

En *LA ECONOMÍA DEL SOFTWARE LIBRE Y OPEN SOURCE: MULTINACIONALES, PYMES Y COMUNIDADES*. Ciudad Autónoma de Buenos Aires (Argentina): Estudios Sociológicos Editora.

# La producción en comunidades FLOSS: empresas, fundaciones y governance.

Vélez, Juan Gabriel y Zanotti, Agustín.

Cita:

Vélez, Juan Gabriel y Zanotti, Agustín (2020). *La producción en comunidades FLOSS: empresas, fundaciones y governance*. En *LA ECONOMÍA DEL SOFTWARE LIBRE Y OPEN SOURCE: MULTINACIONALES, PYMES Y COMUNIDADES*. Ciudad Autónoma de Buenos Aires (Argentina): Estudios Sociológicos Editora.

Dirección estable: <https://www.aacademica.org/agustin.zanotti/41>

ARK: <https://n2t.net/ark:/13683/p6uq/neA>



Esta obra está bajo una licencia de Creative Commons.  
Para ver una copia de esta licencia, visite  
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>.

*Acta Académica* es un proyecto académico sin fines de lucro enmarcado en la iniciativa de acceso abierto. *Acta Académica* fue creado para facilitar a investigadores de todo el mundo el compartir su producción académica. Para crear un perfil gratuitamente o acceder a otros trabajos visite: <https://www.aacademica.org>.

## **Capítulo 2. La producción en comunidades FLOSS: empresas, fundaciones y *governance***

*Juan Gabriel Vélez y Agustín Zanotti*

El capítulo constituye un acercamiento al funcionamiento de las comunidades FLOSS, abordando sus aspectos productivo y organizativo. Los estudios existentes han puesto el foco en diferentes aspectos de las comunidades: las interacciones entre los miembros, las diferentes participaciones y formas de membresía, el papel de las empresas, las motivaciones para la participación voluntaria, entre otros. El objetivo en este capítulo es indagar, a partir de la revisión de literatura y de fuentes secundarias, los modos de *governance* al interior de comunidades a partir de la articulación de proyectos FLOSS y empresas.

La importancia del sector informático en el mundo y en particular en Argentina, lleva a la necesidad de comprender el funcionamiento de las comunidades FLOSS. En la actualidad, este cumple un rol decisivo en los desarrollos tecnológicos y en muchas innovaciones relevantes del sector. Los estudios sobre FLOSS requieren de herramientas analíticas que permiten abordar este fenómeno emergente y sus potencialidades.

Podemos definir a las comunidades FLOSS como instancias de producción entre pares, definida 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). Las comunidades desempeñan funciones organizacionales “clásicas” como: coordinación, división del trabajo, reclutamiento, entrenamiento, creación y cumplimiento de normas, resolución de conflictos y mantenimiento de fronteras, pero lo hacen de un modo diferente.

Indagar las relaciones de poder hacia el interior de las comunidades FLOSS abre una serie de preguntas: cuál es el rol que juegan las fundaciones; cómo

funcionan los mecanismos de mérito y status en la toma de decisiones de los proyectos comunitarios; qué estrategias se dan las empresas para incidir en las comunidades. Para ello nos enfocamos en estudiar comunidades FLOSS en particular: los casos de GNOME, Fedora, Python y LibreOffice.

Este capítulo se compone de una revisión de literatura y una exploración de casos referidos a comunidades FLOSS. En las primeras dos secciones se analizan los modos de organización habituales al interior de este tipo de comunidades y las relaciones que se tejen entre empresas y proyectos FLOSS.

En la exploración de casos, trabajamos con cuatro comunidades de desarrollo: GNOME (entorno de escritorio), Fedora (distribución), Python (lenguaje de programación) y LibreOffice (suite de oficina). Cada una alberga a su vez numerosos sub-proyectos. La manera de pensar estas estructuras es la de una *ecología de sub-comunidades* (Weiss *et al.*, 2006), antes de grandes comunidades homogéneas. Esta idea permite visualizar las interrelaciones que se producen al interior de estas comunidades, donde se comparte una gobernanza común, y existe una activa interdependencia, ya que hay artefactos desarrollados de uso común.

Caracterizamos de manera exploratoria los modos de gobernanza de estas comunidades, y el rol que ocupan las fundaciones en apoyo a los proyectos. Los datos utilizados provienen de fuentes secundarias, de análisis previos sobre los estudios de casos y sitios oficiales de las comunidades consultadas.

### **1. Comunidades FLOSS**

Según Benkler (2017), el FLOSS puede incluirse en un grupo de bienes comunes informacionales. Se trata de una familia de arreglos institucionales que ocupan un rol central en las economías actuales, mayor a la atención que han recibido de parte de la ciencia económica. Ellos se basan en derechos de uso simétricos, por oposición a derechos exclusivos asimétricos.

Los bienes comunes informacionales conllevan implicancias para las teorías de la innovación. En general se presentan en escenarios donde existen:

**(a) externalidades altas persistentes:** innovación incremental, efecto “sobre hombros de gigantes”, derrame de conocimiento (*spillovers*), beneficios que superan el mayor uso.

**(b) incertidumbre respecto al mercado o los usos:** entornos cambiantes, diversos, imprecisos o necesidad de compartir riesgos;

(c) **finalidades sociales:** el modelo *mertoniano* se impone sobre el managerial, diversidad de motivaciones: investigación, flexibilidad, estatus, placer de investigar (Benkler, 2017).<sup>1</sup>

En ese sentido, algunos autores sostienen que una dimensión clave de la producción de FLOSS es la colaboración que se establece en el marco de una economía de regalo o intercambio de dones (en inglés, *gift economy*). La idea, con origen en la antropología, se basa en la reciprocidad mutua que genera entre las partes una lógica de dones y contradones. Con el crecimiento de la importancia comercial de los proyectos FLOSS, este modelo economía transitó hacia una hibridación con la economía de mercado (Berdou, 2010).

Desde un abordaje más cercano a la teoría económica, Hippel y Krogh (2003) refieren un modelo de innovación privado-colectivo, como una combinación del modelo de la inversión privada y la innovación de acción colectiva. El modelo se basa en el supuesto de que los innovadores que crean bienes de acceso público se benefician más que los *free-riders* que solamente consumen dichos bienes. Si bien el resultado de la inversión es igualmente accesible a todos, los innovadores se benefician a través del proceso de creación de estos bienes, en donde adquieren conocimientos, habilidades, reconocimiento y participan en redes que pueden luego capitalizarse.

La comunidad FLOSS está conformada por un ecosistema de actores que llevan a cabo proyectos de software de manera descentralizada. Cada comunidad puede ser conformada o bien por un puñado de programadores, o bien cientos o miles de ellos a nivel global, conectados a través de redes y plataformas de producción colaborativa. En este trabajo nos enfocamos exclusivamente en las comunidades de desarrollo de software, dejando de lado otros espacios de usuarios o activistas de cultura libre, en ocasiones vinculados a la producción informática.

Los desarrolladores FLOSS liberan versiones estables luego de cada ciclo de desarrollo, y en general mantienen accesibles versiones en desarrollo y el código fuente generado. El código utiliza licencias no restrictivas, pudiendo a grandes rasgos distinguirse entre licencias recíprocas (también conocidas como *virales*), que obligan a mantener las condiciones de código abierto, y licencias permisivas,

---

1- Nos referimos aquí al *ethos comunista* de la ciencia enunciado por Merton, que define al conocimiento científico como una propiedad colectiva producto de la colaboración social, donde se aprecian la honestidad, el reconocimiento y la originalidad (Merton, 1977).

que habilitan desarrollos cerrados.<sup>2</sup>

Los desarrollos surgen de una persona o grupo que busca implementar una solución particular, y va integrando el aporte de un creciente número de desarrolladores y personas encargadas de otras tareas conexas. De esta forma pueden generarse estructuras organizativas complejas, institucionalizadas en grandes fundaciones incluso. Pero también es posible que un proyecto FLOSS sea iniciado por una firma, que puede retener el control y la administración del proceso, o bien ampliar la participación a otros actores.

Es común encontrar en la literatura de FLOSS la descripción de la estructura organizativa de los proyectos dividida en dos grupos principales, uno denominado *core* y el otro periferia. Denominado en la literatura como el *onion model* o modelo cebolla. También se refiere como regla 90-9-1 de desigualdad participativa, en la que los usuarios activos (9%) y muy activos (1%) constituyen la menor parte, mientras la mayoría sólo observa o consumen información (Nielsen, 2006).

En el *núcleo del proyecto* se encuentran los líderes o mantenedores, a menudo iniciadores, y los principales contribuidores. Las actividades que realizan son aceptar y rechazar las modificaciones al código fuente que son realizadas por la periferia. Además, se encargan del mantenimiento de la calidad del software y el establecimiento de mecanismos de selección de mejoras (Weiss *et al.*, 2006).

Por su parte, la periferia está formada por un amplio número de colaboradores activos – cuya función es el testeado del software, la detección e información de errores, y la generación de mejoras o parches del código fuente– y pasivos, que sólo hacen uso de los sistemas. También en la periferia se ubican una serie de actividades como la elaboración de documentación o traducciones, que son necesarias para facilitar el acceso y reclutamiento de nuevos miembros. En general este trabajo se considera de inferior jerarquía y los no-programadores difícilmente asumen roles centrales en la comunidad (Berdou, 2010).

La división de los participantes en esta organización no es estructurada, sino que se trata de sociedades meritocráticas, donde los miembros comparten una cultura y unas normas que hacen al consenso del uso y explotación que se puede hacer del software producido, de las licencias implicadas, sus actualizaciones, etcétera. Los proyectos funcionan sin un proceso formal de recruiting. La relación entre el proyecto y los desarrolladores es voluntaria y no mediada por contratos

---

2- Un repaso acerca de los tipos de licencias y sus modificaciones recientes se encuentra disponible en Wheeler (2017), así como en el Capítulo 1 de este libro.

formales. Más allá de eso, quienes se vinculan con estos espacios pueden poner en valor indirectamente sus contribuciones, ya sea como una forma de capital social, tecnológico o simbólico (Zanotti, 2014).

## 2. Empresas y la gobernanza FLOSS

Algunas ventajas del modelo de producción FLOSS fueron enunciadas por el modelo bazar y la ley de Linus (Raymond, 1999). Con el tiempo, el interés de las empresas se proyectó en varias comunidades, desarrollando formas de vinculación diferentes.<sup>3</sup> Capra *et al.* (2011) distinguen entre proyectos comunitarios o comerciales respecto a quien es el iniciador. Si bien en ambos la participación de empleados de empresas es significativa, en los proyectos comerciales son las empresas las que disponen del acceso al código base y definen la estrategia de evolución del proyecto. Proponen tres modelos para comprender este involucramiento que agrupan las diversas formas en que las empresas participan de las comunidades.

- *Modelo de aporte de código*: principalmente a través de la escritura de código y actividades relacionadas; resolución de *bugs*, la personalización de software o provisión de distribuciones empaquetadas particulares. También comprende situaciones donde las empresas donan una base de código pre-existente, abriendo su código; o bien lanzan un proyecto nuevo desde cero fundando la comunidad, y luego aportando las tareas de codificación descritas. Este es un modo indirecto de controlar los desarrollos de la comunidad, ya que haber hecho aportes de códigos otorga cierto poder sobre la coordinación del proyecto.

- *Modelo de management*: tareas de coordinación y *management*, especialmente durante el período de aparición de una comunidad. Sugerencia de requerimientos y de funcionalidades; planificación y diseño de la aplicación o simplemente la coordinación del desarrollo. Pueden tener el objetivo de direccionar el desarrollo del producto para ganar ventajas competitivas, o ser capaz de establecer acuerdos de cooperación con competidores en áreas de negocio estratégicas y diferenciadas. Asumen tareas de administración y de coordinación.

---

3- La Ley de Linus establece que “Dada una base suficiente de desarrolladores asistentes y beta-testers, casi cualquier problema puede ser caracterizado rápidamente, y su solución ser obvia al menos para alguien”. O, dicho de manera menos formal, ‘con muchas miradas, todos los errores saltarán a la vista’” (Raymond, 1999).

· *Modelo de soporte y apoyo a los proyectos OS de la comunidad*: provisión directa o indirecta de apoyo financiero, apoyo logístico con recursos de hardware, alojamiento web, apoyo de marketing a través de publicidad en sus webs o motores de búsqueda. Puede incluir participar de los foros *online* para mantener activa la comunidad, *testing* del producto a través del uso en un ambiente de producción real, oferta de cursos de entrenamiento para usuarios, reportes de *bugs* a diario o escritura de documentación, manuales de usuario o guías de instalación.

Las empresas participan en proyectos comunitarios pensando con un sentido estratégico de cómo intervenir en la dirección de determinada tecnología. Tales modelos no se plantean como excluyentes. Las empresas pueden combinar estos tres tipos de modelo de involucramiento en proyectos de la comunidad, participando en múltiples niveles.

El modelo de aporte de código, es el que debiera recoger la relación que se teje con la comunidad por medio del reclutamiento de programadores que pasan a ser empleados de empresas. Sin embargo, en el análisis concreto de comunidades como GNOME, tal como veremos más adelante, se establecen complejidades que emergen en este proceso de interpenetración entre las relaciones sociales y comerciales-económicas (Berdou, 2010).

Esto ha afectado los mecanismos de gobernanza de las comunidades. El concepto de 'gobernanza' difiere de 'gobierno' en ser un sistema de conducción que involucra una gama de actores y redes más amplias que las existentes en una organización estructurada jerárquicamente. En el caso del FLOSS, las principales características de la gobernanza están dadas por: la apertura, la creación de redes, la participación y la transparencia en los proyectos. Estos proyectos no operan en jerarquías estrictas de comando y control, sino "en un entorno mucho más flexible que (...) permite la existencia de múltiples equipos de participantes que trabajan simultáneamente en una variedad de direcciones, inclusive opuestas" (P2PF, 2019). Esto ha dado lugar al concepto de gobernanza de pares:

En una empresa tradicional, es la administración, encabezada por el CEO y la junta directiva, la que controla la empresa y proporciona el ímpetu para el desarrollo continuo. Si bien la visión del liderazgo es igualmente integral para el eventual éxito de cualquier proyecto de código abierto, su control continuo es más frágil y depende de su relación y respuestas con la comunidad. (Nyman y Lindman, 2013)

Nyman y Lindman (2013) identifican tres niveles de gobernanza, referidos a 1. el desarrollo de software propiamente dicho, 2. la gestión de la comunidad y 3. el ecosistema en que se inserta. Hacen hincapié en la posibilidad de realizar bifurcaciones (*forks*), los cuales implican tomar el código e iniciar una escisión respecto del proyecto original. En la medida en que los *forks* implican según el caso oportunidades o dificultades, son un principio regulador de la gobernanza. La misma marca de este modo relaciones de poder hacia el interior de los desarrollos, especialmente en la toma de decisiones y el peso relativo que pueden mantener actores empresariales, así como la identidad y la influencia que implica ciertos límites para actores externos.

Evangelia Berdou (2010) propone examinar el funcionamiento organizativo de las comunidades FLOSS para penetrar en el análisis de las relaciones que se tejen en estos modos de producción. A partir de aportes teóricos de la sociología económica como el concepto de *embeddedness*, el desarrollo de tecnologías para el gobierno y la noción de comunidad de práctica, establece una crítica a ciertos estudios *sobresocializados*.<sup>4</sup> Para salir de esta dicotomía, Berdou establece el análisis de las comunidades en su relación mercantil, en las dinámicas de mercado y flujos monetarios donde están insertas (*embedded*).

A partir de ello propone la noción de *tecnologías de comunidad* como abordaje al fenómeno de las comunidades FLOSS. Estas tecnologías vendrían a estar constituidas por tres dimensiones: i) un programa meritocrático que estructura el modo de organización; ii) la invocación a la idea de comunidad para movilizar recursos; iii) las herramientas y técnicas para la gestión de la comunidad. Con este aparato teórico se propone la comprensión del sentido social en el desarrollo de FLOSS, identificando reglas, procesos y prácticas que dan sustento a definiciones específicas de mérito y status, que son claves en el acceso y control de la comunidad (Berdou, 2010).

Berdou evidencia la estrategia de las empresas de contratar desarrolladores reconocidos en la comunidad. En estas comunidades, los *maintainers* empleados por firmas tienen acceso a aspectos claves del desarrollo de los proyectos. Al contar con recursos externos, estas personas tienen mayores chances de invertir tiempo

---

4- Esto se refiere a la crítica de Mark Granovetter (1992) de aquellos estudios que se proponen como alternativos a los mecanicismos propios de la economía convencional, pero que caen en otro tipo de determinismo. En este caso, el error sería pasar a una visión donde los procesos sociales, las normas y valores de la comunidad FLOSS, son determinantes de la conducta de las personas, perdiendo de vista las estructuras políticas y económicas existentes.



en consolidar su posición en la comunidad, asistir a los eventos sociales, mejorar sus habilidades técnicas y asumir roles de mayor responsabilidad (Berdou, 2010).

En parte, esta presencia de las firmas por medio de la contratación genera beneficios en las comunidades porque permite retener a personas capacitadas y comprometidas. También les permite a las empresas contar con mayor capacidad de incidir en los procesos de desarrollo. Sin embargo, se establecen ciertos límites a esta capacidad de influenciar de las firmas.

Mientras más establecida esté una comunidad, es mayor la cantidad de firmas que colaboran, y menos probable es que una en particular tenga capacidad de capturar la agenda de desarrollo. También es menos probable que los desarrolladores empleados sean capaces de llevar a cabo las agendas comerciales de sus empleadores. Incluso, pueden desarrollar estrategias para evitar tomar trabajos que puedan ir en contra de los intereses de la comunidad. Algunas de estas tensiones son recogidas y reguladas por medio de la acción de las fundaciones.

### **3. Fundaciones FLOSS**

La tendencia constatable en los principales proyectos FLOSS es unirse a una fundación existente, o bien la creación de una propia. Hay dos tipos de organizaciones sin fines de lucro en la comunidad FLOSS. La primera es la fundación de proyectos grandes. Los casos más conocidos y estudiados son la *Linux Foundation*, *Apache Software Foundation* o la *Eclipse Foundation* (Riehle y Berschneider, 2012). Otros ejemplos se refieren a nichos o proyectos intermedios, como la *GNOME Foundation* o la *Document Foundation*. Estas apoyan y trabajan en estrecha colaboración con un gran proyecto institucionalmente independiente.

El segundo tipo se refiere a la promoción o cuestiones de interés para muchos o todos los proyectos de software libre. La *Open Source Initiative*, *Software Freedom International* y la *Free Software Foundation* son ejemplos de este tipo. Esta última también apoya el Proyecto GNU, por lo que se puede clasificar en ambos campos (P2PF, 2019).

Una fundación tiene muchos beneficios: actúa como representante legal ante conflictos, protege el aporte individual de los desarrolladores; le da credibilidad al proyecto, y lo hace más independiente de personas en particular, lo que aumenta el interés empresarial y le otorga sostenibilidad, permite recolectar fondos (Riehle y Berschneider, 2012). Las fundaciones además patrocinan el desarrollo de proyectos, coordinan esfuerzos, proveen apoyo en asuntos legales y financieros, y gestionan las relaciones con las empresas (Berdou, 2010).

Existen una gran cantidad de fundaciones, cientos de ellas. La alta heterogeneidad respecto a las configuraciones institucionales y sus modos de funcionamiento busca ser captado en la propuesta de Riehle y Berschneider. Los autores proponen seis bloques en su modelo de fundaciones:

**Tabla 2.1. Modelo de fundaciones de desarrollo FLOSS**

<b>Categorías</b>	<b>Atributos</b>
<b>General</b>	Propósito
	Constitución
	Miembros
<b>Filosofía</b>	Postura de Open Source
	Modelo de desarrollo
<b>Propiedad Intelectual</b>	Licencia del proyecto
	Concesión de licencia de patente
	Propiedad de derechos de autor
<b>Gobernanza</b>	Membresía de la junta
	Membresía del proyecto
	Carrera de una persona dentro de la fundación
	Participación de personas jurídicas
<b>Financiamiento</b>	Financiamiento
	Proyectos
<b>Operaciones</b>	Infraestructura
	Back office

Elaboración Propia en base a Riehle y Berschneider (2012)

En el aspecto más formal (General) se establece la pertenencia a la fundación, si va a ser para personas naturales o jurídicas; si el beneficio de la fundación está planteado en términos del público general o sólo sus miembros. Luego hay dos bloques de posturas en torno al desarrollo de software y derechos intelectuales (Filosofía y Propiedad Intelectual).

Respecto de la Gobernanza, establecen las posibles diferencias en torno al modo de elección de la junta directiva. Puede ser democrática, autocrática o

meritocrática, de igual manera que la decisión de pertenencia de un proyecto a la fundación. Por otra parte, se establece el esquema de evolución que puede tener dentro de la fundación, una persona natural o jurídica, según sea el caso. Esta trayectoria describe idealmente el modo en que se puede asumir roles cada vez más importantes en la estructura de decisión.

En relación al financiamiento de las fundaciones, hay cuatro mecanismos habituales: i) pago por membresía; ii) *sponsorship* (patrocinio); iii) donaciones y subvenciones; y iv) ingresos provenientes de una empresa subsidiaria, que es el menos habitual. La última dimensión es Operaciones, la cual implica el modo de gestión cotidiana de la fundación y el uso de infraestructura propia o la que aportan los miembros.

#### 4. *Análisis de casos*

A continuación presentamos cuatro casos de proyectos de desarrollo FLOSS, a partir de los cuales podemos analizar trayectoria de gobernanza. Consideramos que la misma depende del tipo de desarrollo que se trate, motivo por cual incluimos:

- **un entorno de escritorio:** es una implementación de la metáfora del escritorio conformada por un conjunto de programas que se ejecutan en la parte superior de un sistema operativo, que comparten una interfaz gráfica de usuario (GUI) común. Permite los usos más habituales y la configuración del hardware del dispositivo. Generalmente cuenta con utilidades y aplicaciones para la gestión, visualización y edición de diferentes formatos, además de conectividad y navegación en red.

- **una distribución GNU/Linux:** es un sistema operativo hecho de una colección de librerías de software, basadas en el kernel de Linux y, a menudo, en un sistema de administración de paquetes. Las distribuciones se basan en el proceso permanente de empaquetado de una diversidad de proyectos, proveyendo a sus usuarios de nuevas versiones y actualizaciones de seguridad.

- **un lenguaje de programación:** es un lenguaje formal que proporciona una serie de instrucciones que permiten a un programador escribir secuencias de órdenes y algoritmos que a modo de controlar el comportamiento de un dispositivo electrónico, con el objetivo de que produzca diversas clases de datos y resultados.

- **una suite ofimática:** es un conjunto de aplicaciones utilizadas en tareas de oficina, para crear, modificar, organizar e imprimir documentos, planillas de cálculo, presentaciones y bases de datos, entre otras.

#### 4.1. GNOME

*GNU Network Object Model Environment* (Entorno de Modelo de Objeto de Red), conocido como GNOME, es uno de los entornos de escritorio más populares para sistemas operativos Unix. El proyecto fue iniciado por los programadores mexicanos Miguel de Icaza y Federico Mena en octubre de 1997. GNOME está disponible en las principales distribuciones GNU/Linux, siendo en algunas el escritorio por defecto. El proyecto es dirigido por la *GNOME Foundation* y pone un fuerte énfasis en la filosofía de software libre, formando parte oficial del proyecto GNU.

Los desarrollos de GNOME se organizan a través del Proyecto GNOME. Este contempla dos ciclos de desarrollo anuales, que se mantienen regulares desde 2002, publicando ediciones estables en marzo y septiembre (GNOME, 2019b). El escritorio alcanza en 2019 en su versión 3.34 y se encuentra en preparación la versión 4.0. Su repositorio Github cuenta al momento con 429 subproyectos vinculados, que forman parte del conjunto de sus componentes y aplicaciones *core* y *extras* (Gnome Github, 2019).

El GIMP ToolKit (GTK+) es la base de GNOME. Desde sus inicios, este kit de herramientas usa una Licencia Pública General Reducida (LGPL) de GNU, que permite que el software que se vincule con ella utilice un conjunto mucho más amplio de licencias, incluidas licencias de software propietario. Además, GNOME está autorizado bajo la LGPL para sus bibliotecas y Licencia Pública General de GNU (GPL) para sus aplicaciones (GNOME, 2019d).

En el año 2000 se creó la GNOME Foundation. De acuerdo a su Carta de fundación, entre sus objetivos encontramos:

- La Fundación coordinará las versiones de GNOME y determinará qué proyectos forman parte de GNOME.
- actuará como una voz oficial para el proyecto GNOME, proporcionando un medio de comunicación con la prensa y con organizaciones comerciales y no comerciales interesadas en el software GNOME.
- puede producir materiales educativos y documentación para ayudar al público a aprender sobre el software GNOME.
- puede patrocinar conferencias técnicas relacionadas con GNOME y representar a GNOME en conferencias relevantes patrocinadas por otros,
- ayudar a crear estándares técnicos para el proyecto y promover el uso y desarrollo del software GNOME. (GNOME, 2019a)

La Fundación está compuesta por contribuidores que pertenecen como

miembros y tienen el derecho de postularse a cargos ejecutivos en la junta directiva, así como participar en la elección anual. La membresía es personal y la junta directiva es el organismo que gobierna la fundación. Cuenta además con una junta asesora formada exclusivamente por empresas y organizaciones. Las pequeñas firmas hacen un pago anual de U\$S11.500 y las medianas y grandes de U\$S 23.000. Actualmente integran la junta asesora Canonical, Debian, Endless, FSE, Google, Private Internet Access, Red Hat, Sugar Labs, SUSE, The Document Foundation, System76.

Esta junta asesora se encarga de representar los intereses de estas organizaciones, reuniéndose regularmente con la junta directiva para discutir la dirección de los proyectos y prestar apoyo. Esto último es en relación a asuntos de marketing, de respuestas de los clientes, finanzas y prestar otros tipos de recursos. 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, 2011).

En lo que respecta al financiamiento, los pagos provenientes de la junta asesora representan una porción menor de los ingresos. La mayor parte proviene de las donaciones. El proyecto llegó a recibir donaciones anónimas de 1 millón de dólares en 2018 (GNOME News, 2018 25 de mayo).

La GNOME Foundation 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, 2019f). Además, participa de becas de programación como *Google Summer of Code*, programa Desde 2006, lleva un programa de pasantías con rondas organizadas dos veces al año. Su objetivo es involucrar a personas de grupos subrepresentados, tanto por su lugar de pertenencia, su etnicidad o su diversidad de género (GNOME, 2019e). También patrocina y organiza eventos, como GUADEC, que es la conferencia anual de los usuarios y desarrolladores de GNOME.

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).

#### 4.2. Fedora

Fedora es una distribución basada en GNU-Linux extendida entre los usuarios de software libre. El Proyecto Fedora surgió en 2003 bajo el patrocinio de Red Hat Inc., firma líder dedicada al desarrollo, implementación y soporte de tecnologías libres. Opera de este modo una articulación con el espacio comunitario, en el que la empresa provee de código, infraestructura y recursos, mientras se beneficia de la innovación, experimentación y el desarrollo motorizado por sus usuarios. Recientemente, Red Hat fue adquirida por IBM en julio de 2019.

La última versión estable es Fedora 30, publicada en abril de 2019. En cuanto a su gobernanza, el proyecto atravesó diferentes modalidades organizativas a lo largo de su historia, las cuales evidencian el lugar de la empresa y la comunidad en las decisiones y dirección del desarrollo.

El proyecto comenzó organizándose con la idea de una fundación, la *Fedora Foundation*. La misma se definió con los objetivos de:

- proporcionar una entidad sin fines de lucro para organizar y administrar voluntarios.
- garantizar que el trabajo de estos voluntarios permanezca siempre libre.
- proporcionar un brazo de recaudación de fondos para el desarrollo y protección de Fedora y proyectos de código abierto relacionados.
- proporcionar una entidad para la asignación de derechos de autor, de modo que lo que es libre también fuera defendible en un tribunal de justicia.
- financiar solicitudes de patentes para inventores en la comunidad de código abierto, de modo que individuos dedicados puedan ayudar a construir un escudo protector de patentes alrededor del código de código abierto. (Fedora Project Sitio web, 2019)

La entidad legal fue creada en 2005 con estatutos muy básicos y flexibles, y la designación de una junta para que lo ejecute temporalmente. Pero al poco tiempo quedó en evidencia que no se adecuaba a los requerimientos del proyecto. En 2006, la fundación fue reemplazada por la *Fedora Project Board*, que incluía tanto Red Hat como liderazgo comunitario voluntario. La Junta de Fedora comprendía cinco miembros nombrados por Red Hat y cinco miembros elegidos por la comunidad. Además, el líder del Proyecto Fedora tenía poder de veto sobre cualquier decisión de la junta.

Eso, a su vez, fue reemplazado por el *Fedora Council* en 2014. El Consejo de Fedora es actualmente el organismo de liderazgo y gobierno comunitario de alto

nivel. El Consejo está compuesto por una mezcla de representantes de diferentes áreas del proyecto, roles designados por Red Hat y un número variable de escaños relacionados con los objetivos del proyecto a mediano plazo. En el modelo actual, todos los miembros con derecho a voto pueden bloquear asuntos, con una razón válida, y se insiste en un proceso de establecimiento de consensos (Fedora Docs, 2019).

La comunidad también está involucrada en la organización de niveles más bajos de liderazgo, tanto el *Engineering Steering Committee* (Comité Directivo de Ingeniería) como el *Mindshare Committee*, responsables de la supervisión técnica y comunitaria, respectivamente. Ambos son organismos elegidos por la comunidad que administran porciones significativas del proyecto.

El proyecto facilita la comunicación en línea entre sus desarrolladores y miembros de la comunidad a través de listas de correo públicas y páginas wiki. También coordina dos eventos principales, conocidos como la *Conferencia de Usuarios y Desarrolladores de Fedora* (FUDCon) y *Flock* (o Flock to Fedora). FUDCon es un evento que se realiza en diferentes ubicaciones en las dos regiones designadas, América Latina y Asia/Pacífico. Por lo general, gira en torno a alguna combinación de conferencias, eventos sociales y un hackathon (Fedora Project Wiki, 2019b). Flock es un evento similar que reemplazó a FUDCon en Norteamérica y Europa/Medio Oriente (Flock To Fedora, 2019). Los eventos programados de manera flexible y centrados en tareas, conocidos como *Fedora Activity Days*, también reúnen a muchos contribuyentes en varias regiones.

A su vez, grupos locales se organizan para promover la comunidad. Son precedidos por un Community Manager y embajadores, “contribuyentes experimentados del proyecto Fedora cuya responsabilidad es la de representar, promover y expandir el Proyecto Fedora y sus ideales a la gran comunidad de software libre y de código abierto, y al público en general” (Fedora Project Wiki, 2019a). El proyecto cuenta con comunidades en diferentes países de la región incluyendo: Argentina, Brasil, Chile, Colombia, El Salvador, Guatemala, México, Nicaragua, Panamá, Paraguay, Perú, Venezuela y Uruguay

#### 4.3. Python

Python es un lenguaje de programación multiparadigma creado a finales de los ochenta. Alcanzó la versión 1.0 en enero de 1994 y en la actualidad se encuentra en su versión 3.7, con cambios significativos respecto a su versión anterior. Python ha ganado popularidad en el último tiempo, destacando en

áreas como desarrollo web, desarrollo de interfaces de usuario, análisis científico y numérico (incluyendo aprendizaje automático, big data), desarrollo de software y administración de sistemas. El desarrollo de este lenguaje se organiza de forma comunitaria. El repositorio oficial mantiene unos 191 mil proyectos, algunos mantenidos en forma oficial y otros producidos de manera independiente por la comunidad (PyPI, 2019).

El lenguaje posee licencias de código abierto protegidas a nivel global por la Python Software Foundation (PSF), encargada de gestionar las licencias de código abierto para Python versión 2.1 y posteriores, junto a las marcas comerciales asociadas con Python. La fundación mantiene la infraestructura, el sitio web el índice de paquetes, la documentación de Python y muchos otros servicios en los que se basa la comunidad (PSF, 2019a).

La principal fuente de fondos para el funcionamiento de la fundación proviene del aporte de organizaciones como *sponsor*. Para ello, debe haber aprobación por parte de un comité y se cuentan con diversos niveles de aportes (desde 500 a 150.000 dólares al año).

También organiza anualmente la conferencia PyCon de América del Norte, apoya a otras conferencias de Python en el mundo y financia el desarrollo relacionado con Python, con subvenciones y financiamiento para proyectos especiales. Ha contribuido económicamente con proyectos relacionados con el desarrollo de Python, tecnología relacionada con Python y recursos educativos en más de 50 países (PSF, 2019b).

La expansión de Python a nivel mundial fue acompañado de la proliferación de grupos de usuarios locales, como lo es Python Argentina (PyAr): “Nuestro objetivo es nuclear a los usuarios de Python. Pretendemos llegar a personas y empresas, promover el uso de Python e intercambiar información.” (PyAr Sitio web, 2019)

El grupo comenzó a organizarse en 2004 a partir de entusiastas que venían utilizando este lenguaje para propósitos diversos, quienes comenzaron a reunirse mensualmente en Capital Federal para dar forma al espacio, pautar actividades y conformar la lista de correo. PyAr realiza actividades de difusión, desarrollo y soporte. Entre ellos se destacan los *Pycamps*, *Python bug days*, *Pydays*, campañas y más. También comenzó en los últimos años a editar una publicación digital sobre este lenguaje. Sus miembros mantienen a su vez una variedad de proyectos de desarrollo que se nutren del soporte comunitario.



PyAr llegó así a ser la mayor comunidad Python de habla hispana a nivel global y el grupo especializado en un lenguaje de programación más grande del país. La comunidad argentina fue la primera en organizar una *PyCon* (Conferencia de Python) en la región, con presencia de invitados extranjeros destacados.<sup>5</sup>

#### 4.4. LibreOffice

LibreOffice es una suite de oficina que nació como un *fork* de otro proyecto llamado OpenOffice. Este a su vez tenía como código base a la suite ofimática StarOffice, cuyo desarrollo inicial data del año 1985.

Inicialmente los desarrollos eran realizados en el marco del proyecto bajo el liderazgo de Sun Microsystems. La gobernanza del proyecto generó una serie de fricciones en torno a la aceptación de aportes y la relación con empresas, como IBM, que fue mellando la confianza en el proyecto. Esta situación llegó a un límite cuando Sun es comprada por Oracle, empresa que ya había tenido comportamientos negativos con respecto a la comunidad FLOSS. Por ello, y ante la posibilidad de la discontinuidad o la imposición de mayores restricciones, un grupo de desarrolladores decide abandonar el proyecto OpenOffice para dar nacimiento a LibreOffice.

El proyecto LibreOffice es alojado en la Document Foundation, una organización sin fines de lucro con sede en Alemania creada en 2010. Su objetivo es producir una suite ofimática independiente del proveedor con soporte ODF en un entorno de desarrollo libre del control de la compañía (Document Foundation, 2019).

El proyecto es gobernado por la comunidad y pone un fuerte énfasis en la filosofía de software libre, reflejado en el respeto de las cuatro libertades básicas (referidas en el Capítulo 1). Asimismo, incluyen en su visión una serie de valores como el respeto a las lenguas nativas, al empoderamiento de los ciudadanos, acceso al software sin costo. Mantienen una postura de desarrollo en torno al software libre ligada a la creencia de que puede proveer *mejor calidad, confiabilidad, seguridad y flexibilidad que las alternativas propietarias* (Document Foundation, 2019).

Esta fundación es controlada por la propia comunidad de LibreOffice, teniendo la función de proteger los desarrollos vía el uso de licencias de código abierto o

---

5- Algunos casos de empresas argentinas con un modelo de negocios basado en este lenguaje son presentados en el Capítulo 4.

FLOSS, recíprocas (como GNU Lesser GPL) y no recíprocas (Mozilla Public License), así como de contenidos (Creative Commons Attribution-ShareAlike License).

En lo que refiere al aspecto organizativo, los miembros son personas que ingresan a título individual. Cuenta con una Junta Directiva, que representa la fundación y lleva la administración general de proyectos y equipos. Pueden nombrar cargos para que lleven a cabo aspectos administrativos o de funcionamiento básico de la fundación. Además tiene un Comité de Membresía, que administran solicitudes y renovaciones de membresías, así como supervisar la elección de los miembros de la Junta Directiva. Y un *Board of Trustees*, compuesto por todos los miembros que contribuyen activamente al proyecto. Hay que hacer una aplicación para poder pertenecer y se siguen principios meritocráticos, no admitiéndose exclusivamente donaciones para poder pertenecer.

Luego está la dirección de ingeniería (*Engineering Steering*), que es un cuerpo que presta apoyo a la Junta pero no de manera formal. Está compuesto por aquellos que tienen las mayores capacidades técnicas.

El Consejo Asesor, formado por un conjunto de organizaciones que contribuyen al financiamiento y prestan consejo a la fundación. Actualmente lo integran la Free Software Foundation, BMP-Conseil, KDE, la Ciudad de Munich, Kopano, los UK Government Digital Services, Red Hat, Software in the Public Interest, CIB Software, Collabora, RPA RusBITech, Adfinis SyGroup, GNOME y Google.

## **5. Discusión**

Caracterizamos de manera exploratoria cuatro comunidades y sus modos de gobernanza, para encontrar particularidades en sus trayectorias. Podemos observar la relación entre el origen y manejo de los proyectos, así como las formas institucionales que asume su gobierno. Cabe resaltar que la gobernanza se relaciona también con el tipo de proyecto de que se trate, habilitando diferentes conexiones y modelos de negocio con empresas. De acuerdo con Scharschmidt y Von Kortzfleisch (2015), los proyectos FLOSS pueden surgir de diversas maneras. Algunos de ellos son iniciados por una persona o un grupo de personas, que van creciendo hasta volverse una comunidad de desarrolladores, mientras otros son iniciados por empresas. A su vez, respecto de la administración y control, los proyectos FLOSS pueden tener detrás a una única firma o a una multiplicidad de firmas.

En el caso de GNOME, se trata de un escritorio surgido con fuertes ideales libertarios y devenido una de las principales herramientas FLOSS. Su modelo de fundación mantiene un principio democrático en la elección de las autoridades, pero permite la capacidad de incidir de empresas y ubica en roles jerárquicos de los proyectos a desarrolladores pagos. La fundación canaliza recursos tanto de empresas líderes de mercado como emprendimientos de menor escala.

La distribución Fedora resulta de interés porque es uno de los casos en que el modelo de fundación fue dejado de lado en favor de un mayor control por parte de su compañía madre. El diseño de Junta y luego Consejo directivo dejó institucionalizado el predominio de la empresa, la cual cuenta sin embargo con un reconocimiento especial dentro del ecosistema FLOSS. La comunidad se mantiene, sin embargo, para el desarrollo, actividades de difusión y promoción.

En el lenguaje Python, la comunidad se organiza en dos niveles. El primero es el desarrollo de los módulos que constituyen el propio lenguaje, lo que aumenta sus prestaciones y eficacia, y los desarrollos liberados por la comunidad que hacen uso de este lenguaje. La fundación concentra sus esfuerzos en promover el primero de los componentes, mediante subvenciones y conferencias, obteniendo recursos del patrocinio. Mientras tanto, favorece la puesta en disponibilidad de muchas implementaciones, mediante un catálogo de desarrollos.

El caso de Libreoffice puede ser leído en oposición al de Fedora. Luego de sucesivos ciclos de desarrollo bajo la órbita de una empresa madre, el desarrollo es bifurcado (fork) hacia un nuevo proyecto liderado ahora sí por una fundación. Este último refuerza las definiciones libertarias asociadas al FLOSS y los formatos abiertos, y logra una impronta más comunitaria. De este modo, más allá del patrocinio de empresas líderes del sector, no hay firmas que tengan en la actualidad una influencia decisiva sobre la gestión de esta comunidad.

La Tabla a continuación resume las apreciaciones presentadas:

**Tabla 2.2. Clasificación de los proyectos de acuerdo a origen y administración**

Origen	Gestión - administración de los proyectos		
	Única empresa por detrás	Multiplicidad de empresas detrás	Relativa independencia de Firmas
Iniciado por firma/s - Proyecto Comercial	Fedora (Consejo)		LibreOffice (Fundación)
Iniciado por Comunidad FLOSS - Proyecto Comunitario		GNOME (Fundación) Python (Fundación)	

Elaboración propia en base a Schaarschmidt *et al.* (2015) y Capra *et al.* (2011).

## 6. Cierre

A lo largo del texto avanzamos en una caracterización de comunidades de software, asociadas a un modelo de desarrollo de bienes comunes informacionales basado en la colaboración masiva entre pares. Observamos en qué medida la proliferación del FLOSS en la industria ha impactado en los principales proyectos y sus modos de organización y dirección.

Definimos la gobernanza como una forma de gestión organizacional que se diferencia de las empresas tradicionales. El término gobernanza de pares apunta a una forma de control más participativa y abierta, que depende de su relación con la comunidad y el ecosistema en que se encuentran insertos los desarrollos de software.

Resulta relevante establecer el rol que juegan las fundaciones en estos procesos. Esta configuración ha llegado a ser habitual en los desarrollos FLOSS, regulando la multiplicidad de empresas detrás de los proyectos y generando, al mismo tiempo, una relativa independencia de firmas. El apoyo a fundaciones pueden ser parte de la estrategia de las empresas para involucrarse en la comunidad FLOSS, generando un marco de confianza para la participación de voluntarios. Establecen, a su vez, límites a la capacidad de apropiación y direccionamiento del trabajo de las comunidades por parte de empresas particulares.

Las estrategias de las empresas van más allá de las cuestiones de formalidad en las fundaciones, cómo se ve para el caso de contratación de desarrolladores de

peso en los proyectos FLOSS. Por ello la importancia de superar ciertos esquemas, para dar paso a nociones más complejas que den cuenta del modo de organizarse en las comunidades FLOSS.

### ***Fuentes Documentales***

- Document Foundation (2019). Overview. Recuperado de <https://www.documentfoundation.org/foundation/overview/> (Acceso: 9 de octubre 2019).
- Fedora Docs (2019). *Fedora Council Charter*. Recuperado de <https://docs.fedoraproject.org/en-US/council/> (Acceso: 9 de octubre 2019).
- Fedora Project (2019). *Foundation*. Recuperado de <https://fedoraproject.org/wiki/Foundation> (Acceso: 9 de octubre 2019).
- Fedora Project Wiki (2019a). *Ambassadors/es*. Recuperado de <https://fedoraproject.org/wiki/Ambassadors/es> (Acceso: 9 de octubre 2019).
- Fedora Project Wiki (2019b). *FUDCon*. <https://fedoraproject.org/wiki/FUDCon?rd=FUDcon> (Acceso: 9 de octubre 2019).
- GNOME (2019a). GNOME Foundation Charter Draft 0.61 (23 October 2000). Recuperado de <https://wiki.gnome.org/FoundationBoard/Resources/Charter> (Acceso: 9 de octubre 2019).
- GNOME (2019b). Release Notes. Recuperado de <https://help.gnome.org/misc/release-notes/> (Acceso: 9 de octubre 2019).
- GNOME (2019c). The GNOME Foundation. Recuperado de <https://www.gnome.org/foundation/> (Acceso: 9 de octubre 2019).
- GNOME (2019d). GNOME Foundation Guidelines on Copyright Assignment. Recuperado de <https://wiki.gnome.org/FoundationBoard/Resources/CopyrightAssignment/Guidelines> (Acceso: 9 de octubre 2019).
- GNOME (2019e). Outreachy/History. Recuperado de <https://wiki.gnome.org/Outreachy/History> (Acceso: 9 de octubre 2019).
- GNOME (2019f). GNOME Foundation facing lawsuit from Rothschild Patent Imaging. Recuperado de <https://www.gnome.org/news/2019/09/gnome-foundation-facing-lawsuit-from-rothschild-patent-imaging/> (Acceso: 9 de octubre 2019).
- GNOME GitLab (2019). *Details*. Disponible en: <https://gitlab.gnome.org/GNOME>
- GNOME News (2018, 25 de mayo). Anonymous Donor Pledges \$1M Donation Over Two Years. Recuperado de <https://www.gnome.org/news/2018/05/>

anonymous-donor-pledges-1m-donation-over-two-years/ (Acceso: 9 de octubre 2019).

Peer 2 Peer Foundation (P2PF) (2019). Peer Governance. Recuperado de [https://wiki.p2pfoundation.net/Peer\\_Governance](https://wiki.p2pfoundation.net/Peer_Governance) (Acceso: 9 de octubre 2019).

Python Software Foundation (PSF). (2019). Python Software Foundation. Recuperado de <https://www.python.org/psf-landing/> (Acceso: 9 de octubre 2019).

Python Argentina (PyAr). (2019a). Python Argentina. Recuperado de <https://www.python.org.ar/> (Acceso: 9 de octubre 2019).

PyAr (2019b). PSF Annual Report. Recuperado de <https://www.python.org/psf/annual-report/2019/> (Acceso: 9 de octubre 2019).

PyPI (2019). The Python Package Index. Recuperado de <https://pypi.org/> (Acceso: 9 de octubre 2019).