

DESARROLLO FLOSS Y GOBERNANZA DE PARES: EL CASO DEL ENTORNO DE ESCRITORIO GNOME

DESENVOLVIMENTO FLOSS E GOVERNANÇA DE PARES: O CASO DO AMBIENTE DE DESKTOP GNOME

FLOSS DEVELOPMENT AND PEER GOVERNANCE: THE CASE OF THE GNOME DESKTOP ENVIRONMENT

 Agustín Zanotti¹

 Juan Gabriel Vélez²

Cite as – American Psychological Association (APA)

Zanotti, A., & Vélez, J. G. (2020, Sept./Dec.). Desarrollo Floss y gobernanza de pares: el caso del entorno de escritorio Gnome. *International Journal of Innovation - IJI*, São Paulo, 8(3), 438-465.
<https://doi.org/10.5585/iji.v8i3.17114>.

Resumen

Objetivo del estudio: El artículo analiza el caso de GNOME, uno de los proyectos Free/Libre Open Source Software (FLOSS) más populares, iniciado en 1997. El concepto de gobernanza de pares describe la interacción y convergencia de empresas, fundaciones, usuarios voluntarios y profesionales en su desarrollo.

Metodología/ enfoque: La investigación se basa en el método documental y un abordaje de historia organizacional. Para ello se recopilaron fuentes primarias y secundarias digitales: sitios institucionales y FLOSS, blogs, listas de comunidad, documentos y plataformas.

Originalidad/ Relevancia: El trabajo permite avanzar, desde una perspectiva sociotécnica, en la comprensión de los desarrollos FLOSS y la coevolución de sus componentes técnicos y dinámicas comunitarias.

Resultados principales: Se identifican tres niveles de gobernanza: 1. el software propiamente dicho; 2. la comunidad; 3. el ecosistema. Se analizan ciclos de desarrollo; participación y organización comunitaria; actores y modelos de negocio, definiciones y controversias. La trayectoria de GNOME está marcada por la confluencia de intereses y cooperación entre actores corporativos y el ecosistema FLOSS. El ciclo de vida muestra una actividad constante, que implica redefinición de sus componentes, infraestructura tecnológica y liderazgos. Ella no exenta de controversias y bifurcaciones, similares a las evidenciadas en proyectos de este tipo.

Contribuciones teóricas/ metodológicas: Se aporta al concepto de gobernanza de pares y su utilidad para el análisis de casos. Se promueve la reflexión sobre fuentes de datos y recursos de la investigación en Internet.

Contribuciones sociales/ gerenciales: Se contribuye a la comprensión de nuevas formas de gestión y toma de decisión en proyectos de desarrollo tecnológico.

Palabras clave: Floss. Desarrollo de software. Comunidad. Ecosistema. Governance.

¹ Doctor en Estudios Sociales de América Latina, Universidad Nacional de Córdoba, Argentina. azanotti@unvm.edu.ar

² Licenciado en Economía, Universidad Nacional de Córdoba, Argentina. juangabrielvelez@unraf.edu.ar

Resumo

Objetivo do estudo: O artigo analisa o caso do GNOME, um dos mais populares projetos de Software Livre/ Open Source (FLOSS), iniciado em 1997. O conceito de governança de pares descreve a interação e convergência de empresas, fundações, usuários voluntários e profissionais em seu desenvolvimento.

Metodologia/ abordagem: A pesquisa baseia-se no método documental e na abordagem da história organizacional. Para isso, foram coletadas fontes digitais primárias e secundárias: sites institucionais e FLOSS, blogs, listas de comunidades, documentos e plataformas.

Originalidade/ Relevância: O trabalho permite avançar, de uma perspectiva sociotécnica, na compreensão dos desenvolvimentos do software livre e na coevolução de seus componentes técnicos e dinâmica da comunidade.

Principais resultados: Três níveis de governança são identificados: 1. o próprio software; 2. a comunidade; 3. o ecossistema. São analisados os ciclos de desenvolvimento, participação e organização da comunidade, atores e modelos de negócios, definições e controvérsias. A trajetória do GNOME é marcada pela confluência de interesses e cooperação entre atores corporativos e o ecossistema FLOSS. O ciclo de vida mostra uma atividade constante, o que implica redefinição de seus componentes, infraestrutura tecnológica e liderança. Não está isento de controvérsias e bifurcações, semelhantes às evidenciadas em projetos desse tipo.

Contribuições teóricas/ metodológicas: Contribui para o conceito de governança de pares e sua utilidade para a análise de casos. É promovida a reflexão sobre fontes de dados e recursos de pesquisa na Internet.

Contribuições sociais/ gerenciais: Contribui para o entendimento de novas formas de gestão e tomada de decisão em projetos de desenvolvimento tecnológico.

Palabras-chave: Floss. Desenvolvimento de software. Comunidade. Ecossistema. Governança.

Abstract

Objective of the study: The article analyzes the case of GNOME, one of the most popular Free/ Libre Open Source Software (FLOSS) projects, started in 1997. The concept of peer governance describes the interaction and convergence of companies, foundations, voluntary users and professionals in its development.

Methodology/ approach: The research is based on the documentary method and an organizational history approach. For this, primary and secondary digital sources were collected: institutional and FLOSS sites, blogs, community lists, documents and platforms.

Originality/ Relevance: The work allows progress, from a sociotechnical perspective, in understanding FLOSS developments and the coevolution of their technical components and community dynamics.

Main results: Three levels of governance are identified: 1. the software itself; 2. the community; 3. the ecosystem. Development cycles; community participation and organization; actors and business models, definitions and controversies, are analyzed. GNOME's trajectory is marked by the confluence of interests and cooperation between corporate actors and the FLOSS ecosystem. The life cycle shows a constant activity, which implies redefinition of its components, technological infrastructure and leadership. It is not exempt from controversies and bifurcations, similar to those evidenced in projects of this type.

Theoretical/ methodological contributions: It contributes to the concept of peer governance and its usefulness for case analysis. Reflection on data sources and research resources on the Internet is promoted.

Social/ managerial contributions: It contributes to the understanding of new forms of management and decision-making in technological development projects.

Keywords: Floss. Software development. Community. Ecosystem. Governance.

Introducción

El movimiento de software libre surge a principios de los 1980s desde los centros de desarrollo académicos como el MIT, como una reacción a la clausura y privatización de la incipiente industria SSI (Software y Servicios Informáticos). Fue iniciado por Richard Stallman, creador de la Licencia Pública General (GPL), la *Free Software Foundation* y el Proyecto GNU (Sistema operativo GNU, s.f.). Las licencias tipo GPL obligan a las creaciones derivadas a mantener las condiciones de uso originales, garantizando que el código permanezca abierto y no pueda ser apropiado con exclusividad. A finales de los 1990, se produjo una escisión entre un sector del movimiento basado en los principios filosóficos y políticos de la libertad del software, y otro basado en las ventajas técnicas y económicas del código abierto (Raymond, 1999), bajo la órbita de la Open Source Initiative. El término FLOSS reúne los conceptos de *Free/Libre Open Source Software*¹.

Los proyectos FLOSS son instancias de producción entre pares, definidas como una forma de creación abierta y compartida realizada por grupos online que: establecen y ejecutan objetivos de forma descentralizada; manejan un rango diverso de motivaciones, en particular motivaciones no-monetarias; y se separan de formas de propiedad exclusivas y relaciones contractuales (Benkler et al., 2015, p. 2).

Forman parte de un grupo de bienes comunes informacionales, basados en derechos de uso simétricos, por oposición a derechos exclusivos asimétricos (Hess y Ostrom, 2007). Estos recursos ocupan un rol central en las economías actuales (Benkler, 2017).

Buena parte de la literatura sobre innovación la define como algo estrictamente ligado a la empresa. La historia de las innovaciones relacionadas con las Tecnologías de la Información y Comunicación (TIC) y los proyectos FLOSS han mostrado, sin embargo, que esta definición es insuficiente en casos prácticos relevantes (Tuomi, 2002). Ello permite abonar cuestiones emergentes en la investigación sobre innovación abierta, redes y ecosistemas de innovación, en particular: modos de gobernanza en comunidades, formas de apropiación no basados en propiedad intelectual y articulación de actores diversos a escala global (empresas, voluntarios, usuarios y organizaciones sin fines de lucro).

A lo largo del artículo analizamos el caso de GNOME (*GNU Network Object Model Environment*). Iniciado en 1997, es uno de los entornos de escritorio más populares para sistemas GNU/Linux, disponible en las principales distribuciones. Desde un abordaje documental e histórico, se prioriza un enfoque narrativo con orientaciones interpretativas y

¹ En el artículo utilizamos de manera unificada ambas expresiones, dado que refieren al mismo modelo de producción de software en términos productivos.

discursivas (da Costa et al., 2010). Con base en el constructivismo social de la tecnología, utilizamos la definición de trayectoria sociotécnica, como un proceso de “co-construcción de productos, procesos productivos y organizacionales” (Thomas, 2008, p. 249). Esta se inspira en la idea de ensamble sociotécnico de Bijker (1993), la cual busca superar determinismos tanto tecnológicos como sociales en el análisis de procesos de innovación.

El trabajo permite de este modo comprender la coevolución de los componentes y dinámicas comunitarias de GNOME. El abordaje se basa en el concepto de gobernanza de pares e identifica tres niveles de gobernanza: 1. el software propiamente dicho; 2. la comunidad; 3. el ecosistema. Para ello, identificamos momentos críticos que fueron direccionando la evolución del desarrollo. Analizamos sus orígenes, organización e infraestructura; participación y actividad comunitaria; relación entre actores y modelos de negocio, bifurcaciones, definiciones y controversias; así como su inserción en el ecosistema FLOSS.

La investigación se basa en la recopilación de fuentes secundarias: sitios institucionales y temáticos, blogs, listas de comunidad, fuentes de datos oficiales y de terceras partes, extraídos de plataformas digitales.

Marco de referencia teórica

Innovación abierta y ecosistema de innovación

El concepto de innovación abierta fue propuesto por Chesbrough (2003). Su definición más actualizada lo entiende como “un proceso de innovación distribuida que implica flujos de conocimiento gestionados intencionadamente a través de los límites de la organización” (Chesbrough y Bogers, 2014, p. 4). La innovación abierta puede estudiarse en diferentes niveles de análisis: unidades de negocio, ecosistemas/comunidades, empresas, regiones o incluso sistemas nacionales de innovación. Sin embargo, hasta la fecha ha tendido a centrarse en el nivel de empresa, con una atención más reciente dirigida a proyectos y equipos (Chesbrough y Bogers, 2014).

En cierta proximidad, Von Hippel ha desarrollado los términos innovación de usuario, innovación distribuida (2005) e innovación colaborativa (Baldwin y von Hippel, 2011). Tales definiciones tienen en común haber estudiado casos de FLOSS (Piller y West, 2014). Sin embargo, una línea divisoria entre ellas es la conveniencia de que los creadores tengan derechos fuertes para apropiarse de los beneficios de sus invenciones.

El concepto de innovación abierta se ha enriquecido desde su creación a través de otros marcos en innovación, gestión y economía. Algunos desafíos señalados recientemente son:

nuevos enfoques para medir el proceso de innovación; interacción entre estrategias de innovación abierta y las opciones de apropiabilidad fuerte (o débil); formas de colaboración y en red, aplicación de la innovación abierta a nuevos contextos (como organizaciones sin fines de lucro o empresas más pequeñas) y nuevos niveles de análisis (West et al., 2014).

Vallat (2009) propone que los procesos de innovación abierta se basan en el establecimiento de redes entre actores (incluidas industrias, universidades y organizaciones de investigación, entidades públicas, usuarios finales y comunidades de usuarios) y el uso de plataformas funcionales para facilitar la colaboración y participación. Esta se asocia al concepto de ecosistema de innovación, construido a partir del concepto de *business ecosystem* originalmente propuesto por Moore. Mientras el primero busca enfatizar la creación de valor total, la noción del segundo se preocupa más por la captura de valor por parte de las empresas (de Vasconcelos Gomes et al., 2018).

El constructo de ecosistema de innovación enfatiza la co-evolución de una constelación heterogénea de organizaciones que crean valor de forma conjunta (Dedehayir, Mäkinen y Ort, 2018). Este concepto mantiene un sentido biológico de interacción y simbiosis entre diferentes actores, que incluye a usuarios finales, y donde no hay una geografía específica sino que se utilizan plataformas y tecnologías en común que permiten la interacción (Dedehayir et al, 2018).

En la literatura es usual encontrar estudios de la cooperación entre firmas como un fenómeno habitual en diversas actividades productivas (industria automotriz, farmacéutica, aeronáutica, entre otras). Teixeira et al. (2016) han empleado el neologismo *open-coopetition* para designar esta situación de cooperación entre competidores cuando se da en el marco de modelos open-source.

Desarrollo Floss

El modo tradicional de producción de software ha sido transformado por el auge del modelo de producción abierta y colaborativa. Vinculado a los desarrollos FLOSS se encuentra la noción de ecosistema software. Manikas y Hansen (2013) realizaron una revisión sistemática de la literatura en torno al concepto de ecosistema software donde proponen una definición como síntesis de las más utilizadas. Siguiendo a los autores, entendemos como ecosistema software a un conjunto de actores que interactúan sobre la base de una plataforma o tecnología en común, que da como resultado un conjunto de productos y servicios. En esta definición están

presentes tanto los actores que desarrollan en conjunto como el artefacto en sí que constituye la tecnología común sobre la que interactúan.

Dentro de este ecosistema software los actores desarrollan relaciones simbióticas entre sí, y se admite una amplia gama de intereses y motivaciones para participar que incluyen beneficios no monetarios. En el ecosistema existen diferentes roles. Generalmente es una firma la que orquesta o coordina la plataforma o tecnología en común, mientras que otras empresas pueden desarrollar extensiones o funciones en base a ella. Se genera valor y se distribuyen los costos asociados a las innovaciones (Jansen, 2020).

Esta transformación en el modo de producir software tradicional ha generado un desafío en cuanto a la posibilidad de que las firmas logren desarrollar plataformas que permitan el surgimiento de comunidades software a la vez que sean capaces de controlar este proceso. A raíz de ello, la literatura del management ha propuesto modelos de gobernanza que, desde la perspectiva de la firma detrás de la plataforma, logren mantener la salud del ecosistema software. Las prácticas de gobernanza se ubican específicamente en torno a los problemas de creación de valor, coordinación de actores y la apertura y control organizacional (Jansen, 2020).

Más específicamente en los desarrollos FLOSS, nos encontramos que éstos están conformados por numerosos proyectos, donde cada uno cuenta con decenas o cientos de programadores a nivel global, conectados en redes y plataformas de producción colaborativa, que han evolucionado en las últimas décadas generando formas eficientes de administrar el flujo de trabajo. En los grandes proyectos existe una ecología de comunidades y subcomunidades que comparten una gobernanza y producen artefactos que usualmente son compartidos.

Los proyectos liberan versiones estables luego de cada ciclo de desarrollo, y en general mantienen accesible el proceso mediante plataformas abiertas. El código utiliza licencias libres/abiertas existentes, pudiendo distinguirse entre licencias recíprocas (o *virales*), que obligan a mantener las condiciones de código abierto, y licencias permisivas, que habilitan desarrollos cerrados derivados².

Es común encontrar una estructura organizativa de los proyectos dividida en dos grupos principales: *core* y periferia. En el *core*, se encuentran los líderes del proyecto, mantenedores y otros desarrolladores principales. La periferia está compuesta por un gran número (en los proyectos más exitosos) de desarrolladores que prueban el software, informan errores y fallas, proponen parches o mejoras al código (Motta et al., 2019). Es necesario considerar también las

² Un repaso de los tipos de licencias y modificaciones recientes se encuentra en Wheeler (2017).

contribuciones que no son de programación. Estas giran en torno a documentación, traducción, diseño y desarrollo de interfaces de usuario, arte, trabajo comunitario y de divulgación (Alleyne, 2011; Vasilescu et al., 2014).

Los proyectos funcionan sin un proceso formal de recruiting y la relación entre los desarrolladores es voluntaria y no mediada por contratos formales. Más allá de eso, los colaboradores pueden poner en valor indirectamente sus contribuciones, ya sea como una forma de capital social, tecnológico o simbólico (Zanotti, 2014).

Gobernanza de proyectos comunitarios

Encontramos en el FLOSS una forma de gobierno que involucra una red de actores (Vélez y Zanotti, 2020). La misma ha sido comprendida como gobernanza de pares, un modo de gobernanza y un modelo ascendente de toma de decisiones participativas (Bauwens, 2005). En esta no operan jerarquías estrictas de comando y control, sino un entorno mucho más flexible que permite la existencia de múltiples equipos de participantes que trabajan simultáneamente en una variedad de direcciones, posiblemente opuestas (Bruns, 2008, p. 26).

Nyman y Lindman (2013) identifican tres niveles de gobernanza de pares, referidos a:

1. *el software propiamente dicho*, que debe evolucionar permanentemente para satisfacer a sus usuarios y no quedar obsoleto. El principio del FLOSS es "lanzar temprano y lanzar a menudo". Esto significa que los prototipos de software se desarrollan y ponen en circulación rápidamente para su revisión por pares (Alleyne, 2011). Esta práctica, difundida luego al resto de la industria, se materializa en ciclos incrementales e iterativos, en los cuales se realizan mejoras de diseño y se agregan nuevas funcionalidades (Larman y Basili, 2003).
2. *gestión de la comunidad*, en la cual la propiedad colectiva del código abierto permite crear bifurcaciones del desarrollo (*forks*). Se trata de escisiones respecto del desarrollo original, tomando el código e iniciando una nueva comunidad. En la medida en que implican oportunidades y dificultades, son un principio regulador de los conflictos y la gobernanza (Nyman y Lindman, 2013).
3. *el ecosistema* en que se inserta, la ecología de comunidades FLOSS que incluye otros grandes proyectos, actores corporativos y fundaciones, en el marco de relaciones de cooperación y competencia. Utilizamos el término *coopetición*, neologismo propuesto por Teixeira y Lin (2014), para describir la cooperación entre competidores

en el ámbito del código abierto. Las empresas vinculadas al FLOSS contribuyen de diferentes modos al desarrollo, al tiempo que generan modelos de negocio basados en soporte y servicios asociados (Vélez y Zanotti, 2020).

Métodos

La investigación se basa en un abordaje de historia organizacional, centrado en los sentidos y procesos organizacionales (da Costa et al., 2010). Para ello, utiliza el método documental, definido como un enfoque analítico en donde las declaraciones y representaciones de las personas objeto de investigación se sitúan en contexto (Bohnsack, 2014). Esto puede incluirse como documento en la investigación social cubre un espectro potencialmente amplio. En la actualidad, los materiales abarcan una amplia gama de medios tecnológicos, digitales y sociales (Coffey, 2014). Estos últimos fueron seleccionados tomando en cuenta la particularidad del caso de estudio, haciendo uso de las siguientes fuentes primarias y secundarias:

- Fuentes del proyecto: 1. GNOME Gitlab, la plataforma de desarrollo actualmente en uso por la comunidad permite obtener un listado de proyectos activos. 2. GNOME Bugzilla, se utilizaron reportes disponibles en la plataforma para determinar la actividad de usuarios y desarrolladores.
- Fuentes de terceras partes: 1. Open Hub, permite obtener tendencias generales sobre la evolución de numerosos proyectos FLOSS. En particular se utilizaron estimaciones sobre el volumen del proyecto en líneas de código generadas y su valoración comercial. 2. Netmarketshare y Statcounter, se utilizaron aproximaciones sobre cuota de mercado de las distribuciones GNU/Linux en entornos de escritorio.
- Sitios institucionales y listas públicas de comunidad: GNOME, News GNOME, Wiki GNOME, Planet GNOME. Fueron utilizados para la contextualización del proyecto y el rastreo de temas y discusiones. Para ello fueron analizados en orden cronológico y seleccionados intencionalmente en función de su relevancia para la indagación.
- Sitios sobre FLOSS: Muiylinux, Phoronix. Se trata de sitios externos al proyecto que incluyen información periódica sobre lanzamientos y nuevas funcionalidades.

Se realizaron búsquedas por palabra clave o tag para compilar los artículos referidos a GNOME.

- **Documentos:** 1. Documentos de la GNOME Foundation. Fueron utilizados para extraer definiciones y tendencias oficiales del proyecto. 2. Notas de edición, fueron utilizadas para determinar la cantidad de *commits*³ y contribuidores de cada lanzamiento estable. También permitieron reconstruir el historial de versiones del proyecto.
- **Blogs:** blogs.gnome.org, Liberal Murmurs. Fueron utilizados para el rastreo de temas y discusiones. Fueron analizadas sólo aquellas entradas referidas en listas públicas de comunidad.

Las recomendaciones metodológicas seguidas corresponden al uso de fuentes secundarias, la verificación cruzada de datos y la descripción de fuentes de datos. Se cumplen recomendaciones éticas, de privacidad y resguardo en el caso de los métodos digitales para ciencias sociales, así como las prescripciones sobre uso de datos generados por plataformas digitales (Roberts et al., 2016). Siempre que fuera posible, se incluyeron los enlaces correspondientes en la sección de referencias.

Resultados

El abordaje se basa en el concepto de gobernanza de pares e identifica tres niveles de gobernanza: 1. el software propiamente dicho; 2. la comunidad; 3. el ecosistema. Cada nivel, contempla dos dimensiones de análisis: 1.1 Evolución del desarrollo; 1.2 Organización e infraestructura; 2.1 Fundación; 2.2 Actividades comunitarias; 3.1 Relación entre actores; 3.2 Inserción en el FLOSS.

Gobernanza del desarrollo

Evolución del desarrollo

El proyecto GNOME surge para desarrollar un entorno de escritorio y aplicaciones fáciles de usar, basadas completamente en software libre⁴ (GNOME, 2017). A diferencia de iniciativas anteriores, fue íntegramente desarrollado en código abierto -con un decidido apoyo

³ Un *commit*, en el contexto de la informática, es un conjunto de cambios en el código que ha sido revisado e incorporado al desarrollo.

⁴ En 1996 se iniciaron los desarrollos de KDE y de Xfce, pero ambos utilizaban librerías con licencia privativa, Qt y Xforms respectivamente. Con el paso del tiempo, KDE se adecuó a las licencias GPL y Xfce abandonó sus componentes privativos.

del Proyecto GNU y la *Free Software Foundation* (FSF)-, lo que evitaba potenciales conflictos por incompatibilidad de licencias privativas.

El GIMP ToolKit (GTK+) fue desarrollado como base de GNOME, una plataforma que soporta diferentes lenguajes de programación. Desde sus inicios, el kit de herramientas y sus bibliotecas utilizan una Licencia Pública General Reducida (LGPL) de GNU, que permite que el software vinculado con ella utilice un conjunto más amplio de licencias, incluidas las de software propietario. En tanto sus aplicaciones son licenciadas con GPL (GNOME, 2020c).

El nacimiento de GNOME estuvo marcado por la presencia de la comunidad FLOSS mexicana. Sus fundadores, Miguel de Icaza y Federico Mena Quintero, se desempeñaban en esos momentos en la Universidad Nacional Autónoma de México (UNAM). Según relatan:

En 1996 recluté a dos estudiantes de la recientemente creada Licenciatura en Informática en la Facultad de Ciencias de la UNAM. Esos dos estudiantes fueron Arturo Espinosa y Federico Mena. Estaban trabajando en una variedad de proyectos de software libre y relacionados con mi lenguaje favorito en ese momento: Scheme. Trabajaron conmigo en software libre y en proyectos divertidos en el Instituto de Ciencias Nucleares de la Universidad Nacional Autónoma de México. Allí también trabajamos en el Grupo Mexicano de Usuarios de Linux. (de Icaza, 2002).

La primera versión (0.0) se publicó en agosto de 1997 e incluía sólo una aplicación simple y un conjunto de bibliotecas. Desde el inicio contó con fuerte patrocinio de Red Hat, firma pionera en negocios basados en FLOSS, quien contrató a los desarrolladores para continuar el proyecto a tiempo completo:

Para noviembre de ese año [1997], GNOME había alcanzado el impulso suficiente para que tanto Federico como yo abandonáramos otros proyectos y nos dedicamos por completo a trabajar en GNOME. ... Tuvimos una reunión en los primeros días de diciembre con Marc Ewing y DrMike en las oficinas de Red Hat. ... Marc estaba creando *Red Hat Advanced Development Labs*, que sería el primer equipo de desarrolladores financiado que trabajaría para mejorar GNOME. RHAD Labs se creó en enero de 1998. (de Icaza, 2002).

La versión 1.0 se lanzó en marzo de 1999 y se integró en Red Hat Linux como escritorio predeterminado. La serie 1.x tuvo tres lanzamientos durante los siguientes dos años. De Icaza desarrolló el administrador de archivos Midnight Commander, aplicaciones como Gnumeric, el modelo de componentes Bonobo y la plataformas Mono. Esta última generaría cierta controversia por tratarse de una implementación de .Net, bajo licencias de Microsoft (de Icaza, 2002). Mena Quintero comenzó coordinando Gimp, perfeccionó Midnight Commander y desarrolló el calendario Evolution, entre otros aportes.

La firma Helix Code (más tarde Ximian), fundada en 1999 en Massachusetts, creó infraestructura y aplicaciones de GNOME. En 2003 fue comprada por Novell. La startup Eazel de California desarrolló el administrador de archivos Nautilus, que permanece hasta la actualidad como componente principal.

El proyecto se organizó en cuatro grupos principales de módulos: (a) bibliotecas requeridas (como las de interfaz gráfica, impresión, procesamiento XML, CORBA); (b) aplicaciones principales (como sus applets, un editor, un administrador de ventanas y configuración); (c) aplicaciones secundarias y (d) otros (varias docenas de módulos y aplicaciones no pertenecientes al núcleo de GNOME) (German, 2003).

En junio de 2002 se lanzó GNOME 2.0, con mejoras en la experiencia de usuario y desarrollo. Hacia 2003, el escritorio sumaba casi dos millones de líneas de código y más de 500 colaboradores distribuidos en el mundo (German, 2003). La introducción de *Directrices de Interfaz Humana* (HIG) permitieron un entorno más predecible y coherente para los usuarios (GNOME, 2017). La serie 2.x, vio 16 lanzamientos en los siguientes 9 años. Con ella se concretaron partes pendientes del escritorio como GNOME Office -un conjunto de aplicaciones de oficina- y características de accesibilidad avanzadas, entre otras.

En abril de 2011 se presentó GNOME 3.0, un renovado escritorio pensado para múltiples dispositivos⁵. Los cambios incluían la eliminación de la barra de tareas inferior, los botones de minimizar y maximizar, y múltiples escritorios. Estos marcaron un peso creciente del diseño y la experiencia de usuario en el desarrollo del proyecto. Se trabajó en aplicaciones dedicadas a tareas y la integración de cuentas en línea (GNOME, 2017).

GNOME alcanzó en 2019 unas 16,3 millones de líneas de código. Según estimaciones, ello implicaría un trabajo equivalente a 5143 personas durante un año, lo cual genera un valor total de U\$S 282 millones (Open Hub, 2019). La llegada de una nueva serie 4.x se espera para 2021, con el lanzamiento del nuevo *toolkit* (GTK Development Blog, 2020).

Organización e infraestructura

GNOME ha destacado por la regularidad en su desarrollo, en un ecosistema donde pocos logran mantener un ciclo de vida extenso bajo el mismo esquema de lanzamientos. El proyecto contempla dos ediciones estables anuales desde 2002, publicadas en marzo y septiembre (GNOME, 2020g). Las versiones estables se identifican por su número par (por ejemplo 3.34),

⁵ Más allá de algunas adecuaciones, la llegada de GNOME a otros dispositivos como smartphones no sucedería hasta la llegada de Purism en 2017, una iniciativa basada en crowdsourcing que lanzó sus primeros equipos con este entorno.

mientras las en desarrollo por su número impar (por ejemplo 3.33). El escritorio alcanzó en marzo de 2020 su versión 3.36, la decimonovena edición de la serie y 41 desde su lanzamiento.

Para ello, GNOME saca provecho de una estructura modular. Si bien las partes del proyecto están interrelacionadas entre sí, se busca reducir al mínimo su interdependencia para que cada una pueda evolucionar en modo separado. Cada módulo tiene uno o más mantenedores que supervisan el desarrollo, coordinan e integran los *commits* (German, 2004). Algunos pueden contar con una firma patrocinadora no oficial, que emplea desarrolladores pagos para realizar el trabajo (German, 2003).

Los mantenedores coordinan esfuerzos para alcanzar una versión completa del escritorio y sus bibliotecas principales cada seis meses. Ello incluye unas 30 aplicaciones principales (*core*). Otro conjunto más amplio de *extras* están excluidos de estas versiones. El repositorio Gitlab cuenta al momento con 418 subproyectos vinculados, que forman parte de sus componentes y aplicaciones (GNOME Gitlab, 2020).

Cada iteración incluye asimismo un encuentro de desarrollo intensivo (*sprint*) en una diferente locación, la cual usualmente da nombre a la versión final. En los encuentros se planifican acciones, ofrecen charlas y se discute el horizonte y visión del proyecto. El desarrollo también se nutre de becas de programación como *Google Summer of Code*, programa global enfocado en introducir a los estudiantes al software de código abierto.

En cuanto a la infraestructura, GNOME ha utilizado sistemas de versiones⁶ y herramientas de comunicación para permitir el trabajo colaborativo y reporte de errores. Más tarde varias funciones fueron integradas en plataformas de desarrollo de software. Las universidades fueron una fuente importante de recursos en los inicios del proyecto. La UNAM, donde trabajaba de Icaza, proporcionó soporte a través de un servidor FTP, correo y un sistema de control de versiones. Posteriormente diferentes organizaciones sumaron repositorios al proyecto, en ocasiones sin el conocimiento de sus propias autoridades (German, 2003).

Desde su versión 1.0.55 GNOME utilizó Bugzilla, la herramienta de seguimiento de errores desarrollada por Mozilla. La misma facilita a los usuarios reportar los fallos y se mantiene un registro de su estado actual, en el cual son confirmados, clasificados, asignados, resueltos y verificados. Al momento de la indagación, la plataforma mantenía más de 227 000 *bugs* resueltos y corregidos (GNOME Bugzilla, 2020).

También se han desarrollado herramientas de internacionalización. *Malditas mentiras* es la plataforma web que gestiona el flujo de las traducciones en 191 idiomas (GNOME, 2020e).

⁶ Los sistemas de versiones mantiene registro del trabajo y los cambios en los archivos, permitiendo que distintos desarrolladores colaboren simultáneamente. Es una de las bases que hacen posible la producción en masa entre pares.

Cada edición pone en disponibilidad la totalidad de su código fuente en el Servidor FTP del proyecto, que a su vez contiene todo el historial de versiones desde su inicio.

En 2015 se lanzó la versión inicial de Builder, un entorno de desarrollo integrado (IDE) diseñado para ayudar a escribir aplicaciones basadas en GNOME. Builder se mantiene activo hasta la actualidad y sus lanzamientos acompañan las ediciones estables generales.

En 2018 se realizó una transición a la plataforma GitLab, la cual ofrece una solución completa integrada para el alojamiento de código y seguimiento de problemas, con una interfaz simplificada para los menos experimentados (GNOME Wiki, 2020a).

Tabla 1 – Gobernanza del desarrollo

Evolución del desarrollo	Organización e infraestructura
GNOME 1.x:	Ciclos de desarrollo incremental
<ul style="list-style-type: none"> • difusión del proyecto y reclutamiento inicial • creación de aplicaciones y bibliotecas core 	<ul style="list-style-type: none"> • 2 ediciones estables anuales • modularidad, mantenedores por módulo
GNOME 2.x:	<ul style="list-style-type: none"> • sprints • becas
<ul style="list-style-type: none"> • aplicaciones secundarias • accesibilidad • internacionalización 	Plataformas de desarrollo
GNOME 3.x:	<ul style="list-style-type: none"> • control de versiones • repositorios • seguimiento de errores • internalización • IDE propio • Git Lab
<ul style="list-style-type: none"> • mejoras visuales, usabilidad • integración cuentas en líneas • multidispositivo 	

Fuente: Elaboración propia.

Gobernanza de la comunidad

Fundación

La gobernanza de GNOME se organiza desde el año 2000 a través de la *GNOME Foundation* (de Icaza, 2002). De acuerdo a su Carta de fundación, entre sus objetivos encontramos:

- La Fundación coordinará las versiones de GNOME y determinará qué proyectos forman parte.

- Actuará como una voz oficial para el proyecto, proporcionando un medio de comunicación con la prensa y con organizaciones comerciales y no comerciales interesadas en el software.
- Puede producir materiales educativos y documentación para ayudar al público a aprender sobre GNOME.
- Puede patrocinar conferencias técnicas relacionadas y representar a GNOME en conferencias relevantes patrocinadas por otros.
- Puede ayudar a crear estándares técnicos para el proyecto, promover el uso y desarrollo del software (GNOME, 2000).

La Junta Directiva es el organismo que gobierna la fundación. Los colaboradores miembros tienen derecho a postularse en estos cargos ejecutivos, así como participar en su elección anual. La membresía es personal. Participa de la Junta Directiva uno de sus fundadores, Federico Mena Quintero.

La Fundación cuenta además con una Junta Asesora formada por empresas y organizaciones patrocinantes. Las pequeñas firmas hacen un pago anual de U\$S 11 500, las medianas y grandes de U\$S 23 000. Actualmente integran la Junta Asesora: Debian, Endless, FSF, Google, Private Internet Access, Red Hat, Sugar Labs, SUSE, The Document Foundation, System76 y Canonical, que se sumó en 2017 (GNOME, 2020b). Entre los miembros aparecen tecnológicas líderes, empresas especializadas y otras fundaciones FLOSS.

La Junta Asesora se encarga de representar los intereses de organizaciones, reuniéndose regularmente con la Junta Directiva para definir la dirección del proyecto y prestar apoyo. Se interpreta la participación de las empresas como un mecanismo que canaliza la demanda de los usuarios, los requisitos específicos de los usuarios finales y que tienen mayor contacto por intermedio del mercado (Berdou, 2010).

La Fundación actuó en defensa de la propiedad de la marca GNOME, la cual le fue disputada en una oportunidad. También debió afrontar demandas por violaciones de patentes, siendo el último caso en 2019 (GNOME News, 2019). Desde 2006 lleva un programa de pasantías con rondas organizadas dos veces al año. Patrocina asimismo eventos como GUADEC (GNOME User and Developer European Conference).

En lo que respecta al financiamiento, los pagos provenientes de la Junta Asesora representan una porción menor de sus ingresos. La mayor parte proviene de donaciones. A lo largo de su historia el proyecto atravesó algunas crisis de financiamiento, como en 2009, que

pusieron en jaque la continuidad de la fundación, Llegó por otra parte a recibir donaciones anónimas de gran volumen, siendo la mayor de 1 millón de dólares en 2018 (GNOME News, 2018).

En su operatoria, la Fundación cuenta con un Director ejecutivo; Director de operaciones; Coordinador de programas, Gerente de iniciativas estratégicas; Líder de equipo de desarrollo GTK; Ingeniero jefe de sistemas e Ingeniero DevOps (GNOME, 2019c).

Actividades Comunitarias

La comunidad GNOME se define a partir de cuatro pilares: independencia, libertad, conexión y orientación a las personas (GNOME, 2020a). El proyecto ha desarrollado un código de conducta que define una serie de comportamientos aceptados y rechazados:

Valoramos su participación y queremos que todos tengan una experiencia agradable y satisfactoria. En consecuencia, se espera que todos los participantes sigan este Código de Conducta y muestren respeto, comprensión y consideración entre ellos. Gracias por ayudar a que esta sea una comunidad acogedora y amigable para todos. (GNOME Wiki, 2020b).

El proyecto cuenta con varios canales de comunicación, incluyendo listas de distribución e IRC. Entre los espacios comunitarios destaca *planet.gnome.org*, un feed de novedades para compartir experiencias e ideas respecto al mundo FLOSS.

Un evento de comunidad destacado es la conferencia anual GUADEC, la conferencia principal de los usuarios y desarrolladores. El primer encuentro se celebró en 2000, en París, y desde entonces se celebra anualmente. En 2020 se organizó por primera vez fuera del continente europeo, en México (GNOME, 2020d). A esto se suman encuentros propios (como GNOME Summit, GNOME Asia Summit), talleres y hackatones, así como fiestas de lanzamiento de comunidades locales. GNOME también participa en eventos del mundo FLOSS (como Linux Expo, Linux App Summit y Libre Application Summit, entre otros) para compartir avances sobre el proyecto. Los encuentros sirven para el fortalecimiento y la celebración de los vínculos y la identidad común (Coleman, 2010).

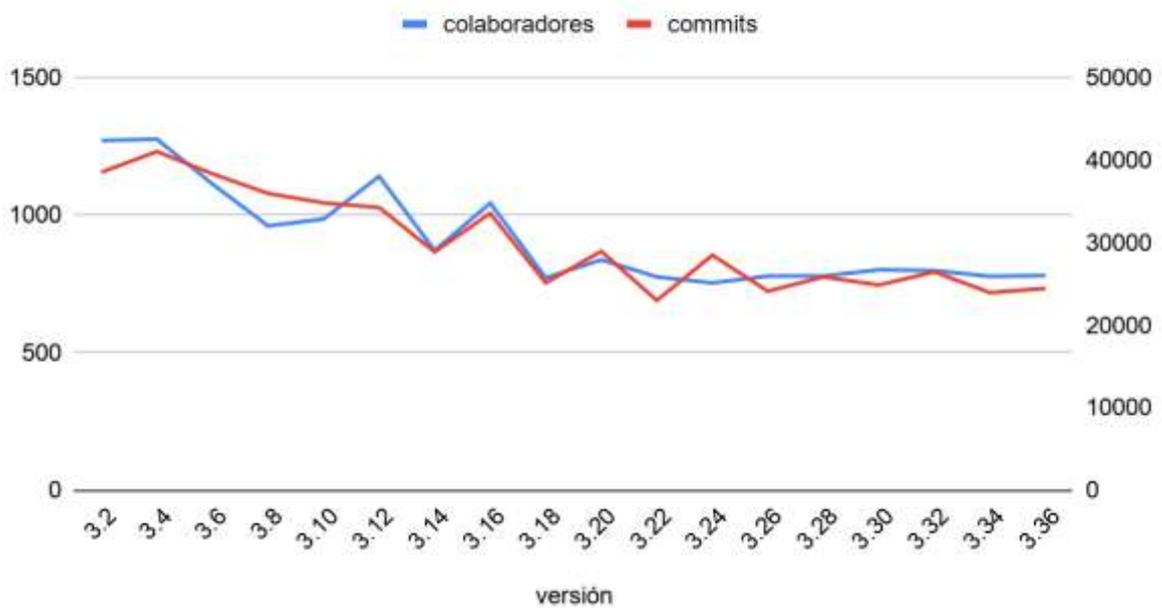
La GNOME Foundation cuenta con un programa de pasantías orientado a ampliar la diversidad de la comunidad. Su objetivo es involucrar a personas y grupos subrepresentados por su lugar de pertenencia, etnicidad o género. En 2019 se conformó además el Equipo de diversidad e inclusión en GNOME:

Nuestro equipo da la bienvenida y alienta la participación de todos. Para nosotros no importa cómo te identifiques o cómo te perciban los demás: te damos la bienvenida. Somos una organización sin fines de lucro y estar abiertos y accesibles para todos está en nuestros fundamentos. Creemos firmemente que la diversidad y la inclusión son puntos clave para una comunidad saludable y exitosa. (GNOME, 2020f)

En cuanto a la participación, el Gráfico 1 permite analizar su ciclo de vida en el periodo 2011 a 2020. Encontramos un paulatino declive tanto en la cantidad de colaboradores activos (de 1270 a 780) como en sus *commits* (de 38500 a 24323) (GNOME, 2020g).

Figura 1 – Evolución de Gnome 3.2 (2011) a 3.36 (2020)

Gráfico 1. Evolución de Gnome 3.2 (2011) a 3.36 (2020)



Nota: figura realizada en base a datos de gnome.org.

La trayectoria de crecimiento, cima y declive gradual es una constante en el modelo de producción entre pares en diferentes dominios. Hill y Shaw (2020), proponen una explicación a partir de la paulatina “clausura” de los proyectos abiertos, en función de proteger el acervo acumulado a lo largo del tiempo. También se ha constatado la impronta de los fundadores y la necesidad de un reclutamiento permanente de participantes para la supervivencia del proyecto (Zanotti, 2014; Cheng et al., 2017).

Con el objetivo de renovar la comunidad y sumar principiantes, GNOME ha desarrollado diferentes iniciativas como la *Newcomers Guide* y el *Community Engagement Challenge*, con incentivos económicos para proyectos en forma de premios (GNOME, 2020b).

Tabla 2 – Gobernanza de la comunidad

Fundación	Actividades comunitarias
• Junta directiva (individuos)	• Valores y pilares
• Junta asesora (organizaciones)	• Herramientas de comunicación, sociales
• Administración de recursos	• Conferencia anual GUADEC
• Membresías	• Reuniones locales
• Donaciones	• Proyectos de diversidad e inclusión
• Representación de marca	• Reclutamiento de nuevos participantes
• Patrocinio de eventos y proyectos	

Fuente: Elaboración propia.

Gobernanza del Ecosistema

Relación entre Actores

Como vimos, la historia de GNOME está estrechamente asociada a voluntarios, empresas, proyectos y fundaciones FLOSS, siendo los más protagónicos integrantes de la Junta Asesora de la GNOME Foundation.

Berdou (2007) observó que un número cada vez mayor de desarrolladores trabajan en forma remunerada para GNOME. Los últimos datos disponibles son el *Censo GNOME* realizado en 2010. De este se desprende que más del 70% de los participantes se identifican como voluntarios, mientras más del 30% de los *commits* son realizados por desarrolladores pagos. Esto significa que los aportes remunerados mantienen un rol activo en la comunidad, aunque existe una amplia periferia. Con todo, los aportes voluntarios se ubican en el primer lugar en cuanto a cantidad (23,5%), seguidos por aquellos sin determinar (17%) (Neary y David, 2010). También se observa que los módulos de infraestructura y aplicaciones *core* han sido mantenidos de forma mayoritaria por desarrolladores pagos de empresas como RedHat, Novell, Intel y Canonical; mientras que los voluntarios tienden a mantener aplicaciones secundarias y librerías (Neary y David, 2010).

Red Hat era en ese momento el mayor contribuyente: 11 de los 20 principales desarrolladores eran empleados actuales o pasados de esta firma. Novell y Nokia también estaban en el podio. También encontramos en los primeros lugares a empresas de consultoría/servicios asociados: Collabora, CodeThink, Openismus, Lanedo y Fluendo, entre otras (Neary y David, 2010).

Otro socio de GNOME son las distribuciones de software. Estas proveen infraestructura y mantienen colecciones de paquetes compilados para los usuarios finales. Debian, una de las

principales, mantuvo desde su inicio soporte para GNOME, convirtiéndose en su escritorio por defecto. En 2003 surgen Fedora (sucesora de Red Hat Linux) y en 2004 Ubuntu, una distribución basada en Debian por la recientemente creada Canonical. Con la ayuda de instaladores, gestión de controladores, soporte extendido y otras herramientas, ambas distribuciones fueron ampliando la cuota de usuarios finales.

El caso de Ubuntu resulta particular⁷. Aunque siempre mantuvo el escritorio GNOME por defecto, una vez alcanzada una amplia base de usuarios buscó diferenciarse con un gestor de ventanas propio (Unity) y otros desarrollos. Al cabo de unos años, volvió a adherirse a GNOME, incorporándose a su Fundación en 2017 (Muylinux, 2017). Este comportamiento es habitual en empresas FLOSS, que realizan aportes mientras ensayan modelos de negocio asociados.

A lo largo de su historia GNOME disputó el entorno de escritorio con KDE, iniciado en 1996. RedHat y Novell han sido colaboradores principales en ambos desarrollos, aunque compitan en el segmento. La rivalidad entre GNOME y KDE no impide que realicen eventos conjuntos como la *Free Desktop Summit* (Alleyne, 2011). Las sinergias entre actores y proyectos se potencian en la medida en que utilizan recursos comunes. Tal como propone uno de sus referentes, el desarrollo del escritorio implica una interacción intensiva con varios niveles de software, por lo que GNOME termina involucrándose en el sistema entero:

El proyecto GNOME descubrió que no era solo una capa y que necesitaba intervenir en componentes y subsistemas de nivel inferior para lograr sus objetivos. Nos convertimos en diseñadores e ingenieros OS. El resultado: a lo largo de los años, los miembros del proyecto GNOME han generado importantes piezas de infraestructura que se han convertido en partes vitales de los sistemas operativos basados en Linux. (The GNOME Way, 2017)

Así es que componentes centrales como GStreamer, D-Bus, NetworkManager, Polkit, systemd, OSTree, entre otros, han sido creados por contribuyentes de GNOME.

Inserción en el FLOSS

GNU/Linux es un caso de éxito en el mundo tecnológico, academia e industria⁸. Sin embargo, no ha sido así en el escritorio. En un posteo de 2001, uno de los fundadores de

⁷ Bajo el lema "Linux para seres humanos" inició una fuerte campaña para conquistar el escritorio. Incluía el envío de CDs de instalación a pedido a todo el mundo en forma gratuita.

⁸ Linux es el sistema operativo en más del 95% del millón de dominios principales. Más del 80% de los nuevos teléfonos inteligentes vendidos funcionan con Android, que se basa en el kernel de Linux. Las 500 mejores supercomputadoras del mundo funcionan con Linux. La mayoría de los mercados mundiales se ejecutan en Linux.... La mayoría de los dispositivos electrónicos de consumo usan Linux por su pequeño tamaño. Más del 75% de las empresas habilitadas para la nube informan que utilizan Linux como su plataforma principal en la nube. Linux es la infraestructura de acceso a los líderes mundiales de comercio electrónico, incluidos Amazon, eBay, PayPal, Walmart y otros (The Linux Foundation, 2020).

GNOME anticipa lo que sería una de sus maldiciones: la incapacidad de alcanzar una adopción masiva de usuarios finales: “¡Estamos condenados! ¡Estamos condenados! Otro reportero con la visión de un profeta ha predicho que Linux en el escritorio no tiene ninguna posibilidad” (de Icaza, 2001).

Parte de esta visión se asocia a su utilización por un nicho de programadores o usuarios avanzados, que prefiere la eficiencia por sobre la experiencia de usuario. Varias campañas se realizaron en años posteriores para ampliar la participación en el escritorio, como 10×10 (que pretendía llegar al 10% de mercado para el 2010) (GNOME Wiki, 2005). Si bien su uso se ha ampliado de la mano de distribuciones más amigables, el objetivo está aún lejos de alcanzarse⁹. El mismo de Icaza, comentó años más tarde que “Linux nunca se las arregló para cruzar el abismo de escritorio”, criticando su fragmentación, múltiples distribuciones e incompatibilidades entre versiones (de Icaza, 2012).

Otro eje de crítica hacia GNOME se relaciona con la falta de objetivos y las decisiones que orientan su desarrollo. Se destaca, en este sentido, una dinámica propia de la producción entre pares, donde los proyectos se van nutriendo de programadores voluntarios o empresas interesadas en desarrollar partes específicas. Eso ocasiona que ciertos componentes queden abandonados o con escaso mantenimiento.

En números anecdóticos: GTK tiene una sola persona trabajando a tiempo completo... Glib ni siquiera tiene eso. Creo que Evolution tiene una situación similar (un cliente completo de email)... Verán, como mucho, 20 desarrolladores full-time. Si... ignoran a las 3 personas que trabajan casi exclusivamente en GStreamer y las 2 que trabajan con traducciones, tendrán 10 empleados de Red Hat y 5 del resto. (Muylinux, 2012)

A raíz de ello se reportan problemas de ingeniería, mal rendimiento e inestabilidad, o falta de madurez de los lanzamientos, lo cual hace que ciertas distribuciones retrasen su incorporación.

GNOME ha enfrentado críticas al diseño y funcionamiento de sus interfaces, sobre todo a partir de la serie 3.x, la cual innovó considerablemente su aspecto visual. Referentes como el propio Linus Torvalds han expresado disconformidad en algunas oportunidades (Muylinux, 2011).

Otras reacciones notables fueron dos *forks* llevados a cabo por miembros de la comunidad. El primero es el proyecto Cinnamon, que deriva de GNOME 3, iniciado en el año

⁹ Fuentes de la industria ubican a las distribuciones GNU/Linux en torno al 1,7%-1,8% de la cuota de mercado de sistemas operativos de escritorio (Netmarketshare, 2020; Statcounter, 2020).

2011 con el respaldo de la comunidad Linux Mint; el segundo es MATE, un proyecto desarrollado en base a GNOME 2.

Más allá de las recepciones iniciales, GNOME 3 logró aceptación y se mantiene como escritorio por defecto en las principales distribuciones. El proyecto ha profundizado el análisis y diseño de experiencia de usuario, tal como se expone en uno de los blogs de desarrollo:

Al llamar la atención sobre la experiencia del usuario, este principio amplía la definición de lo que crea el proyecto GNOME: no solo una pieza funcional de software, sino algo que tiene cualidades estéticas y emocionales distintas. Esto requiere que GNOME 3 sea tratado como una entidad única y no como una colección de partes. (The GNOME Way, 2017)

GNOME estableció desde el inicio en relación con el Proyecto GNU y la FSF, dos instituciones fundantes del FLOSS. Este vínculo se ha visto sin embargo tensionado en algunas oportunidades, en función de las diferencias ideológicas entre software libre y open source, donde algunos desarrolladores se manifestaron en favor del último término (Muylinux, 2009). Un reciente cuestionamiento sucedió en 2019, cuando Richard Stallman -al frente de la FSF desde su inicio en 1985-, se vio involucrado en un escándalo que terminó motorizando su renuncia (McGovern, 2019).

La salida del histórico dirigente sirvió para traer a primer plano algunas problematizaciones que se venían anunciando en sectores del FLOSS. El posteo replicado en planet.gnome.org, *Do we need to rethink what free software is?*, generó recientemente amplia discusión. El tema es la vigencia de las licencias libres en relación a ciertos usos desleales de parte de corporaciones y gobiernos (Garrett, 2019):

¿Cómo nos aseguramos de que la producción de software libre no sea solo un mecanismo para la transformación de la mano de obra no remunerada en ganancias corporativas? Soy lo suficientemente afortunado como para que me paguen por escribir software libre, pero muchos proyectos de inmensa importancia en infraestructura son a la vez fundamentales para múltiples modelos de negocio y con financiación insuficiente. (Garrett, 2019)

Ciertas nuevas licencias intentan, en este sentido, restringir modelos de negocio basados en el suministro de software como servicio. Tal es el caso de la *Commons Clause*, cláusula que impide a terceros la venta del software, obligando a remitirse a los desarrolladores originales (The Commons Clause, 2020).

Al mismo tiempo reaparecen consideraciones sobre las consecuencias éticas del FLOSS. La crítica apunta a la supuesta indiferencia del software respecto de sus usos probables, sobre todo en el caso de gobiernos que utilizan tecnología para acciones de vigilancia masiva,

persecución y violaciones a la privacidad (Garrett, 2019). Mientras licencias como *JSON* incluyen una cláusula ética: “el software se utilizará para el bien, no para el mal” (JSON.org, 2002), la falta de una definición estricta sobre estos términos hace que sea difícil su implementación.

Se concluye así que la ausencia de discusión de nuevos temas dentro de la comunidad, asociada a una crisis de dirección del movimiento, genera el riesgo de que el FLOSS se debilite y fragmente.

Tabla 3 – Gobernanza del ecosistema

Relación entre actores	Inserción en el FLOSS
<ul style="list-style-type: none"> • grandes firmas • startups • distribuciones FLOSS • proyectos y organizaciones FLOSS • desarrollos e iniciativas cruzadas 	<ul style="list-style-type: none"> • controversias sobre adopción, uso y desarrollo • bifurcaciones • definiciones éticas-políticas • adopción de licencias

Fuente: Elaboración propia.

Discusión y consideraciones finales

En el trabajo realizamos un recorrido por la trayectoria de GNOME, siguiendo una indagación histórica documental. Desde un abordaje sociotécnico, analizamos la coevolución de sus componentes tecnológicos y sociales en los tres niveles de gobernanza de pares identificados. La inexistencia de jerarquías rígidas en el gobierno del proyecto es posible por el despliegue de mecanismos que permiten coordinar y conciliar intereses de los actores participantes (programadores, empresas, usuarios, fundaciones), así como alentar la adhesión de voluntarios.

Los hallazgos pretenden contribuir al desarrollo de procesos de innovación abierta centrados en el nivel de un proyecto comunitario. Incorporamos el concepto de sistema de innovación con la finalidad de observar dinámicas de creación de valor y su apropiación. Asimismo, la definición de ecosistema software contempla un conjunto de actores que interactúan sobre la base de una plataforma o tecnología en común, generando un conjunto de productos y servicios. Las características del FLOSS permiten a las empresas establecer relaciones simultáneas de cooperación y competencia, dando lugar al término *coopetición*. Uno de sus principios es evitar el trabajo redundante y liberar el acceso al código fuente.

La gobernanza de los procesos y la asignación de recursos pueden, sin embargo, dar lugar a inconvenientes, como lo demuestra el caso de GNOME. Algunos proyectos pueden sufrir una falta de orientación o estar insuficientemente dotados de personal para su desarrollo y mantenimiento.

El surgimiento de GNOME estuvo marcado por la participación de empresas líderes, emprendedores y voluntarios, con apoyo de universidades. El proyecto demuestra una regularidad en los lanzamientos y ciclos de desarrollo, al tiempo que una innovación constante en sus componentes, infraestructura y plataformas que habilitan el trabajo colaborativo y la productividad.

La transición de comunidades basadas en voluntariado a una creciente participación de empresas, es una constante en proyectos FLOSS exitosos y ha sido abordado por la literatura (Daniel et al., 2011; González-Barahona y Robles, 2013). Peters (2009) se ha referido a efectos potencialmente perjudiciales de un gran número de desarrolladores pagos en un proyecto: las motivaciones intrínsecas que atrajeron a un desarrollador pueden erosionarse y reemplazarse por una puramente económica. Del lado de las firmas observamos estrategias de apropiabilidad débil, basadas en acuerdos de trabajo y el uso de marcas registradas, lo que permite ofrecer una variedad de servicios asociados (Lund y Zukerfeld, 2020).

En conjunto, GNOME muestra una configuración híbrida de voluntarios y desarrolladores pagos en la producción de FLOSS. Encontramos en la investigación una limitación en la falta de datos censales actualizados, los cuales permitirían ponderar la incidencia de cada actor a lo largo de su ciclo de vida. En este sentido, futuras indagaciones podrían profundizar sobre los datos disponibles en las plataformas de infraestructura del proyecto, así como recabados por terceras partes.

En cuanto a su comunidad, GNOME llegó a erigirse como un proyecto global con desarrolladores de diferentes latitudes. Su fundación mantuvo un rol activo en la dirección de la comunidad, con la realización de acciones para fortalecerla, generar identidad e incorporar nuevos miembros. A ello se suman acciones de defensa legal y búsqueda de financiamiento, a partir de donaciones y patrocinio. Los aspectos sociales también influyen significativamente en la forma en que el proyecto evoluciona a lo largo del tiempo, tal como han mostrado investigaciones precedentes (Mens y Goeminne, 2011).

En el nivel del ecosistema, la gobernanza de proyectos FLOSS tiene como desafío mantener unidos a diferentes actores. Cuando esto no sucede, es habitual que se produzcan bifurcaciones o *forks*, tomando el código del proyecto para darle una nueva dirección.

Encontramos controversias en torno a la adopción de GNOME, su usabilidad y desarrollo. También definiciones ético-políticas relacionadas con el mundo FLOSS, la adopción de licencias y su visión de futuro. Más allá de esto, el entorno sigue logrando converger intereses entre actores corporativos y comunitarios para mantener un liderazgo como escritorio GNU/Linux.

Con el paso de los años, GNOME se constituyó en uno de los proyectos de mayor duración dentro del ecosistema FLOSS. Su ciclo de vida muestra una actividad constante, que implica redefinición de componentes, infraestructura tecnológica y orientación. Ella no está exenta de conflictos y derivas, similares a las evidenciadas en otros proyectos de su tipo.

Referencias

- Alleyne, B. (2011). Challenging Code: A sociological reading of the KDE Free Software project. *Sociology*, 45(3), 496-511. <https://doi.org/10.1177/0038038511399620>.
- Baldwin, C. y von Hippel, E. (2011). Modeling a Paradigm Shift: From Producer Innovation to User and Open Collaborative Innovation. *Organization Science*, 22(6), 1399-1417. <https://doi.org/10.1287/orsc.1100.0618>.
- Bauwens, M. (2005). The political economy of peer production. *CTheory*, 12-1.
- Benkler, Y. (2017). Open access and information commons en F. Parisi (Ed.), *The Oxford Handbook of Law and Economics: Volume 2: Private and Commercial Law*. Oxford University Press.
- Benkler, Y., Shaw, A. y Hill, B. M. (2015). Peer production: A form of collective intelligence en T. W. Malone y M. S. Bernstein (Eds.), *Handbook of collective intelligence* (pp. 175-204). The MIT Press.
- Berdou, E. (2007) *Managing the bazaar: commercialization and peripheral participation in mature, community-led free/open source software projects*. (Tesis doctoral no publicada). The London School of Economics and Political Science.
- Berdou, E. (2010). *Organization in open source communities: At the crossroads of the gift and market economies*. Routledge.
- Bijker, W. E. (1993). Do not despair: there is life after constructivism. *Science, Technology, y Human Values*, 18(1), 113-138. <https://doi.org/10.1177/016224399301800107>.
- Bohnsack, R. (2014). Documentary method en U. Flick (Ed.), *The SAGE handbook of qualitative data analysis* (pp. 217-233). SAGE.
- Bruns, A. (2008). *Blogs, Wikipedia, Second Life, and beyond: From production to produsage*. Peter Lang.

- Cheng, C., Li, B., Li, Z. Y., Zhao, Y. Q., y Liao, F. L. (2017). Developer role evolution in open source software ecosystem: An explanatory study on GNOME. *Journal of computer science and technology*, 32(2), 396-414. <https://doi.org/10.1007/s11390-017-1728-9>.
- Chesbrough, H. W. (2003). *Open Innovation: The new imperative for creating and profiting from technology*. Harvard Business School Press.
- Chesbrough, H. y Bogers, M. (2014). Explicating Open Innovation: Clarifying an Emerging Paradigm for Understanding Innovation en H. Chesbrough, W. Vanhaverbeke y J. West, (Eds.), *New Frontiers in Open Innovation*. Oxford University Press.
- Coffey, A. (2014). Analysing documents en U. Flick, (Ed.), *The SAGE handbook of qualitative data analysis* (pp. 367-380). SAGE.
- Coleman, G. (2010). The Hacker Conference: A Ritual Condensation and Celebration of a Lifeworld. *Anthropological Quarterly*, 83(1), 47-72.
- Daniel, S., Maruping, L., Cataldo, M., y Herbsleb, J. (2011). When cultures clash: Participation in open source communities and its implications for organizational commitment en *ICIS 2011 Proceedings* 7.
- da Costa, A. D. S. M., Barros, D. F. y Martins, P. E. M. (2010). Perspectiva histórica em administração: novos objetos, novos problemas, novas abordagens. *RAE-Revista de Administração de Empresas*, 50(3), 288-299.
- de Icaza, M. (mayo de 2001). linux-on-the-desktop. *Miguel de Icaza's blog*. <https://tirania.org/blog//archive/2001/May.html>.
- de Icaza, M. (2002). The Story of the GNOME project. <https://web.archive.org/web/20070818123652/http://primates.ximian.com/~miguel/gnome-history.html>.
- de Icaza, M. (29 de agosto de 2012). What Killed the Linux Desktop. *Miguel de Icaza's blog*. <https://tirania.org/blog/archive/2012/Aug-29.html>.
- de Vasconcelos Gomes, L. A., Facin, A. L. F., Salerno, M. S., y Ikenami, R. K. (2018). Unpacking the innovation ecosystem construct: Evolution, gaps and trends. *Technological Forecasting and Social Change*, 136, 30-48. <https://doi.org/10.1016/j.techfore.2016.11.009>.
- Dedehayir, O., Mäkinen, S. J., y Ortt, J. R. (2018). Roles during innovation ecosystem genesis: A literature review. *Technological Forecasting and Social Change*, 136, 18-29. <https://doi.org/10.1016/j.techfore.2016.11.028>.
- The GNOME Way. (8 de agosto de 2017). *Form and Function*. *Allan Day's GNOME Blog*. <https://blogs.gnome.org/aday/2017/08/08/the-gnome-way/>.
- Garrett, M. (26 de septiembre de 2019). *Do we need to rethink what free software is?* mjpg59's journal. <https://mjpg59.dreamwidth.org/52907.html>.

- German, D. M. (2003). The GNOME project: a case study of open source, global software development. *Software Process: Improvement and Practice*, 8(4), 201-215.
<https://doi.org/10.1002/spip.189>.
- German, D. M. (2004). Using software trails to reconstruct the evolution of software. *Journal of Software Maintenance and Evolution: Research and Practice*, 16(6), 367-384.
<https://doi.org/10.1002/smr.301>.
- GNOME. (2000). *GNOME Foundation Charter Draft 0.61*. GNOME Wiki.
<https://wiki.gnome.org/FoundationBoard/Resources/Charter>.
- GNOME. (2017). *Happy birthday GNOME!* <http://happybirthdaygnome.org/>.
- GNOME. (2020a). *About us*. <https://www.gnome.org/about/>.
- GNOME. (2020b). *GNOME Foundation and Endless Launch Inaugural Community Engagement Challenge*. <https://www.gnome.org/news/2020/04/gnome-foundation-and-endless-launch-inaugural-community-engagement-challenge/>.
- GNOME. (2020c). *GNOME Foundation Guidelines on Copyright Assignment*. GNOME Wiki.
<https://wiki.gnome.org/FoundationBoard/Resources/CopyrightAssignment/Guidelines>.
- GNOME. (2020d). *Events*. <https://www.gnome.org/category/events/>.
- GNOME. (2020e). *Idiomas de GNOME*. <https://l10n.gnome.org/languages/>.
- GNOME. (2020f). *Outreachy/History*. GNOME Wiki.
<https://wiki.gnome.org/Outreachy/History>.
- GNOME. (2020g). *Release Notes*. <https://help.gnome.org/misc/release-notes/>.
- GNOME. (2020h). *The GNOME Foundation*. <https://www.gnome.org/foundation/>.
- GNOME Bugzilla. (s.f.) *Report: Component/ Status*.
<https://bugzilla.gnome.org/page.cgi?id=reports.html>.
- GNOME GitLab. (2020). *Details*. <https://gitlab.gnome.org/GNOME>.
- GNOME News (25 de mayo de 2018). *Anonymous Donor Pledges \$1M Donation Over Two Years*. <https://www.gnome.org/news/2018/05/anonymous-donor-pledges-1m-donation-over-two-years/>.
- GNOME News (25 de septiembre de 2019). *GNOME Foundation facing lawsuit from Rothschild Patent Imaging*. <https://www.gnome.org/news/2019/09/gnome-foundation-facing-lawsuit-from-rothschild-patent-imaging/>.
- GNOME Wiki (2005). *10x10*. Recuperado el 4 de marzo de 2020 de
<https://wiki.gnome.org/10x10>.

GNOME Wiki (2020a). *Development Infrastructure Evaluation*.
<https://wiki.gnome.org/Initiatives/DevelopmentInfrastructure>.

GNOME Wiki (2020b). *GNOME Code of Conduct*.
<https://wiki.gnome.org/Foundation/CodeOfConduct>.

González-Barahona, J. M., y Robles, G. (2013). Trends in free, libre, open source software communities: from volunteers to companies. *it-Information Technology*, 55(5), 173-180.
<https://doi.org/10.1515/itit.2013.1012>.

GTK Development Blog. (13 de febrero de 2020). *GTK 3.98*.
<https://blog.gtk.org/2020/02/13/gtk-3-98/>.

Hess, C. y Ostrom, E. (Eds.) (2007). *Understanding Knowledge as a Commons: From Theory to Practice*. MIT Press.

Hill, B. M. y Shaw, A. (2020). Wikipedia and the End of Open Collaboration? en J. R. Reagle y J. L. Koerner (Eds.), *Wikipedia@20. Stories of an Incomplete Revolution*. The MIT Press.

Jansen, S. (2020). A focus area maturity model for software ecosystem governance. *Information and Software Technology*, 118, 106219.
<https://doi.org/10.1016/j.infsof.2019.106219>.

JSON.org (2002). *The JSON License*. <https://www.json.org/license.html>.

Koch, S., y Schneider, G. (2002). Effort, co-operation and co-ordination in an open source software project: GNOME. *Information Systems Journal*, 12(1), 27-42.
<https://doi.org/10.1046/j.1365-2575.2002.00110.x>.

Larman, C., y Basili, V. R. (2003). Iterative and incremental developments. a brief history. *Computer*, 36(6), 47-56.

Lund, A. y Zukerfeld, M. (2020). *Corporate Capitalism's Use of Openness: Profit for Free?* Palgrave Macmillan.

Manikas, K. y Hansen, K. M. (2013). Software ecosystems – A systematic literature review. *Journal of Systems and Software*, 86(5), 1294-1306.
<https://doi.org/10.1016/j.jss.2012.12.026>.

McGovern, N. (18 de septiembre de 2019). GNOME relationship with GNU and the FSF. *Liberal Murmurs*. <https://blog.halon.org.uk/2019/09/gnome-foundation-relationship-gnu-fsf/>.

Mens, T. y Goeminne, M. (2011). Analysing the evolution of social aspects of open source software ecosystems en Jansen, Bosch, Ahmed, y Campell (Eds.), *Proceedings of the Workshop on Software Ecosystems* (pp. 1-14).

Motta, J., Morero, H. A. y Borrastero, C. (2019). Measuring Non Monetary Innovation In Software: a case study in floss firms from Argentina. *International Journal of Innovation*, 7(1), 135-154.

- Muylinux. (14 de diciembre de 2009). *¿GNOME vs GNU?*.
<https://www.muylinux.com/2009/12/14/gnome-vs-gnu/>.
- Muylinux. (5 de diciembre de 2011). *Linus: «GNOME 3.2 es casi usable»*.
<https://www.muylinux.com/2011/12/05/linus-gnome-3-2-es-casi-usable/>.
- Muylinux. (4 de agosto de 2012). *GNOME, a la deriva*.
<https://www.muylinux.com/2012/08/04/gnome-a-la-deriva/>.
- Muylinux. (4 de abril de 2017). *Ubuntu abandona movilidad, convergencia y a Unity: vuelve GNOME*. <https://www.muylinux.com/2017/04/05/ubuntu-convergencia-unity-gnome/>.
- Neary, D., y David, V. (2010). *The GNOME census: Who writes GNOME?*. <https://tech-insider.org/linux/research/acrobat/100720.pdf>.
- NetMarketShare (2020). *Operating System Market Share*.
<https://www.netmarketshare.com/operating-system-market-share.aspx>.
- Nyman, L. y Lindman, J. (2013). Code Forking, Governance, and Sustainability in Open Source Software. *Technology Innovation Management Review*, 3(1), 7-12. 10.22215/TIMREVIEW/644.
- Open Hub. (2019). *The GNOME Open Source Project on Open Hub: Estimated Cost Page*.
https://www.openhub.net/p/gnome/estimated_cost.
- Peters S. (2009). Open Source Is Changing the Way Work Gets Done en C. Boldyreff, K. Crowston, B. Lundell y A. I. Wasserman (Eds.), *Open Source Ecosystems: Diverse Communities Interacting*. Springer. https://doi.org/10.1007/978-3-642-02032-2_1.
- Piller, F. y West, J. (2014). Firms, Users, and Innovation: An Interactive Model of Coupled Open Innovation en H. Chesbrough, W. Vanhaverbeke y J. West (Eds.), *New Frontiers in Open Innovation*. Oxford University Press. 10.1093/acprof:oso/9780199682461.003.0002.
- Raymond, E. (1999). The Cathedral and the Bazaar. *Knowledge, Technology y Policy*, 12(3): 23-49.
- Roberts, S., Snee, H., Hine, C., Morey, Y., y Watson, H. (Eds.). (2016). *Digital methods for social science: An interdisciplinary guide to research innovation*. Palgrave Macmillan.
- Schaarschmidt, M. (2012). *Firms in open source software development: managing innovation beyond firm boundaries*. Springer Science y Business Media.
- Sistema operativo GNU. (s.f.). *El manifiesto de GNU*.
<https://www.gnu.org/gnu/manifiesto.es.html>.
- Statcounter Globalstats (2020). *Desktop Operating System Market Share Worldwide*.
<https://gs.statcounter.com/os-market-share/desktop/worldwide/>.

- Teixeira, J. y Lin, T. (2014). Collaboration in the open-source arena: the webkit case. *Proceedings of the 52nd ACM conference on Computers and people research*, 121-129. <https://doi.org/10.1145/2599990.2600009>.
- Teixeira, J. A., Mian, S. Q. y Hytti, U. (2016). Cooperation among competitors in the open-source arena: The case of OpenStack. [Documento de trabajo]. International Conference on Information Systems, Dublin, Ireland.
- The Commons Clause (2020). “Commons Clause” License Condition v1.0. <https://commonsclause.com/>.
- The Linux Foundation (2020). *About the Linux Foundation*. <https://www.linuxfoundation.org/about/>.
- Thomas, H. (2008). Estructuras cerradas vs. Procesos dinámicos: trayectorias y estilos de innovación y cambio tecnológico en H. Thomas y A. Buch, *Actos, actores y artefactos. Sociología de la tecnología*. Universidad Nacional de Quilmes.
- Tuomi, I. (2002). *Networks of Innovation: Change and Meaning in the Age of the Internet*. Oxford University Press.
- Vallat, J. (2009) Intellectual Property and Legal Issues in Open Innovation in Services. OISPG, European Commission.
- Vasilescu, B., Serebrenik, A., Goeminne, M. y Mens, T. (2014). On the variation and specialisation of workload—a case study of the Gnome ecosystem community. *Empirical Software Engineering*, 19(4), 955-1008. 10.1007/s10664-013-9244-1.
- Vélez, J. G. y Zanotti, A. (2020). La producción en comunidades FLOSS: empresas, fundaciones y governance en H. Morero y J. Motta (Eds.), *La economía del software libre y open source: Multinacionales, Pymes y Comunidades* (pp. 55-75). Estudios Sociológicos.
- von Hippel, E. (2005). *Democratizing Innovation*. The MIT Press.
- West, J., Salter, A., Vanhaverbeke, W. y Chesbrough, H. (2014). Open innovation: The next decade. *Research Policy*, 43(5), 805-811. <https://doi.org/10.1016/j.respol.2014.03.001>.
- Wheeler, D. A. (2017). *The free-libre/open source software (FLOSS) license slide*. <https://dwheeler.com/essays/floss-license-slide.html>.
- Zanotti, A. (2014). Comunidades de software libre en Argentina: motivaciones, participación, militancia. *Perspectivas de la Comunicación*, 7(2), 55-74.