

GNU/LINUX DESDE LA CONSTRUCCIÓN SOCIAL DE LA TECNOLOGÍA

GNU/LINUX WITHIN THE FRAMEWORK OF SOCIAL CONSTRUCTIVIST OF TECHNOLOGY APPROACH

Sofía Flores Solorzano*
sofia.flores@ucr.ac.cr

Fecha de recepción: 2 octubre 2009 - Fecha de aceptación: 19 enero 2009

Resumen

El sistema operativo GNU/Linux es un producto tecnológico con un modelo de desarrollo ligado a una intensa interacción social. Su estudio ha abordado desde la tradición interpretativa del desarrollo tecnológico y desde el modelo de la construcción social de la tecnología (COST), demuestra que los grupos que desarrollan GNU/Linux (comunidades, iniciativas ciudadanas, empresas) son «grupos sociales relevantes» los cuales han conseguido generar un modelo de uso social de la tecnología al servicio de una cultura de la información. Otras características como su flexibilidad interpretativa (ej. desarrollo del kernel y compilación, existencia de cientos de versiones, etc.), mecanismo de cierre (es un producto estable, pero no cerrado en sus versiones estable e inestable) y marco socio-tecnológico (división en usuarios estratégicos y tácticos) sitúan a GNU/Linux como un ejemplo actual de cómo la tecnología se construye socialmente.

Palabras clave: cultura de la información, construcción social de la tecnología, GNU/Linux, software libre, código abierto.

Abstract

GNU/Linux development is particular for their intense social interaction. The sociology behind this product is analyzed here in terms of social construction of technology (SCOT). Having succeeded in the social use of technology, GNU/Linux developers constitute relevant social groups in the service of the information culture. Concepts closely linked to SCOT are the interpretative flexibility (demonstrated by hundreds of versions of the kernel and the ways to compile it), the existence of a closure mechanism for this technology (stable product in continuous innovation available through their stable and unstable versions) and a socio-technological framework (given by the division in tactical and strategical users), place GNU/Linux as a modern example of how technology is socially constructed.

Key words: information culture, social construction of technology, GNU/Linux, free software, open source.

* Maestría en Comunicación, Sede Rodrigo Facio, Universidad de Costa Rica.

Introducción

El particular proceso de producción del *software* libre (la historia del *software* libre está ligada a Richard Stallman (Manhattan, EEUU, 1953), uno de sus principales activistas. Stallman ha contribuido a conformar las bases técnicas, legales, éticas y políticas del movimiento. Ha desarrollado varios programas y herramientas del proyecto GNU, por ejemplo, su conocido editor *emacs*, además, es autor de la licencia GPL *General Public License*. Stallman ha escrito diversos ensayos y ha dado centenares de conferencias por todo el mundo. Para una recopilación del pensamiento de Stallman ver «Software libre para una sociedad libre» (Stallman, 2004). Asimismo, es el fundador y actual presidente de la *Free Software Foundation* (FSF) y del proyecto GNU) y código abierto, (en 1998 el movimiento del *software* libre se divide debido a puntos de vista distintos en relación con el concepto de libertad, concretamente por la confusión que se daba al relacionar el término *free* con la idea de que el *software* libre podía ser gratis, así como por discrepancias con la licencia GPL *General Public License*, que protege al *software* libre de cualquier intento de apropiación) basado en la cooperación y colaboración voluntaria gracias al soporte comunicativo brindado por Internet, ha generado un gran interés desde diversas disciplinas y perspectivas. Lo más destacable de ambos modelos es que se caracterizan por no restringir el acceso al código fuente, permitiendo su estudio, modificación y redistribución. Sin embargo, hay diferencias ideológicas entre ambos. El código abierto es desde hace varios años, sobre todo, un nuevo modelo de negocio el cual ha introducido nuevas estrategias (por ejemplo, la innovación cooperativa) para competir con los grandes monopolios de la industria del *software*; mientras que el *software* libre es un modelo de desarrollo más interesado en los aspectos comunitarios y filosóficos del *software*, no tanto en convertirlo en una oportunidad de negocio.

La comunidad de investigadores del Instituto Tecnológico de Massachusetts³ (MIT *Massachusetts Institute Technology*; sitio oficial: http://opensource.mit.edu/online_papers.php) es un referente en la investigación del *software* libre

y código abierto. El MIT emplea la sigla FLOSS para referirse a ambos modelos de producción de *software*. FLOSS significa en inglés *Free-Libre Open Source Software*, se trata de un término neutral para referirse tanto al *software* libre como al código abierto. Por convención, en este trabajo, se utilizar las siglas FLOSS al referirse a ambos.

Es importante destacar que GNU/Linux es el nombre con el cual se denomina el sistema operativo compuesto por el núcleo o *kernel* Linux más las herramientas proporcionadas por el proyecto GNU. GNU/Linux es el proyecto más representativo e importante en el mundo del FLOSS, ver «The Cathedral and the Bazaar», clásico ensayo de Raymond (1998), el autor estudia la comunidad Linux y populariza la alegoría de la catedral como la imagen de la compañía transnacional de *software* propietario y el bazar como metáfora de la comunidad de desarrolladores de Linux. GNU es un acrónimo que significa *GNU's Not UNIX*, es un proyecto que pertenece a la *Free Software Foundation* (FSF) cuyo objetivo es desarrollar un sistema operativo completamente libre basado en UNIX. La FSF tiene el *copyright* de una gran proporción del sistema operativo GNU/Linux, así como de una cantidad importante de *software* libre.

Linux es el nombre del núcleo del sistema operativo que se atribuye a Linus Torvalds, este último era un estudiante de informática de la Universidad de Helsinki a principios de los noventa, quien usaba un sistema basado en Unix llamado Minix, e intentaba implementar este sistema en una máquina 386 utilizando el lenguaje C y varias herramientas del proyecto GNU. El resultado de este proyecto fue la primera versión de Linux, una versión muy primitiva que no llegaría a difundirse. Sin embargo, más tarde, el 5 de octubre de 1991, en una conferencia sobre Minix, Linus distribuyó gratuitamente entre los asistentes el código fuente de la versión 0.02 de Linux e invitó a todos a participar activamente en el proyecto. Posteriormente, registra Linux bajo la licencia GPL *General Public License*, promovida por el proyecto GNU. La combinación de Linux más todas las herramientas GNU dio como resultado un sistema operativo muy robusto y completo.

El *kernel* Linux es, por así decirlo, solo el motor del sistema operativo, un componente clave, pero no el único. Para obtener un sistema completo y funcional hace falta tener más componentes. Estos vienen integrados junto con un *kernel* genérico en lo que se conoce como una distribución; es decir, un sistema GNU/Linux completo. De ahí que muchos usuarios y desarrolladores del nuevo sistema prefieran denominarlo GNU/Linux. Las distribuciones GNU/Linux contienen miles de herramientas y aplicaciones, algunas desarrolladas por el proyecto GNU, y otras por otros proyectos de FLOSS como KDE, Gnome, Firefox, Apache, etc. Hoy, existen numerosas distribuciones que se pueden desgargar desde Internet o bien adquirir en algún soporte físico. Detrás de cada distribución, hay un grupo social relevante, ya sea una comunidad o una empresa, y, en consecuencia, una determinada visión tecnológica.

El objetivo de este estudio es analizar el sistema GNU/Linux según las categorías que demarca la construcción social de la tecnología (Pinch, 1997) y el modelo de innovación socio-tecnológico propuesto por Flichy (2003), a fin de comprender el desarrollo de este producto tecnológico.

He sido usuaria de varias versiones GNU/Linux, conocidas como distribuciones: *Debian*, *Ubuntu*, *KnoPIX*, *Mandriva*, *OpenSUSE*, *Fedora*; entre otras, con el fin de conocer las características y las dificultades que entraña el uso de este sistema operativo y las aplicaciones de FLOSS que integra cada sistema. De esta manera, he podido aprovechar mi propia experiencia para analizar el paradigma tecnológico en el cual se basa. Paralelamente, durante más de dos años (de enero del 2006 a julio del 2008), como parte de mi tesis de maestría (Flores, 2009a, 2009b), investigué las comunidades de FLOSS de Costa Rica.

Paradigmas de la innovación tecnológica

El libro de Kuhn, «The Structure of Scientific Revolutions», publicado en 1962, influyó de manera decisiva en la revisión de los estudios sociales de ciencia y tecnología llevada a cabo a partir de los años sesenta: filósofos, historiadores

y sociólogos constructivistas reconocen que son herederos del pensamiento de Kuhn (Cutcliffe, 2003, pág. 28-29). Kuhn criticó las nociones de objetividad dentro del pensamiento científico, sugirió que el conocimiento científico evolucionaba bastante más discontinuamente de lo que había sugerido el modelo positivista. Para Kuhn, la ciencia está organizada alrededor de lo que denominó paradigmas o patrones de organización de creencias y prácticas (Kuhn, 1971).

¿Qué es un paradigma tecnológico? Según Dosi (1982) y Díaz y Lee (1992), el desarrollo tecnológico es el resultado de varias tradiciones y problemas científicos y técnicos. Para estos autores, los artefactos técnicos se encuentran dentro de paradigmas tecnológicos; por eso, resultan relevantes determinados problemas relacionados con el conocimiento, los recursos con los cuales se cuenta y la visión de los actores con respecto a la tecnología. Díaz y Lee (1992) distinguen dos tradiciones fundamentales dentro de los procesos de innovación tecnológica: la tradición instrumental de la innovación tecnológica (TIIT) y la tradición interpretativa (TIN).

La tradición instrumental de la innovación tecnológica surge aproximadamente en los años sesenta y agrupa diferentes teorías y modelos que comparten una concepción similar del fenómeno de la innovación tecnológica: básicamente la tecnología es concebida como una especie de piedra que impacta en la sociedad. De ahí que se hable del impacto de la tecnología, como si la tecnología estuviera en un lado y la sociedad en otro.

La concepción instrumental, promovida por la teoría económica de la innovación tecnológica, conduce generalmente a definiciones deterministas de los procesos de desarrollo tecnológico. Este se mide en función de su explotación comercial, y en definitiva del éxito económico del nuevo artefacto, para ello se insiste en la necesidad de ejercer un control sobre el ambiente, control que descansa obviamente en la tecnología (Díaz y Lee, 1992).

En contraposición, **la tradición interpretativa** (TIN) pertenece a los estudios sociales de la ciencia y la tecnología que rompieron con las narrativas optimistas de la tecnología, y con la tradición mertoniana de la sociología de la

ciencia, y se propusieron abrir las «cajas negras» a partir de la década del 70 (Díaz, 2003).

La sociología del conocimiento científico o sociología de la ciencia ha adoptado a lo largo de los años diferentes rutas empíricas y teóricas de investigación, los tres modelos más importantes son: la construcción social de la tecnología (Bijker, Hughes, y Pinch, 1987), la teoría de la red de actores (Callon, 1987, Latour, 1991, 1992) y los sistemas socio-técnicos (Hughes, 1983, 1987).

Las tres posturas mencionadas comparten lo que Díaz (2003) llama una «metáfora fundacional», pues las tres consideran que la relación entre tecnología y sociedad es parte de un tejido sin costuras (*seamless web*). Esta perspectiva cuestiona la imagen de la tecnología como una piedra lanzada desde el exterior hacia la sociedad y no acepta el surgimiento de la tecnología como un ente externo o ajeno a la sociedad, la política y la economía.

En lugar de medir el desarrollo tecnológico en términos de éxito o fracaso, la tradición interpretativa (TIN) valora los procesos y sus participantes. Esta está más interesada en la comprensión y en la interpretación de la actuación de los grupos involucrados en el proceso de innovación que en determinar una secuencia de etapas sucesivas. Además, cuestiona la idea de la tecnología como un resultado y un producto acabado. La concepción del proceso de innovación es multidireccional y multidimensional y no está constituido en etapas, sino que puede estar conformado por un grupo social relevante, una red de actores o sistemas tecnológicos.

Análisis de GNU/Linux desde la construcción social de la tecnología

De acuerdo con Pinch (1997), la propuesta de la construcción social de la tecnología, conocida por sus siglas COST, continua siendo un modelo de análisis de la tecnología vigente y productivo, pues ofrece un marco conceptual para la comprensión de la tecnología y la realización de estudios de caso de innovación tecnológica, todo ello a pesar de las críticas a las cuales ha sido sometida la propuesta (jerga oscurantista,

planteamiento formulista, poca atención a los usuarios y descuido del contexto político).

Las categorías de análisis propuestas por la construcción social de la tecnología y que a continuación se utilizarán en el caso de GNU/Linux son las siguientes: grupo social relevante, flexibilidad interpretativa, mecanismo de cierre y marco tecnológico.

La comunidad de GNU/Linux como ejemplo de grupo social relevante

Un grupo social relevante es sencillamente un grupo de individuos organizados o no quienes están asociados por y para el desarrollo de un artefacto tecnológico y comparten un significado común con respecto a dicho artefacto. En palabras de Pinch (1997):

«Cada grupo social relevante para un artefacto técnico se demarca a partir de los problemas concretos que plantea, de las distintas soluciones que propone a estos problemas, así como de los significados compartidos que atribuyen al artefacto en cuestión (...) Esta categoría se utiliza para denotar instituciones y organizaciones, así como grupos de individuos organizados o no organizados» (Pinch, 1997, pág. 54).

El concepto de grupo social relevante tiene la ventaja de ser lo suficientemente abstracto y abierto, como para permitirnos comprender el sistema social asociado al FLOSS y concretamente al proyecto GNU/Linux. Estos tienen su propia visión acerca del significado, uso y desarrollo de GNU/Linux. En ese sentido, se debe recordar que inicialmente fue la comunidad *hacker*, Sobre los *hackers* como movimiento contestatario pueden consultarse los planteamientos de Levy (2001), Himanen (2001), Lancashire (2001) y Lizama Mendoza (2005), el grupo social relevante encargado del desarrollo de las herramientas que integra el proyecto GNU y el *kernel* Linux; sin embargo, desde hace varios años se asistió a un proceso de apropiación por parte de otros grupos sociales relevantes. Como se ha comentado anteriormente, dentro de la industria del *software*, empresas como Cygnus Solutions, RedHat, SUSE, etc., han encontrado una oportunidad de negocio en la innovación cooperativa y en la arquitectura de participación.

Dentro de este proceso de capitalización de GNU/Linux, existen algunos grupos sociales relevantes que conservan una visión social y política del desarrollo y uso del *software*. Se habla sobre comunidades de desarrollo como Debian.org, integrada en su gran mayoría por *hackers*. Además, han surgido importantes grupos sociales relevantes en la ciudadanía, como en el caso de la Junta de Extremadura (España) que han desarrollado su propia distribución conocida como GNU/LinEx.

Las distribuciones GNU/Linux desarrolladas por las comunidades son, a diferencia de las distribuciones comerciales, productos en los cuales el soporte técnico lo brinda fundamentalmente la comunidad de usuarios o grupo de usuarios de GNU/Linux. Además, es importante destacar que, en estos casos, los usuarios también intervienen activamente en el desarrollo, lo anterior debido a que se encargan de probar las llamadas versiones beta o inestables. De este modo, su papel es crucial, así en algunas versiones de GNU/Linux; por ejemplo, GNU Debian, si el usuario está de acuerdo, automáticamente el sistema le envía a los desarrolladores los fallos registrados, también muchos usuarios envían voluntariamente a las listas de correo habilitadas para ello, los *bugs* o “bichos” que detectan (en la jerga informática un *bug* es ‘un defecto en el código o en la rutina de un programa’). Además, hay usuarios que, aunque no son desarrolladores oficiales, colaboran esporádicamente en tareas concretas como depuración de código y redacción de documentación.

Asimismo, un grupo social relevante puede negociar con otros grupos (organizaciones, instituciones o empresas) el significado de un artefacto tecnológico y, en consecuencia, también su marco tecnológico, lo cual implica aspectos relacionados con su desarrollo y su uso. Es el ejemplo de la estrategia empresarial que consiste en negociar con las comunidades de desarrollo el marco tecnológico de GNU/Linux, esto las lleva a colaborar activamente con las estas, incluso financiando determinados proyectos.

El caso más paradigmático es la compañía Red Hat que comercializa la distribución Red Hat Enterprise Linux (RHEL), dirigida fundamentalmente para servidores y para el

mundo empresarial, y la distribución no comercial Fedora, que es su distribución de pruebas y; por lo tanto, gratuita, de la misma compañía. La distribución Fedora es libre, tiene el soporte de la comunidad Fedora, se considera más inestable y recomendada solo para usuarios estratégicos. En cambio, Red Hat Enterprise Linux (RHEL) es la versión comercial optimizada de Fedora que Red Hat distribuye con un certificado de calidad y con soporte técnico.

El modelo de desarrollo utilizado por Red Hat *a grosso modo* es el siguiente: contrata a 4 o 5 ingenieros con buenos sueldos, cuya función consiste en colaborar con la comunidad Fedora, esta última es el grupo social relevante que produce el nuevo código. Red Hat toma el código fuente generado por la comunidad y a continuación crea una versión compilada o binaria (RHEL), la cual prueba y optimiza para un determinado tipo de *hardware*, generalmente servidores. Luego, Red Hat ofrece a sus clientes esta distribución binaria a través de un acuerdo de suscripción, en la que el cliente queda obligado a pagar anualmente por el soporte técnico. Es importante aclarar que Red Hat y otras empresas distribuidoras de GNU/Linux, como SUSE, no venden una licencia, sino una suscripción y el código fuente se puede obtener de manera gratuita. El modelo de Red Hat es, sin lugar a dudas, un modelo exitoso y ha sido copiado por otras empresas, las más conocidas comercializan las distribuciones Mandriva, Yellow Dog Linux, CentOS, entre otras.

Desde el punto de vista de la construcción social de la tecnología (COST), las comunidades, empresas o iniciativas ciudadanas que desarrollan sistemas GNU/Linux son grupos sociales relevantes. Sus miembros se organizan en torno a este sistema operativo y comparten un significado común de su uso y desarrollo. Asimismo, estos grupos sociales relevantes se han caracterizado por una gran capacidad de innovación. Por ello, desde hace varios años, GNU/Linux ha dejado de ser visto como un producto para *geeks* de la informática, y constituye, en este momento, una clara estrategia de negocios para grupos sociales relevantes constituidos por empresas (Red Hat, SUSE, Novell, Dell, Asus, etc.), pero también es una clara estrategia política para los gobiernos, instituciones y organizaciones sociales en todo

el mundo, los cuales han decidido migrar a este sistema en aras de procurar el acceso de los ciudadanos a la información mediante estándares abiertos y *software* libre.

Flexibilidad interpretativa

Un concepto clave de la COST es el de flexibilidad interpretativa, esta noción ha sido utilizada en múltiples estudios constructivistas, radicales y moderados, básicamente consta de dos principios:

1. Cualquier artefacto técnico está sujeto a una pluralidad de interpretaciones.
2. No hay un solo camino posible ni una forma correcta para diseñar los artefactos técnicos.

La flexibilidad interpretativa se refiere a los significados radicalmente nuevos y diferentes que son dados a los artefactos técnicos por distintos grupos sociales, como señala Pinch (1997) «de la pluralidad de interpretaciones pueden resultar aplicaciones o usos de los artefactos muy distintos a los que estaban ceñidos por una normativa original».

Este concepto se puede aplicar tanto al desarrollo como al uso estratégico de cualquier artefacto tecnológico. En ese sentido, el sistema GNU/Linux se caracteriza por su permanente cambio y reinterpretación, pues, al ser libre y abierto, cualquier usuario con los conocimientos técnicos suficientes puede adaptarlo a sus necesidades o a las de su negocio.

Por ejemplo, el desarrollo del *kernel* es un caso de flexibilidad interpretativa. El grupo social relevante encargado del desarrollo de este es la comunidad *Kernel.org* (<http://www.kernel.org>). El proyecto cuenta con varias listas de correo a las que cualquier usuario o desarrollador puede enviar sus mejoras o informes. Esta manera de proceder no significa que cualquiera pueda cambiar el *kernel* Linux solo con enviar sus propuestas, pues aunque en la actualidad el desarrollo de este es un proyecto abierto y participativo, es dirigido por Linus Torvals, quien decide en última instancia qué mejoras o nuevas funcionalidades se incorporan y cuáles no. Es

decir, Linus Torvals es el encargado de negociar con los cientos usuarios estratégicos que participan en el proyecto, el marco tecnológico del *kernel* Linux.

Otro ejemplo de la flexibilidad interpretativa que ofrece GNU/Linux se refleja en su capacidad para adaptarse a un determinado *hardware*, es lo que se conoce como la compilación del *kernel*. Muchos usuarios avanzados o estratégicos prefieren compilar su propio *kernel* en lugar de usar un *kernel* genérico; de esta manera, consiguen el máximo rendimiento de sus computadoras y un mejor aprovechamiento de los recursos particulares del *hardware*.

Desde la perspectiva de la construcción social de la tecnología, la multiplicidad de significados que ha suscitado GNU/Linux es un ejemplo paradigmático de flexibilidad interpretativa. Resulta prácticamente imposible y desbordaría los objetivos de la presente investigación intentar referirse a todas las versiones de GNU/Linux que existen en la actualidad, pues hay cientos de ellas, algunas mantenidas por grupos y otras por una sola persona (para conocer más acerca de las distribuciones más populares, recomiendo el portal <http://www.linux.org>).

Mecanismo de cierre

Se refiere al momento cuando una de las interpretaciones de la tecnología se impone sobre el resto y; por lo tanto, se da por concluida y cerrada. A veces, existe una negociación entre los grupos sociales relevantes para determinar cuando un artefacto técnico se estabiliza; aunque no siempre ocurre así, pues puede darse el caso de que una solución se haya impuesto sobre otra. Esta noción no debe interpretarse como una acción definitiva, al contrario, una tecnología una vez cerrada puede ser reabierta y particularmente adoptada y usada por diferentes grupos sociales.

Así, GNU/Linux es una versión libre y mejorada de UNIX. Mediante un procedimiento conocido como ingeniería inversa, los usuarios estratégicos han construido GNU/Linux a partir de UNIX. GNU/Linux al igual que su predecesor se caracteriza por ser un sistema multiusuario, multitarea, multiplataforma y con gran capacidad para la gestión de redes.

UNIX era un producto terminado y, de esta forma, cerrado a principios de los años ochenta. Sin embargo, fue reabierto por distintos usuarios, entre los más conocidos, Richard Stallman, Andrew Tanenbaum y Linus Torvals. ¿Está cerrado GNU/Linux en la actualidad? Se considera que no porque, aunque es un producto estable y competitivo, está siendo continuamente reabierto por los usuarios estratégicos y desarrolladores de todo el mundo. En ese sentido, su modelo de desarrollo tiene un intenso ritmo de innovación que pasa por tres etapas conocidas como las versiones estable, en pruebas e inestable.

La versión estable, como su nombre lo indica, es la recomendada para equipos de trabajo o producción, así como para usuarios tácticos. Es la versión más probada y segura de todas. **La versión en pruebas** es aquella que pasará en breve a estable, pero que todavía está en período de pruebas hasta estabilizarse, es la versión candidata para convertirse en estable. **La versión inestable** es donde tiene lugar el desarrollo, es la versión usada solo por los desarrolladores o los usuarios estratégicos quienes colaboran en el desarrollo. Es la versión más innovadora de todas, aunque no todas las nuevas funcionalidades y mejoras se incorporan en la siguiente versión.

Marco socio-tecnológico

Dentro del desarrollo teórico de la COST, Bijker (1987) propuso la noción de marco tecnológico. Un marco tecnológico es el marco de significados relacionados con una determinada tecnología, compartido entre varios grupos sociales, que guía y da forma al desarrollo de los artefactos. En palabras de Pinch (1997), « (...) está compuesto por los conceptos y técnicas que una comunidad emplea para la solución de sus problemas; es una combinación de teorías aceptadas, conocimientos tácitos, prácticas de ingeniería, procedimientos especializados de experimentación y prueba, objetivos y manejo y uso de prácticas».

Este concepto permite establecer un vínculo entre la sociedad en la cual se encuentra inmersa la tecnología y su trayectoria de desarrollo (Pinch, 1997, pág. 27). Dicha noción ha sido

retomada y ampliada por Flichy en su trabajo «L' innovation technique» (2003), y es la que se utilizará para nuestro análisis.

Flichy (2003) propone una nueva aproximación para el análisis de la innovación tecnológica. Su principal categoría es el llamado marco de referencia socio-tecnológico, muy similar al marco técnico o esquema tecnológico propuesto por Bijker (1987). El marco socio-tecnológico de Flichy está conformado por dos partes complementarias:

- El marco de la función.
- El marco del uso.

El marco de la función se relaciona con el uso técnico mientras que el marco del uso con el uso social. Así, define los conocimientos necesarios para «abrir la caja negra» y eventualmente modificar un aparato técnico, es el marco que comparten los diseñadores, técnicos y los usuarios estratégicos. Mientras que **el marco del uso** es al mismo tiempo la capacidad de operar el aparato más los usos sociales.

Flichy (2003) establece una interesante distinción entre usuarios estratégicos y usuarios tácticos. Los **usuarios estratégicos** generalmente negocian con los diseñadores el marco de referencia e intervienen en su creación, pues pueden especificar las funcionalidades y utilidades de las herramientas tecnológicas que quieren utilizar. En contraposición, los **usuarios tácticos** no negocian el marco de la función, pero pueden influir en la transformación del marco del uso. Según Flichy, en una actividad tecnológica llega un momento cuando ocurre una estabilización del marco de referencia; en ese momento, todos los actores son tácticos.

La mayoría de los usuarios de las computadoras son usuarios tácticos, poseen un marco de uso; es decir, saben lo necesario para operar la computadora, más una serie de usos sociales, como ver el correo, buscar información, grabar música, etc. Sin embargo, no saben, en realidad, cómo funciona la computadora, ni como cambiar o mejorar sustancialmente su funcionamiento. El marco de la función son los conocimientos técnicos compartidos por los diseñadores, técnicos y usuarios estratégicos. Por tanto, el marco de la

función es necesario para «abrir la caja negra» y modificar el aparato tecnológico.

En las comunidades de desarrollo de GNU/Linux de todo el mundo participan muchos usuarios estratégicos, programadores en su mayoría, quienes conocen el marco de la función de los sistemas GNU/Linux, incluso algunos intervienen en su creación y colaboran activamente en proyectos de desarrollo concretos. Hay muchas maneras de colaborar, desde probar las últimas versiones y reportar los posibles errores, hasta aportar nuevo código e incluso convertirse en desarrollador oficial. Al igual que una comunidad de científicos, las comunidades de GNU/Linux tienen su propio marco de referencia socio-tecnológico que es necesario conocer para poder formar parte del grupo social relevante.

De acuerdo con esta perspectiva, GNU/Linux es el resultado de un proceso multidireccional donde participan diversos actores e intereses, y, en este sentido, la construcción social de la tecnología constituye una aproximación útil para el análisis de su proceso de innovación tecnológica.

Conclusiones

1. GNU/Linux ha sido y continúa siendo un artefacto tecnológico desarrollado por diversos grupos sociales relevantes, tales como comunidades, esta última integrada por desarrolladores y usuarios (Kernel.org, la comunidad Debian), empresas (Red Hat, Canonical Ltd) u organizaciones sin fines de lucro *Free Software Foundation*. Cada grupo social relevante se organiza en torno a este producto tecnológico y tiene su propia visión acerca de su significado, uso y desarrollo.
2. La flexibilidad interpretativa que presenta GNU/Linux como *software* libre es mayor que la del *software* privativo, por el hecho de entregar a cualquier usuario o desarrollador el código fuente. Cualquier persona con conocimientos en programación puede libremente reutilizar total o parcialmente el código fuente y brindar soluciones para sí mismo o para terceros.
3. Los usuarios que participan en el desarrollo de GNU/Linux son usuarios estratégicos, programadores en su mayoría, quienes conocen el marco de la función, incluso algunos intervienen en su creación y colaboran activamente en el desarrollo de *software*: prueban las llamadas versiones inestables, informan a los desarrolladores oficiales sobre los errores en el código, en ocasiones aportan nuevo código, e incluso algunos pueden llegar a ser desarrolladores oficiales de importantes proyectos.
4. GNU/Linux es un ejemplo claro de un producto tecnológico construido socialmente (COST). Nadie puede negar el papel crucial que ha jugado la comunidad de usuarios y desarrolladores de GNU/Linux en todo el mundo. Sin el trabajo colectivo de miles de programadores que han colaborado en el proyecto GNU/Linux a través de la red, no sería lo que es ahora, un sistema operativo robusto y estable que ha penetrado con fuerza no solo en las universidades, sino también en el mundo empresarial, desbancando a sistemas operativos sustentados por grandes corporaciones y grandes presupuestos.

Referencias bibliográficas

- Bijker, W., Hughes, T., y Pinch, T. 1987. *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*. MIT Press, Cambridge.
- Bijker, W. E. 1987. The Social Construction of Bakelite: To-ward a Theory of Invention. *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*.
- Callon, M. 1987. Society in the making: The study of technology as a tool for sociological analysis. W. Bijker, T. Hughes, y T. Pinch, eds., *The social construction of technological systems*. MIT Press.

- Cutcliffe, S. 2003. Emergencia histórica del cts como campo académico. *Ideas Máquinas y valores, los estudios de ciencia tecnología y sociedad*, 7-55. Antrhopos.
- Dosi, G. 1982. Technological Paradigms and Technological Trajectories: A Suggested Interpretation of the Determinants and Directions of Technical Change. *Research Policy*, 11(3): 147-162.
- Díaz, R. 2003. Itinerarios de conocimiento: formas dinámicas y contenidos. Matilde Luna, ed., *Contra el exilio de los objetos un acercamiento a la teoría de la red de actores*, 79-104. Antrhopos ISS-UNAM, México.
- Díaz, R. y Lee, M. 1992. La innovación tecnológica: dos aproximaciones teóricas en competencia. Campos y Varela, ed., *Prospectiva Social y Revolución Científico-Tecnológica*, 55-71. UNAM-UAM.
- Flichy, P. 2003. *L' innovation technique*. La Découverte, París.
- Flores, S. 2009a. *Las comunidades de software libre de Costa Rica*. Tesis de maestría, Universidad de Costa Rica, San José.
- Flores, S. 2009b. Las comunidades de software libre de costa rica. *Revista de Ciencias Sociales*, En prensa.
- Himanen, P. 2001. *The Hacker Ethic, and the Spirit of the Information Age*. Random House, New York.
- Hughes, T. 1983. *Networks of Power: Electrification in Western Society 1880-1930*. Johns Hopkins University Press, Baltimore.
- Hughes, T. 1987. The evolution of large technological systems. W. Bijker, T. Hughes, y T. Pinch, eds., *The social construction of technological systems*. MIT Press, Cambridge.
- Kuhn, T. S. 1971. *La estructura de las revoluciones científicas*. Fondo de Cultura Económica, México.
- Lancashire, D. 2001. The fading altruism of open source development. *First Monday*, 6(12). URL [http://firstmonday.org/issues/issue6/s\do4\(1\)2/lancashire/index.html](http://firstmonday.org/issues/issue6/s\do4(1)2/lancashire/index.html).
- Latour, B. 1991. *Nunca hemos sido modernos*. Debate, Madrid.
- Latour, B. 1992. *Ciencia en acción: cómo seguir a los científicos e ingenieros a través de la sociedad*. Labor, Barcelona.
- Levy, S. 2001. *Hackers: Heroes of the Computer Revolution*. Penguin Books, New York.
- Lizama Mendoza, J. A. 2005. *Hackers en el contexto de la sociedad de la información*. Tesis doctoral, Facultad de Ciencias Políticas y Sociales, México, UNAM.
- Pinch, T. 1997. La construcción social de la tecnología: una revisión. *Innovación tecnológica y procesos culturales: Nuevas perspectivas teóricas*, 20-38. Universidad Autónoma de México - Fondo de Cultura Económica, México.
- Raymond, E. S. 1998. The cathedral and the bazaar. *First Monday*, 3(3). URL [http://www.firstmonday.org/issues/issue3/s\do4\(3\)/raymond/](http://www.firstmonday.org/issues/issue3/s\do4(3)/raymond/).
- Stallman, R. 2004. *Software libre para una sociedad libre*. Traficantes de sueños, Madrid.