mayor parte del tiempo en discusiones informales por los pasillos que escuchando las charlas.

## 18.- Jiří Kosina

### Nombre:

Jiří Kosina. Pero como la letra 'ř' es imposible de pronunciar para casi nadie que sea de fuera de la República Checa, he terminado por utilizar Jiri.

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

El archivo de MANTENEDORES me nombra como la persona responsable para múltiples subsistemas dentro del área del kernel. Las más interesantes son, en orden decreciente en cuanto a relevancia:

- Soy el mantenedor de los subsistemas HID (dispositivos de interfaz humana en inglés: human interface devices) Esto significa manejar una amplia variedad de hardware, empezando por USB/Bluetooth micrófonos/teclados/joysticks, y también incluyendo dispositivos multitáctiles (donde es Benjamin Tissoires el que está realizando la mayor parte del trabajo). Otras partes interesantes van apareciendo en el horizonte y son HID sobre I2C soporte para sensores HID, etc



- Disposición aleatoria de la dirección del espacio para los ejecutables del espacio de los usuarios (N.T: La frase original es: "Address space layout randomization for userspace executables." si alguien tiene una versión más acertada que la comparto!) Gran parte de este código fue extraído originalmente de ExecShield pero que he remodelado en varias ocasiones, se ha unido a la rama estable del código de Linus, y se ha ido depurando con el paso de los años. Había algunos errores muy extraños, de hecho en su mayoría relacionados con la versión libc5 (sí, todavía hay gente funcionando en kernels 3.x) versiones muy antiguas de emacs asumen esta dirección como la última de las variables globales declaradas y marca el principio del brk(). Una experiencia rara.

- Soporte APM (para aquellos que no son muy viejos para recordarlo, sí esto es lo que utilizábamos antes de que se inventase ACPI ). Me encargué de esto porque todavía tengo hardware que intento mantener actualizado con kernels al día, y sólo funcionan con APM. Está siendo tranquilo aunque a nadie parece preocuparle mucho (excepto para el apoyo inactividad de la CPU que está siendo obsoleto y actualizado constantemente).

- El controlador del Floppy. Me metí en esto recientemente. Tuve que luchar con el actual mantenedor hasta la muerte en la "cueva del trueno" y... bueno actualmente el controlador ha estado desatendido durante mucho tiempo y se ha convertido en algo muy espinoso. Lo más divertido de esto es que siempre tenía que estar, pero a nadie le importaba, el hardware real del floppy es demasiado lento para provocar un desarrollo en el código. Pero para aquellos que son lo suficientemente valientes para virtualizar el controlador del floppy les pillará esto por sorpresa, el dispositivo del floppy virtualizado es endemoniadamente rápido. Después de haber solucionado esto el controlador vino a parar a mí. Bueno, no es la carrera con la que siempre soñé.

- Y otras ramas más triviales. Principalmente recoger parches por los que nadie más se preocupa, y mandárselos a Linus para que los revise.

### **¿De dónde recibes tu nómina?**

SUSE. Trabajo como responsable del equipo del “núcleo del kernel” en el departamento de SUSE Labs y soy el mantenedor responsable de los kernels que dan forma a los productos de SUSE Linux Enterprise.

### **¿En qué parte del mundo vives, y porque allí?**

Praga, en la República Checa. He nacido aquí, y excepto un intermedio cuando estuve trabajando en el CERN, realmente nunca he tenido una razón para vivir en cualquier otro sitio.

### **¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Soy usuario de KDE desde mis primeros tiempos. Pero KDE4 estaba en un punto crítico, no pude evitar la tentación de cambiarme a otros entornos más livianos. Así que ahora utilizo icewm lleno de xterms, vim, gcc, git, quilt, (al)pine, navegadores, algún cliente de mensajería instantánea, twinkle para voip, mplayer... y poco más.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Estábamos configurando un router para Internet (usando un módem a 9k6 bps po supuesto) cuando íbamos al instituto en 1996, y había algo que no funcionaba bien, creo que fue un punto muerto en la tarjeta de red lo que nos estaba bloqueando. Me dí cuenta que podía obtener el código fuente en algún sitio y tratar de investigar qué pasaba. Me costó muchas semanas sin dormir bien hasta que entendí todo pero al final fui capaz de solucionarlo... sólo apuntar que fue solucionado ya en la versión 2.0.17. Pero fue una experiencia sorprendente.

### **¿Qué es lo que hace que sigas interesada en esto?**

La combinación única de alta tecnología y alta interacción social en esto.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Me gustan las sesiones de depuración mediante correo electrónico. Tratando de averiguar por qué el sistema de la persona que reporta el bug se comporta como lo hace, poniendo todas las piezas de información en conjunto, tratando de encontrar maneras inteligentes de cómo instrumentar el código para conocer un poco mejor lo que está sucediendo. Es como leer una novela policíaca dramática, con el añadido de que además puedes participar en ella.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Si necesitas preguntar cosas como: “Quiero ser un desarrollador del kernel, porque pienso que está genial, ¿en qué puedo trabajar?” empieza cambiando primero tu forma de pensar.

Estoy convencido de que hay muchas cosas que detestas de Linux, así que ¿por qué no empiezas cambiando eso? ¿No es tu servidor casero muy lento? ¿No se estropea tu audio cuando estas haciendo tareas intensivas de Entrada/Salida de datos? ¿No odias que se produzca un fallo OOM que siempre decide “colgar” tu editor de texto cuando estás escribiendo una carta importante de amor?

Realmente mi consejo básico sería: deja de ignorar las pequeñas cosas que te molestan “sólo un poco” y empieza por resolverlas. El resto vendrá solo.

**¿Qué escuchas mientras programas?**

Cuando trabajo en algo que mi cerebro considera que será un duro problema, necesito silencio. Pero en otras ocasiones tengo un amplio gusto musical, que van desde la música clásica (Mozart sobre todo) hasta el rock o el punk

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Listas de correo de la comunidad: Principalmente LKML y linux-input, más de vez en cuando usb-devel

IRC: /whois jikos

Conferencias: Kernel Summit + las conferencias aledañas, y una selección aleatoria de otras conferencias relacionadas con el kernel, a las que no asisto necesariamente año tras año.

## 19.- Chuck Lever

### Nombre:

Chuck Lever

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Trabajo sobre NFS para Linux e implementaciones FedFS

### ¿De dónde recibes tu nómina?

Oracle

### ¿En qué parte del mundo vives, y porque allí?

Ann Arbor, Michigan. Vine al colegio aquí hace ya 30 años, me gustó y me quedé. Estaba cerca de la familia. En la década anterior un cuadro de desarrolladores de NFS se formó aquí. Trabajar con una comunidad global como Linux significa viajar, y ciertamente he aprendido a apreciar la singularidad de mi ciudad y de muchas otras ciudades.



### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Esto es un anatema para algunos, ya que tengo instalado MacOS en mi escritorio y hago el desarrollo del kernel de Linux en un entorno virtualizado. MacOS ofrece herramientas fáciles de utilizar además de un gran abanico de herramientas enfocadas para la productividad. Una de estas herramientas que utilizo mucho es Evernote.

La virtualización significa que puedo tener tantos sistemas desarrollados como necesite sin preocuparme de la potencia, del ruido, del enfriamiento, de la compra de nuevo hardware y el reciclado. Prefiero herramientas desde la línea de comandos y vim, y me gusta Stacked Git para el manejo de parches.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Algunos recordarán un embarazoso informe durante la época de expansión de internet a finales de los 90 que concluía que Windows/NT era una plataforma de servidores más escalable para internet que Linux. Un programa de investigación de la universidad de Michigan empezó a ayudar a Linux a mejorar los servicios de internet. En aquel entonces, esto significaba sólo lo básico: LDAP, IMAP y HTTP. Yo estaba cansado de ser un administrador de sistemas a gran escala, por lo que me apunté. Finalmente, el trabajo pasó a centrarse en NFSv4.

### ¿Qué es lo que hace que sigas interesada en esto?

Ayudo a mejorar el software que la gente usa todos los días para hacer su trabajo. La escalabilidad del rendimiento, la facilidad de administración y la confiabilidad de almacenamiento de datos es crucial para todos los sistemas de computadoras del mundo.



### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Se necesitan muchas habilidades técnicas importantes: programar de manera fluida, manejar herramientas de control del código fuente, revisar el código programado por otros, resolución de problemas, probar lo que has escrito y así una vez y otra. Lo más importante de todo esto y lo más difícil de hacerlo bien son las revisiones de código y las pruebas exhaustivas. Debes aprender a manejarlo pronto.

Un joven desarrollador puede encontrar un mentor que entienda y respete su estilo de trabajo. Una de las mejores cosas que puedes aprender de un buen mentor es cómo criticar de manera constructiva. Otra cosa importante es cómo trabajar bien con otros desarrolladores. Lo amplio de nuestra comunidad es algo que las metodologías del código privativo no pueden alcanzar.

### **¿Qué escuchas mientras programas?**

Ambient/downtempo, clásica instrumental, y alguna vez jazz tradicional. Cualquier cosa con letra o cantado me distrae muy fácilmente.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

#linux-nfs on oftc.net... LSFMM (Linux Storage, Filesystems, and Memory Management), y en alguna ocasión en , LPC (Linux Plumber's Conference). Recientemente he acudido a los encuentros IETF (Internet Engineering Task Force).

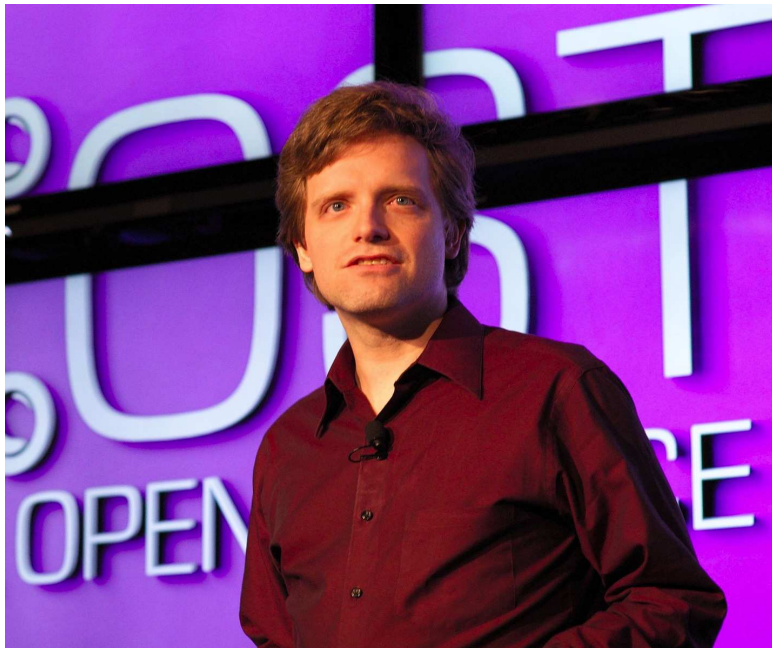
## 20.- H. Peter Anvin

### Nombre:

H. Peter Anvin. La “H” es sinónimo de Hans, pero nunca la he utilizado como primer nombre, ni tuvo intención de serlo.

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

En la actualidad mi principal proyecto está siendo como co-mantenedor de la arquitectura x86 junto con Ingo Molnar y Thomas Gleixner. He hecho un montón de cosas a lo largo de los años, incluyendo la creación de kernel.org, el sitio de distribución del Kernel de Linux; la creación del cargador de arranque Syslinux y klibc; escribir la versión original de autofs; mantener Netwide Assembler; y muchas otras cosas. Con unas cuantas todavía estoy involucrado, y otras las he ido dejando por el camino.



### ¿De dónde recibes tu nómina?

Intel Open Source Technology Center. Un lugar divertido con un montón de gente divertida. Trabaja siempre es agradable cuando lo haces rodeado de gente inteligente.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en San José, California. El tiempo es fantástico, y me gusta el ambiente de Silicon Valley. Es un lugar donde la creatividad y la originalidad son valoradas, y hay una actitud de “poder hacer” que realmente es estimulante.

### ¿Cuál es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Git quizás sea mi herramienta favorita de desarrollo sin duda, La rapidez y flexibilidad permite hacer cosas que nunca habría imaginado de usar las antiguas herramientas de control de versiones.

Tengo instalado KDE en mi PC. Siempre digo que tengo que probar cosas nuevas, pero siempre suelo regresar rápidamente. Creo que podría utilizar Xfce si pudiera pasar un rato configurándolo a mi gusto, pero por el momento no encuentro una razón para perder ese tiempo.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

En 1992, cuando todavía no me había graduado, me topé con Unix en la universidad y me gustó de veras. Había visto Microsoft Windows y me pareció realmente horrible. Utilizaba el 286 de mi padre, pero había ahorrado dinero para un ordenador durante bastante tiempo. Se acercaba el momento y empecé a preguntarme que sistema operativo debería instalar y había estado siguiendo

una serie de artículos en Dr. Dobbs Journal titulados “Porting BSD to the 386” que describían el todavía inacabado proyecto 386BSD.

En aquellos días, Usenet era tan pequeña que cuando te registrabas, tu lector de noticias te preguntaba si querías unirte a cada grupo creado recientemente. Así que cuando un grupo llamado “alt.os.linux” apareció de repente me uní y me conecte con el proyecto Linux. Poco después tuve un accidente y pasé unas semanas en el hospital. Mi padre decidió deshacerse del PC, y terminamos por encargar un increíble 486 (con 8 megas de RAM y 500 megas de disco duro) desde la cabina del hospital. Cuando lo tuve, empecé a experimentar con Linux, pero también probé a aprender OS/2. Los manuales de OS/2 que había pedido me los terminaron robando(!) así que terminé pasando mi tiempo con Linux, que obviamente terminó siendo una buena cosa. Así que a quien quiera que me robara los manuales, felicidades. Espero que haya pasado mucho tiempo tratando de ser un reputado desarrollador de OS/2.

Linux estaba lejos de ser el primer “clone barato de Unix” en esos días. Sin embargo, al menos para mis ojos en ese tiempo, las dos grandes diferencias entre el “Unixen real” y el “recién llegado” eran X11 en cuanto a gráficos y TCP/IP para la red. Cuando empecé a jugar con él, Linux no tenía ninguno. Pero seis meses después, en otoño de 1992 tenía ambos. Ahí fue cuando me dí cuenta que Linux había llegado para quedarse.

### **¿Qué es lo que hace que sigas interesada en esto?**

Por un lado, que los retos nunca terminan. Mi portátil hoy es al menos 10.000 veces más potente que el “monstruoso” 486 de hace 20 años, y Linux hoy es usado en ordenadores que son miles de veces más potentes que ese, y miles de millones si cuentas los clústers. Al mismo tiempo mi teléfono a día de hoy, que también es mucho más rápido que el viejo 486, también utiliza Linux.

Linux se ha colado en casi todos los rincones de la sociedad moderna, debido a su versatilidad, y estoy firmemente convencido de que la versatilidad es un resultado directo de su naturaleza de código abierto.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

En lo que se refiere a algo gracioso, nada mejor que lo que se publica entorno al 1 de Abril (April's fool). Particularmente me gustan aquellas que son creíbles pero totalmente inútiles. Por ejemplo, RFC 2324 Hypertext Coffee Pot Control Protocol (Protocolo de control de Hypertexto cafetera). Es parte de la cultura hacker que me encanta.

En cuanto a discusiones encarnizadas, solía tener un colaborador en la lista de correo del kernel Linux que escribía extensos correos con explicaciones detalladas de ordenadores arcaica (N.T: ¿?) principalmente dirigidos a principiantes. Esto habría sido absolutamente fantástico si no fuera por el problema de que las explicaciones eran erróneas. Los desarrolladores más experimentados simplemente lo ignoraban, pero los novatos podían ser engañados, lo que no era justo para ellos. Ya en un punto de exasperación, varias personas incluyendo a Matt Mackall (de Mercurial fame) y a mí empezamos a comentar sus sin sentidos empezando los correos con “Querido bot-erróneo” y se quedó con el mote. Dejo de publicar al poco tiempo.

Y referente a logros increíbles... creo que el desarrollo temprano de git habla por si solo. Pasar de literalmente nada a algo útil en aproximadamente dos semanas fue un logro realmente sorprendente debido no sólo a su potencia si no también a su modelo de desarrollo, era muy fácil conseguir que otros desarrolladores se sumaran al carro y profundizaran en su funcionalidad. Ciertamente es una

clase de situación extraña el tener multitud de desarrolladores experimentados altamente motivados, pero oye, así es como suceden las cosas raras.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Empieza poco a poco. Haz algo que te interese a ti. Pero si tienes la oportunidad, inténta tu primer proyecto, con algo que no interese a mucha gente, esto hace que la apuesta no sea muy alta y exista el riesgo de que tu primera prueba sea una prueba de fuego. Es mejor familiarizarse primero con el proceso y después preocuparse por los grandes asuntos.

Si puedes, búscate un mentor.

Espera alguna respuesta cortante, especialmente en los círculos del kernel de Linux. No te las tomes de forma personal. Yo he tenido personalmente respuesta de Linus Torvalds llamando a las cosas “estúpidas” e “idiotas” en numerosas ocasiones, y muchas de las veces estaba en lo cierto, pero por lo que sé el nunca me ha llamado a mí personalmente estúpido ni idiota. Hay una gran diferencia en eso: incluso las personas muy inteligentes con regularidad a hacer cosas que probablemente parecían tener sentido en el momento, pero por alguna razón no salieron demasiado bien.

### **¿Qué escuchas mientras programas?**

La mayor parte del tiempo nada, pero de vez en cuando pongo un tema de música que suena una y otra vez. “Hotel California” de The Eagles es mi tema favorito por una razón particular.

¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?

Más de las que puedo enumerar. No leo toda la lista de correo de Linux Kernel, poca gente hay que lo haga, si es que hay alguien, pero sí que la ojeo regularmente. Y actualmente estoy conectado en 40-50 canales de IRC, pero la manera más fácil de encontrarme probablemente sea en #syslinux en Freenode.

Y en cuanto a conferencias, trato de ir a Linux Plumber’s Conference y a cualquier conferencia a la que este unida la cumbre de Linux Kernel cada año. Más allá de eso, depende.

## 21.- Steven Rostedt

### Nombre:

Steven Rostedt (me llaman Steven o Steve, no me llames Stephen, ese no soy yo y creeré que estás hablando de uno de los muchos otros desarrolladores del kernel que se llaman Stephen)

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

El trabajo que supongo que hago es trabajar en los parches de tiempo real del kernel de Linux. Mantengo los lanzamientos estables de los parches de tiempo real, y debo seguir la línea principal de los lanzamientos estables.

Hace tiempo, estuve trabajando en conseguir incorporar una infraestructura de la latencia de rastreo del parche -rt dentro del kernel, también combinado con una herramienta de rastreo que escribí hace años y que acabó siendo lo que ahora se conoce como ftrace.

Ftrace ha terminado siendo un pionero en la rama principal del kernel como infraestructura de rastreo. Otras herramientas trataron de convencer de la utilidad del rastreo a los desarrolladores del kernel, pero no fue hasta que llegó ftrace que los desarrolladores del kernel se dieron cuenta. Entonces todos empezaron a usarlo y en la actualidad hay alrededor de 700 eventos en todo el kernel.



Mantenerse actualizado con la demanda me ha apartado de trabajar más en -rt (aunque todavía doy prioridad al mantenimiento de -rt estable) Lo que me ha llevado a crear otras herramientas como trace-cmd y kernelshark. Lo que ha implicado otros eventos perf de software.

### ¿De dónde recibes tu nómina?

Red Hat me paga por mi afición

### ¿En qué parte del mundo vives, y porque allí?

Vivo en Endwell, Nueva York, que comparte el mismo código de correos que Endicott, Nueva York, que es donde me crié. Una pequeña curiosidad. Hace 100 años IBM empezó su andadura en Endicott, así que nací en la misma ciudad (si quieres llamarlo así) que Big blue.

Mi padre fue director de IBM, y le trasladaron a Endicott a finales de los 60. Aunque IBM ya no está aquí hace tiempo, ya que era muy grande a principios de los 90. Lo que me lleva a preguntarme porque sigo aquí.

Me gradué en Ciencias de la Informática (N.del T: No sé si eso será un término correcto) en 1991, durante el periodo de recesión lo que fue un problema para encontrar trabajo. Incluso estuve trabajando de introductor de datos para la oficina del servicio de impuestos de Nueva York. Finalmente mis deudas eran mayores que mis ingresos y en vez de terminar viviendo por las calles, lo dejé todo y volví a la casa de mis padres.

Allí encontré un trabajo en GE (N.del T: ¿General Electric?) y mi primera toma de contacto con sistemas en tiempo real. Me permitió poder dejar la casa de mis padres. Hice pruebas para el control

del motor C17. Después GE fue comprada por Martin Marietta y después por Lockheed por lo que se convirtió en Lockheed Martin. Al mismo tiempo fui a trabajar a IBM Federal systems en Owego, Nueva York, que más tarde fue comprado por Loral, que más tarde fue también comprado por Lockheed Martin (parece que no puedo escapar de ellos). Trabajé para 5 empresas diferentes y sólo tuve 2 despachos. Estaba tan confundido de para quien trabajaba que al final empecé a contestar al teléfono diciendo únicamente “La Compañía”

Volviendo a tu pregunta, la razón por la que sigo aquí es porque me casé, compré una casa y tuvimos unos hijos. Red Hat me permitió trabajar de manera remota y no he encontrado una razón para irme de aquí. Los terrenos y las casas aquí son realmente muy baratas y no encontraría nada comparable que pudiera pagar si me mudara a la zona de California a Boston.

### **¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?**

Trabajo con dos monitores y dos teclados en mi equipo de escritorio, que estan conectados a dos sistemas diferentes. Mi PC de trabajo tiene instalado Fedora 13 (no he necesitado actualizarlo) y mi otro PC (personal) tiene Debian testing. Para Fedora utilizo Gnome 2, y en Debian utilizo Xfce 4. No estoy dentro de los paradigmas del usuario clásico.

Mi herramienta favorita de desarrollo podría ser Emacs, mi sistema operativo por encima de los otros. Eso y por supuesto mi propio script ktest.pl. Mi trabajo lo desarrollo básicamente: Actualizar código con emacs y guardarlo en la línea de comando escribo `./ktest.pl <box> .conf`

Y mientras espero a que ktest realice su trabajo leo lwn.net o heise.de, o instalo y arranco el kernel con mis modificaciones actuales. También puedo ejecutar pruebas, pero principalmente lo tengo para arrancar el kernel donde por medio de ssh arranco y pruebo cómo funcionan las cosas. Ya hace mucho que no escribo el típico “make” o “make install”.

Antes de subir el código a la rama principal, siempre hago varias pruebas usando ktest, para asegurarme de que mis parches funcionan y no causan ninguna regresión conocida.

Así como las herramientas mencionadas como son: git, perf, trace-cmd y kernelshark. Son herramientas que también utilizo de manera habitual.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Mientras trabajé en las compañías que he mencionado antes, trabajaba principalmente con Sun (Solaris) y AIX, un sistema Unix. Trabajando principalmente en un entorno militar sentía que mi carrera se estaba encasillando. Parecía que todo lo divertido y excitante estaba sucediendo entorno a los ordenadores, y ese sistema operativo nuevo de Microsoft. En 1996, por fin fui capaz de seguir mi camino en un proyecto basado en Windows NT. Fue utilizando C++ usando Windows Visual C++.

Fue un desastre. Honestamente me sentí como si tuviera que programar con una mano atada a la espalda. Después de pasar varios días fiabilizando una variable corrupta, encontré que el problema no sucedía donde creía, si no que el depurador de Visual C++ decidió decirme que la variable cambió cuando en realidad no lo hizo. Añadiendo dos printf estratégicamente, solucione este caso. Harto de tanta frustración exclame bien alto: “Por que no pueden inventar un sistema Unix para el PC!”. Un compañero que estaba sentado cerca de mí se giró y me preguntó si había oído hablar de Linux?.

Hice lo típico, descargar los 13 disquetes que contenían la imagen de Slackware, y mi viaje comenzó. Pensé que era increíble que pudieras ver el código fuente y el kernel. Había un error en el kernel para uno de mis dispositivos, y después de una búsqueda rápida en internet encontré el parche que lo solucionaba. Bien no era un parche, eran instrucciones para cambiar el kernel. Pensé que eso era fantástico: “Estoy cambiando el kernel en mi PC!” Pero no es así como me hice desarrollador del kernel.

En 1998 mientras estudiaba un Master en la Universidad de Binghamton, uno de los profesores daba una clase sobre redes. El proyecto principal era convertir Linux 2.0 TCP stack en un protocolo credit/NAK. Durante esta clase encontré mi llamada. Me enamoré del todo de la programación del kernel. Me obsesioné con ello. Por desgracia lo más que podía hacer en mi trabajo era utilizar VxWorks kernel, que no era tan divertido.

En 2001 una compañía que empezaba llamada TimeSys abrió una oficina cerca de donde vivía. Tuve que luchar duro para conseguir un trabajo allí. Mi experiencia laboral era más que nada como usuario, y mi trabajo con el kernel era sólo como afición. Gracias a mis amigos que ya trabajaban allí fueron capaces de convencer a los directores de que tenía experiencia con el desarrollo del kernel ya que lo hacía en mi tiempo libre.

### **¿Qué es lo que hace que sigas interesada en esto?**

La comunidad. Somos como una gran familia disfuncional, y me gusta eso. Hay mucha rivalidad en la lista de correo, pero después cuando te encuentras en una conferencia, puedes tomarte con ellos una cerveza y reírlos juntos. Tienes que ser un tipo de persona peculiar para vivir en este entorno, y eso no es fácil. El truco es no llevar las cosas al terreno personal. Tienes que sopesar que es lo mejor para ti y que es lo mejor para la comunidad. Estamos intentando conseguir lo mejor de eso, pero al mismo tiempo, estamos intentando ayudar a otros. A veces la gente se olvida de este detalle y se dañan sentimientos.

Programar es un arte, y me gusta todo el arte, es un reflejo de lo que eres. Cuando alguien te dice que tu código apesta, puedes sentirte como si te lo dijeran a ti personalmente. Pero recuerda, si tu arte mejora tu también mejorarás. Si puedes sacudirte todas esas críticas, y mejorar tu trabajo, también mejorarás tu.

Me he sentido a veces tan hundido que he estado a punto de dejarlo. He pensado seriamente abrir una tienda de Starbucks en la carretera y dedicarme a eso. Pero no lo he hecho, y siento que soy mejor persona debido a ello. Además, tener mi propio Starbucks suena interesante.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Lo que más me ha sorprendido fue mi “#define if()” (ver include/linux/compiler.h) Estaba asombrado de cómo funcionaba. Estaba incluso tan asombrado que lo incluí en el kernel. Cuando no se configuraba, no producía ningún daño en el kernel, así que no era un riesgo.

Todavía lo utilizo para encontrar heurística en varios códigos. Alguien me sugirió una forma de modificarlo para que fuera actualmente más útil para otros sin riesgos de rendimiento cuando este era habilitado. Todavía hoy está en mi extensa lista de cosas por hacer.

### **¿Cuál sería tu consejo para los desarrolladores que quieran implicarse?**

Encuentra algo en lo que quieras trabajar y hazlo. Una cosa que he aprendido trabajando en los

parches -rt es que no puedes añadir cosas en el kernel. Esas cosas son las que necesitan que sean añadidas. No vengas diciendo que eres el mejor en una nueva característica y que todo el mundo debería tirar pétalos de flores a tus pies. Necesitas convencer a los mantenedores actuales que lo que tienes les beneficiará. Si puedes conseguir eso, tendrás a gente que se interese por tu código.

Necesitas empezar a ayudar a otros y a la comunidad, y esto no significa arreglar espacios en blanco o errores tipográficos. Si puedes contribuir con un beneficio de rendimiento, o simplemente limpiar código para hacerlo más sencillo de mantener y entender, sin sacrificar el rendimiento, estarás haciendo a la comunidad un gran servicio.

### **¿Qué escuchas mientras programas?**

Los ventiladores de mi PC. No escucho música. Además mientras espero a que ktest termine su tarea, suelo leer algún podcast en Alemán para recordarlo y practicarlo. Ultimamente también he hecho un curso rápido de español para estar preparado para mi viaje a Barcelona

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Estoy en LKML, y [linux-rt-users@vger.kernel.org](mailto:linux-rt-users@vger.kernel.org). También me suelo pasar por el canal de OFTC #linux-rt (N.del T: “where we like to throw around large fish with thermo-deficiencias.” esta frase supongo que sea un tipo de frase hecha que no he sabido encontrar un parecido en español, si alguien tiene alguna idea...)

Normalmente suelo asistir a cinco conferencias al año. Me puedes encontrar en Linux Collaboration Summit y en Plumber. A veces también asisto a Embedded Linux, LinuxCon y Linux Users . También a OSADL’s Realtime Linux Workshop de la que he regresado hace poco y me gustó. Pero aquí de nuevo es la conferencia que más me gusta.



## 22.- Stephen Hemminger

### Nombre:

Stephen Hemminger

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Soy uno de los desarrolladores del subsistema de red. Mis contribuciones se extienden a todo este campo, desde escribir controladores para nuevos dispositivos o solucionar errores hasta integrar nuevos aportes de proyectos de investigación.

### ¿De dónde recibes tu nómina?

Trabajo para Vyatta que es una distribución de Linux especializada en redes. Debido a que somos una empresa pequeña, he terminado haciendo de todo. Desde mantenedor del kernel hasta ingeniero de rendimiento. Esto me ha dado lugar para aprender sobre Pearl, protocolos de routing y Debian.



### ¿En qué parte del mundo vives, y porque allí?

Aunque me crié en la costa este, me mudé a Portland, Oregon hace ya unos cuantos años para poder trabajar en una estación con Unix en Tektronix. Eso no duró mucho, ya que las oportunidades aquí me permitieron trabajar tanto en grandes empresas como en empresas que estaban empezando.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

La mejor herramienta productiva ahora mismo es KVM. Me permite crear nuevas máquinas (o incluso toda una red) con un mínimo de esfuerzo. Todavía paso mucho tiempo moviendo hardware y cableado de un sitio a otro, pero ahora mucho menos que hace unos años.

Desde que Vyatta está basado en Debian, es muy simple simplemente hacer funcionar Debian en un PC de escritorio. Mi afición secreta consiste en estar actualizándolo todo. Quizás se deba a mi creencia optimista de que el mundo siempre va a ir a mejor, pero resulta que siempre termino probando (y tratando de solucionarlo) nuevos lanzamientos todo el tiempo.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Cuando empecé en OSDL fui contratado como responsable de la iniciativa del Centro de Datos de Linux (Data Center Linux en inglés: DCL). DCL fue un proyecto en búsqueda de un problema, desde que Linux ya era un éxito sin marketing adicional o esfuerzos técnicos. Esto ocurrió durante la mitad del desarrollo del ciclo de la versión 2.5 cuando muchas cosas cambiaron todas a la vez. Pat Mochel estaba trabajando en sysfs y yo lo adelanté y lo implemente para dispositivos de red.

El otro punto clave fue cuando Linus preguntó: “¿Podremos deshacernos de brlock?” lo que me llevó a cambiar todo el protocolo de red. Entonces David quiso solucionar el arranque de dispositivos de red, lo que significaba cambiar todos los controladores de dispositivos de red. Estos son ejemplos de cómo los desarrolladores de Linux no tienen miedo de hacer una profunda revisión del código que otros proyectos evitan hacer.

### **¿Qué es lo que hace que sigas interesada en esto?**

En el aspecto técnico, me apasiona ser capaz de mantener actualizado e implementar nuevo hardware y protocolos. Mi lista de proyectos (como la pila de cosas de mi garaje) sigue creciendo. Lo más importante es que la comunidad da mucho apoyo y es activa. Lo que permite trabajar con gente de todo el mundo, lo que es genial.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Lo que más me sorprende es el tirón que pueden tener proyectos relativamente pequeños. Hice algunas remodelaciones para aislar el control de congestiones de TCP (la parte de TCP que decide cuanto enviar de una vez). Cada módulo fue cambiando el mismo sitio en el kernel, y por sugerencia de otro desarrollador, añadí una función de interfaz virtual. Esto tuvo el efecto contrario ya que añadía más congestión de módulos de control al TCP. Esto me llevó a trabajar con investigadores de varias universidades, incluyendo un mes sabático en el Instituto Hamilton de Irlanda.

La misma interfaz fue integrada en otros proyectos. Fue portada al simulador de red usado durante el proceso de investigación. Esto permitió a los estudiantes probar algoritmos en un simulador sin tener que escribir ningún código del kernel. Los desarrolladores de BSD tomaron la misma idea.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Aunque Linux parezca una comunidad monolítica enorme, está en la actualidad dividida in muchos pequeños grupos. Involúcrate y lee las listas de correo de las áreas en las que tengas experiencia.

Como primer proyecto, elige algo que puedas probar de manera sencilla. Por ejemplo no te metas con algo como la programación o los algoritmos TCP. No te desanimes fácilmente, sigue intentándolo y aprende de tus errores.

### **¿Qué escuchas mientras programas?**

Mi viejo teléfono Nexus ha sido reciclado como una radio por internet. Mis gustos musicales son variados, desde alternativo hasta música swing, celta o incluso J Pop. Mi emisora favorita de radio es una emisora de Australia llamada Triple J porque es divertido escucharla y no tiene anuncios. O quizás porque me ayuda a pensar en la próxima Conferencia de Linux en Australia. El teléfono funciona muy bien con podcast. Un sitio destacado para operaciones con podcast por la red es "Packet Pushers".

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Solía pasar bastante tiempo en los canales IRC, pero me distraía mucho. Ahora me he unido a netdev y a otras listas de correo. Linux Plumber's fue la mejor conferencia (en lo que se refiere a tecnológica) en los últimos años. Japón también estuvo genial; Me encontré con un grupo de usuarios de Vyatta el último año.

## 23.- Glauber Costa

### Nombre:

Nací como Glauber Costa, más recientemente Lord Glommer I de Sealand, pero la gente de la comunidad me llama indistintamente Glauber, o Glommer, mi nick de toda la vida

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Mi principal tarea es asegurar de que todos se diviertan y no se tomen las cosas muy en serio. Así que soy básicamente un payaso.

Actualmente no sé cómo programar. Lo que hice fue secuestrar a una de esas personas de la comunidad que están empleados en grandes empresas y esconderle y mantenerle en mi sótano y hacer que trabaje para mí. De esta manera nadie sospecharía de mí.



Yo/él está actualmente trabajando en tecnologías de contenedores, centrándose en el control de recursos y aislamiento. Trato de evitar decir que esto se traducirá a “cgroups”, porque quiero mantener mis amigos. Pero al final así será. Durante el último año, me he centrado en el control de memoria de cgroups, tratando de extenderlo por lo que también será responsable de todo el tipo de memoria del kernel.

La compañía que me ha contratado, Parallels, tienen un solución para el contenedor bien desarrollada, que es ampliamente usada en producción (OpenVZ). Funciona en una rama del kernel de Linux y nuestra meta es implementar toda esa tecnología en la rama principal del Kernel. Yo básicamente he venido para ayudar en ese esfuerzo.

También he estado relativamente muy implicado con KVM es el pasado, debido a mi anterior empleo.

### ¿De dónde recibes tu nómina?

Del banco.

### ¿En qué parte del mundo vives, y porque allí?

Nací en Santos, Brasil, pero actualmente vivo en Moscú, Russia, que está a 10 segundos de la oficina de Parallels. Y sobre el “porque allí”, cada vez que se acerca el invierno, como ahora, me hago la misma pregunta!

### ¿Cual es tu herramienta favorita para el desarrollo de software?

Como editor de texto, utilizo Vim, y eso es todo. Ni siquiera conozco las fantásticas posibilidades que ofrece. Algunos me llaman loco por eso, pero la verdad es que no me importa mucho en absoluto.

La gente no para de alabar sus herramientas: “Te permitirán ahorrar algunos segundos, que al final del mes serán horas”. Esto es genial, y si alguien me enseña un truco con vim que pueda recordar, empezaré a usarlo, no estoy en contra de eso, ni nada por el estilo.

Pero creo que la verdadera productividad no se gana en esas cosas. Se gana en pizarras de dibujo, papeles, buena comunicación, y otras cosas por el estilo que pueden ahorrarte días o incluso meses. Así que en eso trato de centrarme.

Para el resto de tareas, creo que soy muy normal. Soy un gran fan de git, desde el inicio, pero creo que hay algunas tareas en las que estoy más cómodo usando quilt.

### **¿Y que tienes instalado en tu PC?**

Tengo instalado Fedora desde hace muchos años. Ahora mismo tengo Fedora 17. No me preocupo mucho sobre lo que tengo en el escritorio. Ni siquiera cambio mi fondo de escritorio. Lo que Fedora utilice, yo lo uso.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

Empecé en la universidad por el año 2000, y por aquel entonces no estudiaba ciencias de la computación (N.d.T: Sí ya sé que suena raro. Alguna propuesta?) Estaba en ingeniería mecánica (para ser sinceros “estudiar” no es exactamente lo que hacía) y de casualidad asistí a un curso de programación y estaba eufórico por poder escribir un programa que pudiera calcular cuando sería la Pascua, y eso fue todo

Un año después, asistí a otro curso de ciencias de la computación, de estructura de datos, y mi vecino Martim Carbone (actualmente trabajando en Vmware, qué irónico) me introdujo en el mundo de Linux. Me dijo que usándolo, haría de una manera más fácil mis ejercicios de programación. Me encantó, y nos mese más tarde me dijo que podría leer el código fuente de todo, debería querer.

No tenía los paquetes del código fuente instalados, no sabía exactamente que era eso, creía que los archivos de cabeceras era todo lo que había. Me metí en el directorio `/usr/include/linux` y empecé a leerlo. Recuerdo que no entendí nada en absoluto, pero estaba muy entusiasmado como nunca antes había estado. Puede que haya sido el primer caso en la historia de alguien desconcertado por una definición de una constante. En ese momento, yo sabía lo que quería hacer en el futuro previsible.

Empecé a leer todos los libros que conseguía. Recuerdo haber leído “Entendiendo el kernel Linux” (basado en la versión 2.4), “Desarrollo del kernel Linux” (cuando la versión 2.6 estaba a punto de salir) y también otros libros sobre la gestión de memoria interna escritos por Mel Gorman, que no recuerdo los títulos. Realmente los devoré.

En algún momento, después de suspender la asignatura de “Resistencia de Materiales” en el 5º curso, decidí que ya había tenido suficiente, y era momento de convertir mi afición en mi profesión. En paralelo a esto, empecé a trabajar para IBM LTC en Brasil en 2004 y empecé a dirigir mis acciones hacia esa meta.

### **¿Qué es lo que hace que sigas interesada en esto?**

El sentimiento de que todavía tengo cosas que aportar aquí. No podría dejarlo aunque quisiera, porque no tengo esa sensación de querer cerrarlo. Quizás algún día lo deje o quizás no.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Esta es una pregunta interesante. Es difícil acordarme de todo, pero creo que si tuviera que elegir, elegiría todas las discusiones sobre el propietario de derechos de bitkeeper y como se produjo todo

el desarrollo de git.

A mi particularmente me gusta esa historia porque muestra los puntos fuertes de nuestra comunidad: cómo de suelta es la ilusión de control que uno puede tener por obtener una licencia y como todo puede, al final, ser radicalmente cambiado por las acciones de gente que no planea conseguir ninguna visión general de esta manera (N.d.T: He tratado de traducir lo mejor que sé este párrafo pero no sé exactamente a que se refiere. Si puedes aportar algún dato que aclare o mejore la traducción bienvenido sea!!)

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Hazlo si te apetece, durante el tiempo que quieras.

**¿Qué escuchas mientras programas?**

Cuando programo, escucho a Queen. Cuando no programo escucho a Queen. Cuando programo y no programo, escucho a Queen. Y si estoy cansado de eso, entonces escucho temas en solitario de Freddie para cambiar.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

Actualmente sigo linux-mm muy de cerca, y no muchas otras. Mi nick es glommer en freenode y en oftc, pero ahora no estoy muy activo como solía ser.

Y en conferencias, eso es algo que de verdad me encanta. Suelo ir a todas las que puedo. Me gusta hablar, tanto para mis amigos como para el público. Trato de dar conferencias en cada conferencia a la que asisto. Ayuda a crear en la mente de la gente que sé de lo que estoy hablando.

## 24.- Jonathan Corbet

Es raro que alguien dentro de la comunidad Linux no conozca a Jon. Es un contribuidor del kernel, un orador habitual en los eventos de Linux y editor de la publicación más leída entre los desarrolladores del kernel, LWN.net. Jon también es coautor del informe anual de la Fundación Linux “Quién escribe Linux” y ha publicado numerosos documentos útiles, incluyendo Linux Weather Forecast y “How to Participate in the Linux Community” (Cómo participar en la comunidad Linux).

Le conozcas o no a Jon, mucho o poco, esperamos que descubras algo nuevo de él, en este retrato, desde cómo acabó en Boulder, Colorado donde hay una pista de ski que se llama como su padre, hasta que es lo que tiene instalado en su Pc o cuales son sus consejos para los nuevos que quieren involucrarse en esta comunidad.

### Nombre:

Jonathan Corbet

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Me especializo en volverme loco en internet, escribiendo y hablando sobre temas relacionados con el kernel a un público que sabe más que yo mismo. Soy el autor principal de los Drivers para dispositivos Linux, y he escrito unos cuantos drivers de video para Linux. En algunas otras ocasiones también meto mano en otras partes del kernel, por ejemplo hice una pequeña parte del trabajo para finalmente deshacerse de parte del bloqueo del kernel.



### ¿De dónde recibes tu nómina?

LWN.net es mi trabajo principal, complementado con alguna consultoría por aquí y por allá.

### ¿En qué parte del mundo vives, y porque allí?

Boulder, Colorado, EEUU. Primero vine aquí para ir a la Universidad de Colorado y nunca me pude escapar. Boulder es una gran combinación de ciudad universitaria, enclave hippy, centro de alta tecnología y un paraíso natural, se me hace muy difícil pensar en dejarlo.

Soy originario del norte de Wyoming, donde hay una pista de ski que lleva el nombre de mi padre.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Tiendo a tener un varias distribuciones diferentes instaladas por ahí así puedo echar un vistazo a lo que estan haciendo los compañeros, mi PC de escritorio actualmente está instalado Debian Testing. Mi herramienta preferida de desarrollo es sin duda Emacs, además de Claws-mail y un intérprete de Python.

### **¿Cómo te involucraste en el desarrollo del kernel de Linux?**

A principios de los 80, cuando todavía no me había graduado, tuve la oportunidad de “arreglar” un problema de la gestión de memoria en un VAX corriendo bajo una incipiente versión de BSD. Afortunadamente, el código resultante hace tiempo que se ha perdido en la memoria, pero eso me llevó a estar cerca del desarrollo del kernel desde entonces.

Configuré mi primer sistema Linux en 1993, pero fue un par de años después cuando tuve la oportunidad de mandar mi primer parche.

Pero en 1997 parecía claro que la meta de Linus (de broma) de dominar el mundo podía suceder. Así que decidí dejar mi trabajo seguro diario (que se estaba poniendo cada vez más espinoso para mi gusto) y me uní a ello.

### **¿Qué es lo que hace que sigas interesada en esto?**

Siempre hay diversión alrededor con grupos de gente inteligente, y la comunidad del kernel ciertamente lo certifica. Los retos son interesantes y la comunidad que hemos construido para dirigirlos no tiene precedentes. Mejorar la vida a un gran número de usuarios es extremadamente gratificante, pero la diversión es anterior al boom de Linux

Al final, es una aventura. Hemos cambiado el mundo, y todavía no hemos terminado. ¿Quién no quisiera ser parte de eso?

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Casi diez años después, recuerdo la debacle SCO sobre nuestra comunidad. Como grupo, estudiamos emprender acciones legales y luchar contra ese ataque en todos los frentes imaginables. Mientras tanto el código del kernel fue puesto bajo microscopio y millones de dólares se gastaron en intentar encontrar código legalmente cuestionable, pero no pudieron encontrar nada. ¿Quién hubiera imaginado que un proceso de desarrollo por separado en el que participaron miles de personas sería tan limpio?

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Lee la documentación de la rama del kernel, hay un montón de información de cómo funciona el proceso. Leer las listas de correo durante un tiempo y hacer lo posible para entender las discusiones y el código. Solucionar problemas, es la mejor manera de hacerse respetar en la comunidad.

### **¿Qué escuchas mientras programas?**

Todo tipo de cosas, pero la mayoría de cosas están dirigidas por nuestra emisora de radio local formada por voluntarios, que muy acertadamente se llama KGNU. Hay pocas emisoras de radio como esa en el mundo, que más puedo decir.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

En listas de correo estoy en linux-kernel. No soy muy aficionado a los IRC, no tengo tiempo para eso, y es muy perturbador para mi estilo de trabajo. En cuanto a conferencias: mi respuesta normal suele ser “a todas ellas” pero puede que me quede escaso para el año que viene.

## 25.- John Stultz

### Nombre:

John Stultz

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Mi trabajo con el subsistema de “timekeeping” ha sido un tema continuo a lo largo de todo el tiempo que llevo en la comunidad, pero también he trabajado en la habilitación de hardware de Linux para grandes sistemas NUMA, lo que implica trabajar en la mejora de la robustez del conjunto de parches CONFIG\_PREEMPT\_RT.

Recientemente me he estado centrando en el conjunto de parches Android, que me ha obligado a investigar en un montón de áreas del kernel a las que no había revisado antes.

### ¿De dónde recibes tu nómina?

Trabajo en el centro tecnológico de Linux en BM, actualmente estoy asignado al proyecto Linaro



### ¿En qué parte del mundo vives, y porque allí?

Vivo en Portland, Oregon (EE.UU) y me gustan mucho los barrios peatonales o en los que puedes ir en bici, la buena comida y el ambiente relajado. La escena creativa es realmente genial, y aunque la escena en cuanto a tecnología es un poco limitada, hay una buena comunidad de gente de tecnología que organizan regularmente eventos sociales (ver: calagator.org).

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

ssh, sshfs, git, kvm, no es una lista muy original. Trato de mantener mi desarrollo en un gran sistema construido en un laboratorio, y acceder a la máquina desde diferentes clientes ligeros (más conocidos como kvm entornos de escritorio de máquinas virtuales, y un netbook, esto no son medios para hacer grandes tareas de cálculo) todavía me gustan utilizar editoriales con interfaz gráfica (sóla estar encantado con nedit, pero las modernas distros de GNU/Linux dan poco soporte a esto) así que vía sshfs monto mi máquina de desarrollo que me permite editar rápidamente de manera local y hacer ciclos remotos de compilación, y testarlos con kvm.

Ah! Y una herramienta creada por Dave Jones Trinity test tool me ha sido muy útil para encontrar errores.

En cuanto al escritorio, trato de no ir contracorriente, ya que tengo pocas máquinas, y no quiero aburrirme en ajustes de entornos sincronizados. Cuanto menos configuración requiera, mejor. Así que suelo utilizar las configuraciones por defecto que traiga la distro que use. Actualmente: Ubuntu 12.04 con Unity. No puedo decir que me guste, pero me sirve.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Empecé echándole un vistazo allá por 1997, al inicio de la universidad y por aquel entonces el



compilar tu propio kernel esa una tarea comúnmente recomendada para los nuevos usuarios de Linux.

Y poco después me dedique en exclusiva a Linux, cuando asistí a más cursos de sistemas, mi interés por el desarrollo creció. Siguiendo la lista de correo lkml.org y lwn.net se hacía evidente que la comunidad de Linux era el sitio más interesante en el que desarrollar el kernel. Después de mi graduación busqué trabajos como desarrollador del kernel de Linux y empecé a trabajar en IBM's Linux Technology Center, desde donde estoy muy a gusto.

### **¿Qué es lo que hace que sigas interesada en esto?**

Me gusta mucho la naturaleza colaborativa de la comunidad. Aunque siempre existen pequeños roces, los colegas están principalmente dispuestos a ayudar, y esto da la posibilidad de que gente interesante de todo el mundo se una. Incluso si un cambio o enfoque es totalmente rechazado, siempre habrá alguien que aprenda de eso lo mejore y venga con una cosa todavía mejor. También me gusta que aunque cambies de equipos o proyectos, puedes seguir participando, ayudando y tener voz y voto como si estuvieras trabajando en ello.

El proceso de la comunidad del kernel no es definitivamente algo fácil, y hay frecuentes contratiempos, pero la mejora constante, y la resistencia férrea a cambios rápidos hace que el resultado funcione. Es un muy buen sentimiento cuando compartes algo, y sabes que los otros lo usaran y compilarán durante años.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

No puedo pensar en algo en específico si no en algo en general, creo que la actitud distendida y graciosa que muestran los tipos inteligentes es que están disfrutando mucho de lo que hacen y eso siempre es genial de ver. Por lo general, se suelen recopilar en las citas lwn.net de la semana.

¿Cual sería tu consejo para los desarrolladores que quieran implicarse?

Encuentra un área en la que involucrarte y aprende de ella.

Por encima de todo, creo que son importantes la persistencia y la empatía. Lleva trabajo entender la perspectiva de otros, especialmente si estas en desacuerdo. Evita dejar que destaque tu ego. Las discusiones no son personales (aunque suenen como tal), así que si estás molesto con algo, siéntate y contesta al día siguiente. Y espera desechar tu trabajo y empezar de nuevo, quizás varias veces.

### **¿Qué escuchas mientras programas?**

Me gusta escuchar con atención la música cuando me pongo a ello, así que puede ser más una distracción cuando estoy trabajando. Pero de vez en cuando escucho algo suave o cosas por el estilo Polica o Telephone Tel Aviv cuando estoy programando.

¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?

# linux-rt, # Linaro-kernel, lkml. Me las he arreglado para estar en Linux Conf Plumbers unos años seguidos, y una serie de conferencias Linaro Connect, pero yo no soy realmente muy constante en esto de las conferencias.

## 26.- James Bottomley

### Nombre:

James Bottomley

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Mi principal trabajo es mantenedor del subsistemas SCSI. También colaboró en arquitecturas obsoletas y poco queridas empezando por NCR voyager system, lo que me lleva a ser mantenedor de arquitecturas parisc



### ¿De dónde recibes tu nómina?

Parallels. Soy director de tecnología de virtualización de servidores, así que tengo un trabajo a jornada completa, que es estar conectado a mi kernel que funciona (Ayudo a Parallels a conducir su tecnología de contenedores en el kernel), y también abarca un amplio rango fuera de actividades con el código abierto.

### ¿En qué parte del mundo vives, y porque allí?

Londres (el que está en el Reino Unido, ya que en EEUU tienen unas copias pululando por ahí). Porque a mi mujer le gusta y es mucho más corto cuando tengo que viajar a Moscú que cuando vivía en Chicago.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Supongo que Git y emacs para el desarrollo de código. También uso Slitex o OpenOffice para las presentaciones (un día tengo que utilizar beamer para un proyecto, nunca lo he utilizado) Mi escritorio es openSUSE+Xfce.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Lo estuve siguiendo una temporada despues de graduarme en la universidad. Usábamos estaciones de trabajo SPARC que eran casi 10 veces el precio de un PC, y me figuré que si comprábamos 10 sistemas Linux por el precio de un SPARC, yo podría tener una (eso era en 1993, que utilizaba 10 discos de instalación de SlackWare)

Mi primer parche del kernel fue en los tiempos de la versión 0.99.15: Nuestro entorno home distribuido utilizaba NFS y había un error con el kernel en sistemas NFS que causaba muchos cuelgues, lo reparé con la ayuda de un colega. Después reparé el cargador del módulo del kernel del userspace para que funcionara con ELF para ayudar la a.out a la transición ELF.

Siempre he ejecutado Linux en mi sistema de escritorio (desde 1993) excepto un par de semanas en Bell Labs (donde era obligatorio Windows) y estropeaba mi sistema tan a menudo que el Administrador de sistemas de la empresa accedió a dejarme instalar Linux si dejaba de llamarle.

Mi primera incursión seria en el desarrollo del kernel fue en 1998 cuando lleve Linux a un sistema SMP Voyager reescribiendo la capa de hardware de acceso x86 para que trabajara sin APICs.

### **¿Qué es lo que hace que sigas interesada en esto?**

El hecho de que siempre hay algo interesante ocurriendo, y siempre hay alguna materia que, aunque no este relacionada con el área en la que trabajo, me puede interesar (como los Contenedores en la nube o UEFI el arranque seguro)

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

El punto fuerte y el punto débil del proceso de desarrollo colaborativo es que cualquiera puede hacerlo... incluso alguien sin ninguna experiencia (ni sentido común). Para hacer que la gente te crea debes hablar con autoridad. Solía sorprenderme por la cantidad de información incorrecta que exponían algunas personas.

Desde que tuvimos cerca un “liante” en las listas de correo SCSI donde un controlador de canal de fibra en ciernes casi se frustra por que el autor hacía caso a una de esas personas he estado mucho más vigilante para retarlos y ponerlos en evidencia.. También tengo un apartado en mis presentaciones del desarrollo del kernel sobre cómo identificar a los individuos en las listas de correo, a quienes escuchar y a quienes ignorar.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Encuentra un error que te moleste y arréglalo. De esta manera es como empezaron una gran cantidad de desarrolladores del kernel. El punto clave es tener algo que te interese. Encontrar algo que interese a alguien más es normalmente menos interesante para conseguir buenos resultados porque no conseguirás la pasión necesaria y el entusiasmo para ello. (por eso es por lo que no creo en las listas de “Cosas por hacer”... si realmente interesa a alguien, se hará)

### **¿Qué escuchas mientras programas?**

El sonido del silencio. Escucho los podcast de Radio 4 de la BBC, cuando hago trabajo que no requieren tanta atención (como aplicar parches en git) pero creo que los programas de radio requieren de mucha de mi concentración para necesitarlos mientras escribo código para solucionar problemas.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

#parisc y #storage en oftc además de las listas de correo linux-scsi en Vger. No leo ninguna de las listas de correo del kernel de Linux (LKML) así que no mandes tus parches de SCSI sólo a esas. Voy a muchas conferencias, y me he planteado seriamente dejar de ir a tantas como en los últimos cinco años... Como el número de conferencias a las que voy ha crecido... creo que necesito ayuda.

## 27.- Chris Mason

### Nombre:

Chris Mason

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Mantengo el sistema de archivos Btrfs y la mayoría de mi trabajo es dentro del campo del sistema de archivos o dentro de los caminos de entrada/salida.



### ¿De dónde recibes tu nómina?

Fusion-io.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en Rochester NY. Es donde fui a la universidad, y trabajo en Linux la verdad es que nunca he tenido que mudarme.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Uso Arch Linux, principalmente porque el modelo de actualización continua (N.d.T: rolling release) es perfecto para trabajar. Utilizo awesome como gestor de ventanas, en vez del entorno de escritorio. Es una mejor manera de manejar muchas ventanas.

Mis herramientas de desarrollo son las básicas, simplemente vi y make. Para navegar entre el código del kernel no he encontrado nada mejor que cscope, pero espero que alguien integre una base de datos actualizada del estilo de cscope en vi.

Utilizo mutt para el correo electrónico, aunque he intentado utilizar alguna que otra vez programas con interfaces gráficas. Todas tienen ventajas, pero la integración de mutt-kz de notmuch se ha convertido en una muy buena manera de indexar y trabajar con el correo.

Git es una parte importante dentro de la productividad con el kernel y no se puede subestimar como el sistema de trabajo tanto git como el kernel fueron diseñados uno para el otro. De vez en cuando, termino enseñando a alguien nuevo a manejar git y me vuelvo a dar cuenta lo complejo que son todas las partes que se manejan.

Para el análisis de sistemas de archivos, escribí una herramienta llamada seekwatcher. Utiliza partes de blktrace para visualizar que está pasando en el disco, haciendo más fácil la tarea de seguir problemas de rendimiento.

Recientemente reimplementé cosas en C en vez de en Python e hice una nueva herramienta llamada iowatcher, donde espero añadir todo tipo de funcionalidades para observar que ocurre en unidades flash.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

No fue con el desarrollo del kernel, pero en 1994 empecé ayudando con los controladores para una tarjeta gráfica no soportada. Fue genial hacer funcionar a mi hardware, y todavía tengo los CD's de SUSE con los que agradecían a los que contribuían.

Unos años después, me hice administrador de sistemas, y quise utilizar Linux en nuestro centro de datos. En ese tiempo, Linux no tenía un sistema de archivos con registro y no podía utilizar Linux en producción sin uno. Nunca antes había trabajado en almacenamiento o en el kernel, así que fue algo bastante difícil mientras me imaginaba como funcionaba todo.

Pero fue una de esas características que todo el mundo estaba deseando, y tuve mucha ilusión y ayuda durante todo el proceso. Recién terminado eso, empecé a trabajar de manera completa en el kernel.

### **¿Qué es lo que hace que sigas interesada en esto?**

Me permite interactuar directamente con los usuarios, lo que significa que tengo un “feedback” instantáneo de nuevas características y soluciones. Ir a conferencias, siempre te encuentras con nueva gente que usa o trabaja con Linux. Es una gran fuente de nuevas ideas y motivación para mejorar las cosas.

Linux es utilizado de tantas maneras que es realmente imposible aburrirse.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Las discusiones pueden parecer una locura vistas desde fuera, pero a veces son una parte importante para arreglar cosas.

Después de una larga discusión acerca de la facilidad con que se podría provocar la corrupción de datos durante un fallo de energía, me senté he hice una prueba. Los resultados nos sorprendieron a muchos de nosotros, y volví a ejecutar el mismo test años después. Probablemente no habría empleado tanto tiempo en esa prueba sin ese extra de motivación que te da el ganar la discusión.

Mi discusión favorita es sin duda la de O\_PONIES. Alguien incluso hizo una camiseta en su honor de aquello. Estoy seguro de que no haría justicia si la cuento aquí, pero es un gran ejemplo de cómo los compromisos que tomamos como desarrolladores pueden crear expectativas imposibles en el futuro.

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Mi primera sugerencia es coger un proyecto que te guste usar. El kernel tiene una dura curva de aprendizaje, y puede ser muy difícil trabajar a tu manera dentro de un grupo de desarrollo. Si te gusta lo que haces, será más sencillo unirse.

Después de eso, solucionar problemas es la manera más rápida para conocer el código. Elige algo en lo que puedas aportar con facilidad y será más fácil el conseguirlo. Así que hazlo una y otra vez hasta que conozcas el código suficientemente como para revisar parches de otros. Entonces ya conocerás bien como funciona la comunidad y podrás trabajar en cualquier campo.

### **¿Qué escuchas mientras programas?**

Normalmente me gusta que la habitación este en silencio.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

En el canal #btrfs en Freenode. Y en cuanto a conferencias, la Linux Filesystem, Storage y MM Summit cada primavera siempre hay algo con temas interesantes. La Fundación Linux constantemente hace un gran trabajo en todas sus conferencias, y trato de asistir siempre que me lo permite mi agenda.

## 28.- Herbert Xu

### Nombre:

Herbert Xu

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Co-mantengo los sistemas criptográficos. También participo en el desarrollo de la pila de red (network stack)

### ¿De dónde recibes tu nómina?

Red Hat Inc

### ¿En qué parte del mundo vives, y porque allí?

Shangai, China. Es un gran lugar para estar. Y también alberga una comunidad activa de Linux.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Mi herramienta favorita de desarrollo es elvis, un clon del editor vi. Tengo instalado Gnome en mi Pc de escritorio, pero casi siempre tengo una terminal ejecutando screen. (N.del T.Unos enlaces útiles para esta herramienta pueden ser estos: <http://www.slideshare.net/jza/aprendiendo-linux-screen>, <http://rz0r.blogspot.com.es/2007/12/screen-otra-util-herramienta-en-linux.html>, <http://linux.die.net/man/1/screen>)



### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Empecé como mantenedor del kernel de Debian en 1997, y ahí empezó todo.

### ¿Qué es lo que hace que sigas interesada en esto?

Estar a la vanguardia de la tecnología siempre me emociona.

### ¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?

Cómo la gente trabaja todo el día en un solo problema y todavía son capaces de mantenerse fresco gracias a los husos horarios.

### ¿Cual sería tu consejo para los desarrolladores que quieran implicarse?

Mi consejo es el mismo consejo que dió Jay Leno para conducir en China: “Se precavido, pero mantente firme”.

### ¿Qué escuchas mientras programas?

Silencio

### ¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?

#netdev  
LinuxCon

## 29.- Frédéric Weisbecker

### Nombre:

Frédéric Weisbecker

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

Eso tiende a evolucionar con el tiempo. He estado trabajando en el seguimiento con los eventos ftrace y subsistemas perf. Luego pasé algún tiempo ayudando a eliminar el bloqueo del gran kernel. Mientras contribuyo con algunas limpiezas aquí y allá. Actualmente estoy trabajando sobre todo en el soporte a dynticks.

### ¿De dónde recibes tu nómina?

Red Hat. Red Hat es para mí realmente una gran compañía para trabajar porque, además de asignarme tareas específicas sobre el kernel de Red Hat, también me dan la libertad y el tiempo necesario para trabajar en temas como dynticks.



### ¿En qué parte del mundo vives, y porque allí?

Al noreste de Francia, porque mi familia y amigos están aquí.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Nada extravagante en lo referente a las herramientas que uso: git, editores de texto y scripts. En mi PC principal uso Fedora, también Xubuntu en PC's de prueba y en otro portátil.

### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

Un amigo mío instaló Linux en su PC, hará unos 9 años. Yo ni siquiera estaba metido en la programación por aquel entonces, pero coincidió en la época en la que me empecé a preguntar cómo se hacía el software. Eso fue cuando apenas tenía “veintipocos años” Empecé a profundizar cada vez más, pasando desde programación web (PHP) a Python o C o ensamblador X86. Entonces me encontré con el límite del núcleo.

Desarrollé una gran curiosidad de cómo el software se comunica con el hardware. Y encontré respuestas detalladas de que eso era difícil y complicado. Leí “Linux Device Drivers 3ª Edición” varias veces. Pero verdaderamente le metí mano al asunto unos años después. Tenía una tarjeta Wifi Atheros y el driver (ath5k) que no estaba preparado del todo. Quise ayudar a su desarrollo, así que intenté usar mmioTRACE para hacer un seguimiento de las E/S hecho por el driver propietario. Había sólo una característica que faltaba para insertar mensajes del usuario en el medio de la trama para marcar algunos pasos. Así que hackee eso con ftrace y finalmente conseguimos descifrarlo.

### ¿Qué es lo que hace que sigas interesada en esto?

Trabajar con gente muy capaz. Debatir nuevos diseños. Y tener una gran base de usuarios. También como el tiempo pasa, me preocupo más de consolidar el trabajo. Lo mejor viene cuando todo termina con un diffstats negativo. Esta es una manera de retrasar un poco la evolución de la cuestión final: la implosión del núcleo bajo su complicación cada vez mayor.

### **¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

Me he divertido muchas veces con los parches que se mandan el día 1 de Abril (N.d.T: Día en que se gastan bromas, parecido al día 28 de Diciembre) Aquí puedes ver unos ejemplos.:

<http://thread.gmane.org/gmane.linux.power-management.general/15888> o

<https://lkml.org/lkml/2012/3/31/131>

### **¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Una buena cosa sería empezar resolviendo problemas específicos que uno tenga: un bug que cuelga tu máquina, o un driver que falta. Ahora esa clase de cosas es difícil de encontrarlas en estos días en que cada vez hay más hardware soportado.

Otra manera es revisar bugs reportados a las listas de correo del kernel, o tratar de investigar y testear los parches de otros usuarios.

También revisar nuevos desarrollos a parches posteados en las lista de correo del kernel (esos normalmente e listan aquí). Después aplicar esos parches, ejecutarlos, y reportar errores que encuentres. Una vez que te familiarices con este proceso, puedes intentar a resolver problemas por ti mismo. Familiarízate con un subsistema y que se necesita hacer allí.

Este proceso puede llevar un tiempo y hay que ser paciente y tenaz, pero merece la pena.

### **¿Qué escuchas mientras programas?**

La verdad es que no puedo escuchar música mientras programo, a menos que este haciendo algo de manera mecánica. Pero escucho mucha música y me gustan muchos estilos en general. Tengo una especial afinidad con el Metal y la música electrónica.

Y respecto al Metal me gustan: doom, black o gótico. Por citar alguna bandas: My Dying Bride, Verdunkeln, On Thorns I lay, Therion. Después cuando la sobredosis de metal me produce dolor de cabeza, cambio a The Gathering, Emiliana Torrini, NIN, Hooverphonics y más bandas que no tienen nada que ver entre sí.

### **¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

LKML. Y también suelo pasar el tiempo conectado a #linux-rt . En cuanto a conferencias, depende. Isuelo ir a Real Time Linux Workshop, algunas veces a Linux Plumbers Conference, Linux Foundation Collaboration Summit, y LinuxCon. Y normalmente no me pierdo Chemnitzer Linux Tage.



## 30.- Paul McKenney

### Nombre:

Paul McKenney

### ¿Qué papel desempeñas dentro de la comunidad y en que subsistemas trabajas?

La actualización de copia de lectura (RCU) es mi pequeño; aunque ocasionalmente cojo otras partes del kernel y en otros proyectos de modo usuario, incluyendo userspace RCU y “Es programar en paralelo difícil, y si lo es, ¿que puedes hacer al respecto?”

### ¿De dónde recibes tu nómina?

IBM Linux Technology Center financia mi adicción al desarrollo del kernel.

### ¿En qué parte del mundo vives, y porque allí?

Vivo en Beaverton, Oregon, EEUU.

Una razón es que tanto mi mujer como yo crecimos en Oregon. Otra razón es que Beaverton fue donde se encontraba Sequent Computer System, y fueron lo que me volvieron a contratar en 1990 cuando llegó el tiempo de regresar a casa desde Bay Area.

### ¿Cual es tu herramienta favorita para el desarrollo de software? ¿Y que tienes instalado en tu PC?

Las herramientas normales de producción: git, cscope, vi, awk, y para la pelea con python. ¿Por que vi? El sistema compartido que he estado usando hace 30 años, puede soportar siete u ocho sesiones concurrentes de vi, pero sólo una sesión de emacs. En ese entorno, por lo tanto, el uso de emacs era socialmente irresponsable.

En este momento utilizo Ubuntu con Unity2D.

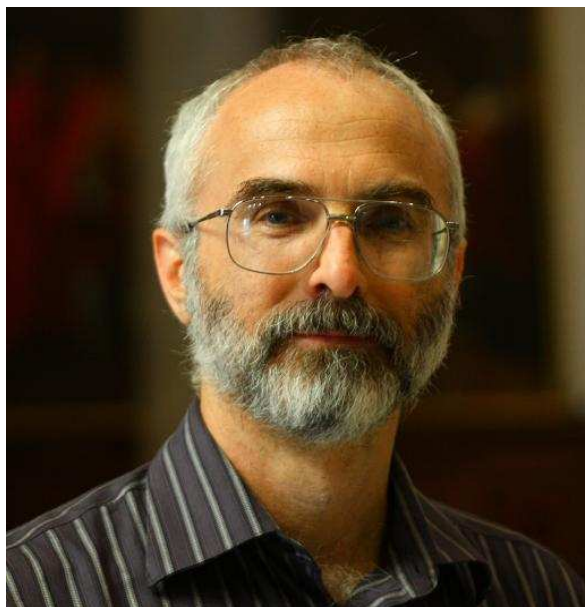
### ¿Cómo te involucraste en el desarrollo del kernel de Linux?

La primera vez fue en 1997, cuando estaba trabajando en DYNIX. Tuve un correo de alguien con una dirección .ru preguntando por una máquina lectora de copia de un viejo documento mío. Le mandé el documento y le pregunté para que lo iba a utilizar. Me contestó que estaba trabajando en un kernel llamado “Linux”. A pesar de que había oído hablar de Linux, sería algunos años antes de aprender el significado del nombre “Alexey Kuznetsov”.

La segunda vez fue en 2000, cuando me incorporé a la IBM Linux Technology Center.

### ¿Qué es lo que hace que sigas interesada en esto?

El desafío constante de mantenerse al día con lo que para que está la gente usando el kernel de Linux. La necesidad de escalabilidad SMP, respuesta en tiempo real, la huella de memoria pequeña (small memory footprint), y la eficiencia energética (por no hablar de la fiabilidad necesaria para soportar millones de dispositivos) ha dado lugar a una larga serie de problemas muy interesantes



para resolver.

**¿Qué es lo más gracioso o curioso que te ha sucedido durante el proceso de desarrollo colaborativo (discusión encarnizada, petición de un código ridículo, un logro increíble)?**

A menudo me sorprende cuando una sola solución técnica solucionan problemas que aparentemente no tienen nada que ver en absoluto unos con otros. La primera vez fue con CONFIG\_NO\_HZ siendo requerido por la alimentación de la batería por un lado y los mantenedores principales por otro: Lo más pequeño de los más pequeño y lo más grande de los más grande. Después me sorprendí de cómo el -rt patchset fue efectivo en encontrar errores en SMP. Más recientemente, un parche en el que estoy trabajando para las comunidades de HPC y real-time puede también ser efectivo para la gestión eficaz de la energía.

**¿Cual sería tu consejo para los desarrolladores que quieran implicarse?**

Leer todos los grandes consejos de los otros 29 desarrolladores anteriores. No se me ocurre nada que decir que no hayan dicho ya.

**¿Qué escuchas mientras programas?**

Escucho música contemporánea. Si han pasado más de tres o cuatro siglos, ya tengo una difícil relación con ella.

**¿En qué lista de correo o en qué canal IRC puede encontrarte la gente? ¿En que conferencias?**

En cuanto a listas de correo, LKML y linux-rt-users, y canales de IRC #linux-rt. Dicho esto, no puedo decir que realmente este al día con alguno de ellos.