

La economía del software libre y open source: Multinacionales, Pymes y Comunidades

Hernán Morero y Jorge Motta
Editores

Morero, Hernán

La economía del software libre y open source : multinacionales, pymes y comunidades / Hernán Morero ; Jorge Motta ; editado por Hernán Morero ; Jorge Motta. - 1a ed. - Ciudad Autónoma de Buenos Aires : Estudios Sociológicos Editora, 2020.

Libro digital, PDF

Archivo Digital: descarga y online

ISBN 978-987-3713-40-8

1. Software Libre. 2. Sociología. I. Motta, Jorge. II. Título.

CDD 005

Diseño de Tapa: Romina Baldo

Diagramación y corrección: Juan Ignacio Ferreras

© 2020 Estudios Sociológicos Editora

Mail: editorial@estudiosociologicos.com.ar

Sitio Web: www.estudiosociologicos.com.ar

Primera edición: febrero de 2020.

Hecho el depósito que establece la Ley 11723.

Libro de edición argentina.

El presente libro puede ser descargado desde el sitio web de nuestra editorial

Indice

Introducción.....	11
--------------------------	-----------

Capítulo 1.

Conceptos esenciales para el abordaje económico y productivo del FLOSS

<i>Hernán Alejandro Morero.....</i>	<i>21</i>
1. El abordaje económico-productivo del FLOSS: los aspectos conceptuales y los avances de la literatura.....	22
1.1. Características económicas de la producción de FLOSS a nivel empresarial.....	22
1.2. La investigación económico-productiva sobre el FLOSS: un resumen de la literatura.....	33
2. El FLOSS en la industria del Software.....	45
2.1. La importancia del FLOSS en la industria global del sector de software....	45
2.2. El uso y producción de FLOSS en el sector de software de la Argentina....	47
3. Reflexiones Finales.....	52

Capítulo 2.

La producción en comunidades FLOSS: empresas, Fundaciones y *governance*

<i>Juan Gabriel Vélez y Agustín Zanotti.....</i>	<i>55</i>
1. Comunidades FLOSS.....	56
2. Empresas y la gobernanza FLOSS.....	59
3. Fundaciones FLOSS.....	62
4. Análisis de casos.....	64
4.1. GNOME.....	65
4.2. Fedora.....	67
4.3. Python.....	68
4.4. LibreOffice.....	70
5. Discusión.....	71
6. Cierre.....	73
Fuentes Documentales.....	74

Capítulo 3.

Grandes corporaciones de software y el FLOSS: cooperar en entornos abiertos como estrategia de competencia

<i>Ignacio Juncos y Carina Borrastero</i>	77
1. Antecedentes y marco conceptual: las grandes corporaciones, el FLOSS y la open cooepitation.....	80
2. Resultados empíricos.....	84
2.1. El mercado global del software en la actualidad.....	84
2.2. Las empresas dominantes del mercado global del software con participación relevante en FLOSS.....	87
2.3. El caso de una empresa líder íntegramente FLOSS: Canonical.....	102
3. Discusión.....	103
Fuentes documentales.....	108

Capítulo 4.

Las PyMEs de Software y el FLOSS

<i>Denise Gutiérrez Montecino y Florencia Manzo</i>	113
1. El uso y producción de FLOSS en Pymes de software.....	115
2. Estudios de casos de PyMEs FLOSS argentinas.....	119
A. GCoop.....	120
B. XTech.....	123
C. ECIC Systems.....	124
D. Kunan.....	126
E. Machinalis.....	128
F. TECSO.....	130
G. Entornos Educativos.....	132
3. Una síntesis de los casos de pymes estudiados.....	133

Conclusiones y Reflexiones Finales:

La presencia del FLOSS en la industria del software y las limitaciones del <i>mainstream</i> en economía para comprenderlo	139
-----------------------------------------------------------------------------------------------------------------------------------------	-----

Bibliografía	146
---------------------------	-----

Datos de autores y autoras	160
-----------------------------------------	-----

Capítulo 1. Conceptos esenciales para el abordaje económico y productivo del FLOSS

Hernán Alejandro Morero

Para adentrarse al estudio y la investigación de la actividad de producción de FLOSS es necesario partir de algunas definiciones y precisiones básicas. Éstos van desde la propia distinción de lo que es un producto de software respecto de un servicio informático, cómo en ello se distingue software de tipo privativo y de tipo open source, hasta las maneras en que el FLOSS ha abierto los espacios y alcances de los modelos de negocios de las empresas, descomponiendo las actividades de la cadena de valor de producción informática. Presentar estos conceptos y definiciones básicas es el objeto y dedicación de la sección 1.1 de este capítulo. Estos conceptos tienen, además, una utilidad transversal a todo el libro, definiendo conceptos que luego serán utilizados y referidos en todos los capítulos, cuándo no ampliados en algunos de ellos.

También es interés aquí ofrecer un punto de partida de estado del arte para insertarse en la literatura de estudios del open source. Aunque la economía no ha incorporado las particularidades del FLOSS en su conceptualización de la actividad empresarial, sí pueden identificarse aportes desde otras ramas disciplinarias que lo han estudiado. La sección 1.1 procura sistematizar una revisión de cuáles han sido los principales intereses, tópicos e investigaciones empíricas sobre el fenómeno del FLOSS desde distintos campos (la sociología, el *management*, las ciencias informáticas y también interdisciplinariamente, la economía) que confluyen en lo que podemos denominar Estudios del OSS.

Por último, en este capítulo ofrecemos evidencia cuantitativa disponible sobre la importancia y participación del FLOSS en la industria del software. En la sección 2.1, en base a información secundaria, ofrecemos indicadores sobre la importancia del OS en empresas del sector en diversos países y algunos datos sobre el crecimiento en la cantidad de proyectos FLOSS y líneas de código escritas, a

nivel global. Para visualizar y visibilizar la actividad del FLOSS en el sector de software de la Argentina contamos con una fuente de datos primaria.

Nuestro equipo de investigación como parte de distintos proyectos de investigación alrededor de la medición de la innovación en la actividad del FLOSS, está llevando adelante desde fines de 2018 y principios de 2019, una tercera onda de estos dos antecedentes, que trata de cubrir las limitaciones de los anteriores. Esta tercera onda se está relevando a nivel nacional e incorpora un diseño que amplía y completa el abordaje del FLOSS en el sector del software. Como parte de las actividades de investigación del equipo entre 2018 y 2019, ejecutamos una encuesta a nivel nacional, que considera las particularidades del FLOSS y permite su identificación y distinción en el sector de software. En base a este relevamiento, denominado “*III Encuesta sobre Innovación en el sector de Software de la Argentina 2019*”, en la sección 2.2 presentamos estadísticas nacionales que permiten ver la participación del OS en la producción de software y el uso de OS.

1. El abordaje económico-productivo del FLOSS: los aspectos conceptuales y los avances de la literatura

En esta sección presentaremos una conceptualización sobre el abordaje económico de la producción de FLOSS y su contextualización en la industria del software. Esto nos lleva a desarrollar dos puntos: a) ¿Qué es y qué tiene de particular la producción de FLOSS?; y b) ¿En qué tópicos y de qué manera se han preocupado, tanto la economía como otras disciplinas, de la producción de FLOSS?

1.1. Características económicas de la producción de FLOSS a nivel empresarial

Un punto de partida para comprender lo que implica que el software sea de código abierto y qué efectos tiene ello sobre la manera en que se produce, está en la misma definición de lo que es el software en sí. Un software es un programa informático, pero como actividad productiva es algo más que la sola escritura de código, involucra toda una serie de actividades “inmateriales” a su alrededor. Por ello, es que una de las dificultades analíticas del estudio de este sector ha estado en los difusos límites entre lo que constituye un producto y lo que hace a un servicio informático, y en buena medida no es una cuestión totalmente zanjada en la literatura.

Buscando una alternativa operativa se puede considerar al software como un producto y definirlo como la licencia de un programa informático o parte de él, que es necesaria para su uso. La licencia puede ser única o duplicable. Cuando la licencia es única, se trata de un producto a medida. Cuando esta puede ser

duplicada todo lo deseado, se trata de un producto estandarizado o enlatado. Por su parte, los servicios informáticos son las actividades ofrecidas *alrededor de estos productos*, alrededor de un software determinado: desde la provisión de diversas actividades de consultorías, la propia implementación, el soporte o la capacitación sobre el uso de un software.

Estas dos distinciones, permiten configurar una matriz de productos y servicios (UNU MERIT y Berlecon Research, 2002), como grafica la Figura 1: por un lado quedan definidos productos estandarizados, servicios estandarizados y, por otro, a la provisión de soluciones, comprendiendo tanto el desarrollo de software a medida, que involucra conjuntamente productos y actividades de servicios, como a la provisión de servicios informáticos personalizados.

Figura 1.1. El software como producto y como servicio

	Productos	Servicios
Estandarizados o empaquetados	Software empaquetado	Servicios estandarizados
A medida	Soluciones (software a medida y servicios individuales)	

Fuente: UNU MERIT y Berlecon Research (2002)

Estas definiciones iniciales nos permiten avanzar en dos aspectos que son: i) entonces ¿cuándo un software es un FLOSS? ¿qué lo distingue?; y ii) ¿cómo estos bordes difusos entre servicios y productos informáticos se reflejan en la actividad productiva y económica de las empresas? ¿en qué afecta sobre ella el FLOSS?

Con respecto al primer aspecto, tomemos el software como un producto, tal y como lo hemos definido, como un programa informático con una licencia. Las licencias establecen términos y condiciones entre el que hace un software y quienes acceden a él, y pueden ser de distinto tipo (en el **Recuadro 1**, ofrecemos una breve clasificación de los principales tipos de licencias de software). Pueden ser cerradas/propietarias o incluso abiertas, que contrariamente a restringir el uso del software y su código, obliguen a quien lo utilice, que mantenga de modo público derivaciones que pueda obtener al cambiarlo y usarlo como insumo en nuevos desarrollos. Es el caso de muchas licencias que caracterizan al FLOSS.

Recuadro 1. Las licencias de software

Un programa informático puede adoptar distintos tipos de licencias.

i) Licencias propietarias / de código cerrado. Son aquellas que se especifican para evitar el acceso al código fuente del software. No permiten que el software sea modificado, desensamblado, copiado o distribuido de formas no planteadas en la propia licencia. Restringe el número de copias que pueden ser instaladas e incluso los fines para los cuales puede ser usado. Pueden incluirse aquí todo lo que son acuerdos de licencia de usuario final o *end-user license agreement* (EULA).

ii) Software de dominio público. Es cuando se publica un programa informático cediendo todos sus derechos patrimoniales y éste está disponible para cualquier ciudadano. Esto implica que cualquiera puede obtener todo su código, modificarlo e incluso publicar sus modificaciones bajo una licencia diferente

iii) Licencias de código abierto o FLOSS. Son aquellas que establecen ciertas condiciones para la visualización del código fuente de un programa informático y se distinguen por aplicar en diversas medidas el principio del *copyleft*. Las variaciones que existen entre ellas son prácticamente inabordables, pero a grandes rasgos pueden distinguirse aquellas que son *recíprocas* de las no recíprocas. **a) Licencias FLOSS recíprocas:** Éstas imponen que derivaciones posteriores de un software deban ser licenciadas bajo la misma licencia original del programa (tales como las licencias GNU-GPL). **b) Licencias FLOSS no recíprocas:** permiten que desarrollos posteriores que usan un código determinado usen otras licencias de la original, incluso de tipo propietarias, como por ejemplo la licencia BSD. Este tipo de licencias establecen otro tipo de cláusulas tales como la necesidad de dar créditos de atribución, protección de marca o proteger derechos previos de patentes.

iv) Licencias de contenido. Son aquellas que se aplican al contenido, como por ejemplo la documentación de un programa informático, y no al código del mismo. Son ejemplos de este tipo de licencias las creative commons o las GNU Free Document Licenses (GFDL).

Hay que distinguir las licencias de lo que son estrategias de licenciamiento. Por ejemplo, se suele decir que un software tiene una licencia dual. El **licenciamiento dual** no es específicamente una licencia, sino que es cuando el desarrollador de un programa lo ofrece bajo distintas licencias/condiciones para distintos grupos de usuarios. Así, provee con una licencia FLOSS a la comunidad un software, y la misma aplicación la provee con una licencia comercial a otros usuarios. Algo similar debe decirse sobre la práctica del **shareware**. Ésta se trata en realidad de una modalidad de distribución de los programas, donde el usuario puede acceder a modo de prueba a una serie de funcionalidades del software (o por un plazo determinado) y para acceder al uso completo debe pagar una licencia. El **freeware** es un software gratuito, pero que mantiene su licencia privativa, y su código fuente no es accesible, ni se puede modificar.

Como producto, un **programa informático se define como Software Libre** si sus usuarios tienen la libertad de ejecutar, estudiar, modificar y mejorar, copiar, y distribuir el producto (las 4 libertades del software libre), para lo cual su código debe estar abierto. Esta libertad se garantiza estableciendo con el software una forma particular de licencia, conocida como *copyleft*, que establece lo contrario de los derechos de propiedad privada: es un tipo de licencia sobre un software que obliga al que lo use, o lo modifique, a mantener libre y en la esfera pública el código que pudiera producir a partir de él. Esto mantiene el software protegido de ser apropiado por particulares, y garantiza que los desarrollos que se hagan a partir de él, también lo sean.

Un **programa es Open Source** (de código abierto) cuando el código fuente está disponible con sus versiones ejecutables. Para ser considerado un software libre además debe: i) estar disponible en la esfera pública; y ii) respetar las cuatro libertades básicas mencionadas. En gran medida la diferencia entre las corrientes de *Open source* y de Software Libre es “filosófica”. Desde un punto de vista operativo, a nivel productivo en empresas y en términos de su impacto económico, los términos pueden usarse indistintamente, o bien conjuntamente, como **FLOSS**. De este modo, un programa informático es FLOSS cuando es un Software Libre o es un Software Open Source.

Estas condiciones no implican necesariamente de que el software desarrollado deba ser gratuito.¹ Podemos distinguir de este modo, según la libertad de acceso al código fuente (su apertura) y el pago necesario para que el usuario acceda al programa, unas cuatro formas de software (UNU MERIT y Berlecon Research, 2002), como detalla la Figura 2. El software propietario o privativo, donde el código fuente no está disponible con el producto, sino que es distribuido de forma binaria, y no es gratuito, sino que se distribuye de modo comercial. También el código es cerrado para el *shareware* (un software gratis, pero de funcionalidades limitadas y código cerrado) y el *freeware* (un software que es gratuito, pero que no permite acceder a su código) y si bien ambos comparten que su distribución es gratuita, en el primer caso éste carácter está limitado a un período inicial, mientras que por el segundo no se cobra derechos de licencia en absoluto (por lo menos para la versión *freeware*). Por su parte tenemos dos tipos de FLOSS según su gratuidad monetaria, donde el producto se distribuye junto con el código

1- De hecho, es la razón por la que se incluye el término Libre dentro del acrónimo FLOSS, para evitar el carácter polisémico del término *Free* en inglés.

fuelle en ambos: el FLOSS comercial (que no es gratuito e involucra un pago) y el FLOSS no comercial (cuando es gratuito).

Figura 1.2. Formas de software según gratuidad y apertura del código

		Apertura del código fuente	
		Sí	No
Precio para el usuario	Gratuito	FLOSS no comercial	Freeware Shareware
	Pago	FLOSS comercial	Software privativo comercial

Desde el punto de vista económico, el aspecto más distintivo del FLOSS es su forma de producción. A diferencia de la forma “típica” de producción de software (privativo) – dentro de una empresa donde ésta forma un equipo de personal interno (programadores, testers, project managers, analistas funcionales, etc.) – para llevar adelante un proyecto de desarrollo de un programa informático, o de una funcionalidad, etc.; aquí la producción se realiza tanto *in-house*, como públicamente en proyectos colectivos de la comunidad. Ésta comunidad, donde los proyectos de desarrollo se realizan con programadores que se ofrecen voluntariamente (o como parte de su trabajo en empresas) a programar para el proyecto de desarrollo, se denomina **Comunidad FLOSS**.² Los **proyectos FLOSS** son el modo de llevar a cabo la producción en cada comunidad y desarrollar programas informáticos de forma colaborativa y colectiva, según diversos tipos de arreglos y acuerdos institucionales.

En los proyectos FLOSS participan tanto desarrolladores individuales por su cuenta, como académicos, organismos y dependencias públicas, como trabajadores de empresas. Las maneras en que las empresas, tanto grandes corporaciones, como pymes de software, participan de esta forma de producción son muy diversas y activas. Miles de nuevos software y servicios informáticos se producen a diario de esta forma sin intermediación de derechos de propiedad privada que resulte relevante para los procesos de desarrollo e innovación empresarial.

2- O Comunidad de Software Libre, o Comunidad Open Source.

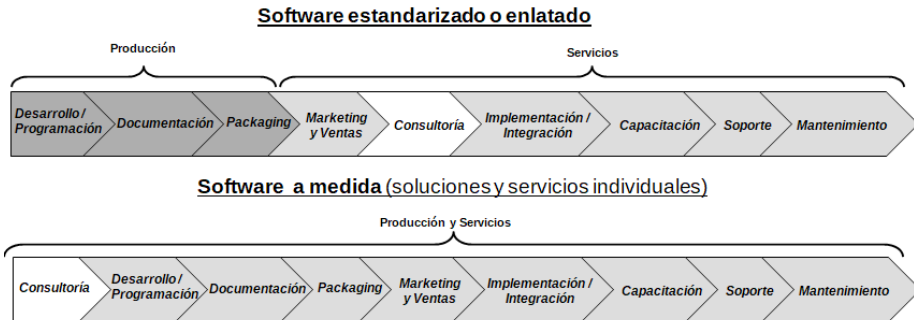
Esto nos lleva al segundo aspecto a abordar en esta sección. ¿Cómo es posible, entonces, que las empresas se involucren en actividades de desarrollo colaborativo, innoven y obtengan ingresos económicos en esta forma de producción donde los derechos de propiedad privada están ausentes? Las formas en que lo hacen son variadas, pero podemos agruparlas en dos formas de obtención de ingresos: **1) Pueden obtener ingresos de la venta y distribución de software de código abierto**, tanto si éste fue desarrollado por terceros (otra empresa, una comunidad de programadores, una dependencia pública o académica) como si fue desarrollado internamente; tanto si éste se trata de un producto enlatado o a medida. **2) Puede proveer una amplia diversidad de servicios alrededor de un software FLOS**, también tanto si éste ha sido desarrollado internamente como si ha sido creado por terceros (incluso un “competidor”).

Muchos de los servicios y actividades en las “etapas del proceso productivo” del software son similares si se realizan a partir de software privativo como si se realizan a partir de FLOSS. Pero no todas, ni en su mayoría.

Siguiendo una estilización de la cadena de valor de la producción de software en etapas productivas (ver figura 3) desarrollada por investigadores de la UNU-MERIT pueden distinguirse las siguientes actividades (que pueden o no realizarse dentro de una misma organización): el **desarrollo de software**, la **confección de documentación**, el **embalaje o *packaging***, el **marketing y ventas**, la provisión de servicios de **consultoría**, la **implementación e integración** de software; **capacitación** en el uso del software, el **soporte** y la **gestión de la aplicación o**

mantenimiento. 3 4 5 6 7 8 9 10

Figura 1.3. Cadena de Valor de la producción de software



Fuente: Elaboración propia en base a UNU MERIT y Berlecon Research (2002)

3- Esta estilización sigue una propuesta del Floss Project de la Universidad de Naciones Unidas (UNU-MERIT) plasmada en su reporte de 2002 como resultado de una profunda investigación sobre FLOSS. En particular, esta estilización de la cadena de valor recoge tanto una reseña exhaustiva de la literatura y la investigación online realizada a esa fecha, así como muchas charlas de expertos y presentaciones en conferencias y rondas de negocio, como la Wizard of OS 2, en Berlin en octubre de 2001 o la LinuxWorld Conference de noviembre 2001 en Frankfurt.

4- Se trata de la propia escritura del código técnico, incluyendo la producción de herramientas de desarrollo.

5- Es la agregación y edición de información relacionada con la aplicación y uso de los productos de software. Pueden mencionarse tres tipos de documentación relevante: 1) El software en sí, incluye el código fuente en el caso de FLOSS; 2) Documentación electrónica explicando el código dentro del software; y 3) Documentación editada sobre la funcionalidad y uso del software.

6- Que es la agregación, integración y optimización de los productos de software o sus componentes (pudiendo incluir la corrección de errores).

7- Que va desde el análisis de situación del cliente, la definición de requerimientos (funcionales y técnicos) y hasta la selección de soluciones de software acorde a ello.

8- Es la instalación, configuración y customización del producto.

9- Que incluye todas las acciones tendientes a ayudar al cliente a solucionar distintos problemas que vayan surgiendo en el uso.

10- Son todas las actividades para asegurar que el software se desempeñe en el modo deseado, como la gestión de la operación, el *upgrading*, mantenimiento y *versioning* del software

La mayor parte de las actividades de servicios (consultoría, implementación, capacitación y la gestión de la aplicación) no tienden a presentar diferencias significativas si son provistas alrededor de software propietario o de FLOSS. La actividad de desarrollo de software suele presentar características diferenciadas entre una forma de organizativa “más jerárquica” (siguiendo el principio de la “catedral”), o una más “horizontal y dispersa” (bajo el principio del “bazar”), aunque la realidad tiende a mostrar predominantemente la existencia de modelos híbridos y no es posible ser tajante en ese sentido.¹¹

El resto de las actividades de la “cadena de valor” sí presentan características muy diferenciadas respecto si presan vinculados a un FLOSS o si se realizan alrededor de software privativo. La Tabla 1 muestra y resume estas diferencias. La irrupción del FLOSS en la cadena de producción de software torna distintas las actividades de producción de documentación, el empaquetado, la distribución y marketing y las actividades de soporte; respecto a cuando estas mismas actividades se prestan vinculadas a un software privativo.

Todas estas diferencias productivas del FLOSS han potenciado la posibilidad de desintegrar estas etapas/actividades de una misma unidad productora, dando origen a la oportunidad de diversos modelos de negocios especializados en una o algunas de estas etapas, lo que la oferta de software exclusivamente privativo no habilita. El hecho de que, por un lado, una firma puede especializarse en una o algunas de estas actividades (y allí las combinaciones ya son muy variadas), pero además, de que las empresas puedan ofrecer algunas de estas actividades sólo a partir de FLOSS, otras sólo a partir de software privativo, y otras alrededor de ambos tipos de software; ha llevado que todo un conjunto de estudios y esfuerzos hayan fracasado en ofrecer una caracterización comprensiva de todos los modelos

11- Los principios del bazar y de la catedral son dos metáforas introducidas en un famoso ensayo por Eric Raymond, figura líder del Movimiento OS. La idea en el ensayo de Raymond (1999) caracteriza a la forma de desarrollo de software según un principio, que es el de la catedral, como un esquema verticalista asociado a la producción de software comercial (por lo menos hasta fines de la década de los 1990's); mientras que en un extremo opuesto se encuentra el software producido dentro del mundo Linux, con una dinámica más horizontal, equiparada al funcionamiento de un bazar. Contrariamente la construcción detallada, planificada y sistemática de los grandes sistemas de gran envergadura (como la edificación de una catedral), el desarrollo en el mundo Linux “se asemeja más a un bullicioso bazar de Babel, colmado de individuos con agendas y enfoques dispares (...), de donde puede surgir un sistema estable y coherente (...)” (Raymond, 1999).

de negocios a partir del FLOSS posibles.¹²

Una de las conclusiones del libro de Lerner y Schankerman (2013) “*The comingled code: Open source and economic development*”, es que las empresas de software mezclan actividades de desarrollo OS con software propietario en proporciones muy variables entre distintos tipos de empresas (en las que el OS llega en algunos segmentos, como el de medianas empresas, a representar el 53% de las firmas de software).¹³ También Bonaccorsi *et al.* (2006) muestran la prevalencia casi absoluta de modelos de negocio “híbridos” en las pymes de software, que mezclan software privativo y OS en su oferta productiva.

Por ello, cabe volver a señalar que los intentos de clasificar las empresas de software según la medida en que incluyen el FLOSS en su modelo de negocio han sido insatisfactorios para proveer una tipología comprensiva del sector.¹⁴ Han logrado, sí, caracterizar algunos casos emblemáticos, como el de las empresas que proveen versiones originales de sistemas operativos Linux (p.e. Canonical con Ubuntu o RedHat), o algunas pymes especializadas en brindar soporte, capacitación e implementación alrededor de un FLOSS de nicho, pero realmente las formas de establecer una oferta productiva en software son sumamente variables.

12- Ver, por ejemplo, las limitaciones señaladas en Morero et al. (2017a) y en Morero et al. (2017b).

13- Y nos referimos a una muestra que roza las 1900 empresas de unos 15 países en el mundo.

14- Entre ellos cabe listar una serie de trabajos: Hecker (1999), UNU MERIT y Berlecon Research (2002), Lerner y Schankerman (2013), Hauge *et al.* (2008) y Seppänen *et al.* (2007)

Tabla 1.1. Actividades de la cadena de valor de producción de software.
Diferencias FLOSS y Software Propietario

<i>Actividad</i>	<i>Producción / Programación</i>		
	<i>Desarrollo de Software</i>	<i>Documentación</i>	<i>Packaging</i>
<i>Provisión al proveedor de Software Propietario</i>	Los modelos pueden combinar modelos verticalistas u horizontales en ambos casos.	El software es usualmente vendido con la documentación, por lo que esta etapa suele integrar uno de los últimos pasos del proceso de desarrollo dentro de la empresa.	El software se vende empaquetado y es parte de uno de los últimos pasos del proceso de desarrollo.
	- La mayor diferencia entre ambos es la apertura del código fuente	Los FLOSS normalmente no son un producto finalizado, y la documentación no suele ser suficiente para los usuarios.	Este es una parte importante de muchos modelos de negocio basados en FLOSS que reúnen, complian y agregan el software, lo optimizan y lo venden empaquetado.
<i>Provisión al proveedor de un FLOSS</i>	Es por eso que la producción de documentación junto con las ventas es una actividad común de las empresas que proveen FLOSS, tanto haya sido desarrollado por	Diversas empresas FLOSS se dedican a la etapa de distribución, marketing y ventas, tanto de sistemas operativos originales, como de software especializado o de nicho. Esto involucra tanto la distribución como la agregación de los componentes, sus cambios y ocurre porque quien desarrolla el software FLOS (p.e. una Comunidad OS) no es necesariamente quien lo promueve, quien le hace el marketing y quien lo vende y distribuye, porque estas	

Servicios			
Consultoría	Implementación / Integración	Capacitación	Mantenimiento y Gestión de la aplicación
No presenta diferencias significativas			
<p><i>Proveedor de Software Proprietario</i></p>	<p>Generalmente es ofrecido por una empresa especializada o el propio desarrollador del software. Suele hacer una distinción entre usuarios privados (asistencia simple, ayudas en los programas y telefónica) y corporativos (una ayuda más personalizada, con soporte de administradores).</p>	<p>El soporte es ofrecido, en primer lugar, por los foros de la comunidad.</p>	<p>No presenta diferencias significativas</p>
<p><i>Proveedor alrededor de un FLOSS</i></p>	<p>Como esto no es aceptable para todos los tipos de usuarios, surgen ofertas de soporte especializado, por parte de distribuidores y empresas OSS independientes (p.e. soporte corporativo)</p>		

Es claro, sin embargo, que existen maneras altamente heterogéneas a través de las cuales las empresas obtienen ingresos de la incorporación del FLOSS en sus actividades, y que ello tiene un impacto económico y productivo importante en el sector del software. En la Sección 2 mostraremos la dimensión de este impacto a través de distintas estadísticas disponibles, tanto de forma secundaria como primaria. Pero antes de ello, nos preguntamos ¿en qué aristas ha estado concentrada, tanto la literatura económica, como otras clases de estudios, en investigar sobre la producción del FLOSS? La subsección siguiente procura ser un mapa a los diversos focos de estudios y avances que ha habido en la investigación del FLOSS en las últimas décadas.

1.2. La investigación económico-productiva sobre el FLOSS: un resumen de la literatura

El estudio del FLOSS en su faceta económica y de producción, es un campo donde han convergido disciplinas desde puramente informáticas, hasta sociales e ingenieriles. Así lo reflejan las más de 14 *Open Source Systems Conferences* que desde 2006 ha realizado a lo largo del mundo el IFIP Working Group 2.13 on Open Source Software de la *International Federation for Information Processing*.¹⁵

La literatura puramente económica en relación al estudio del FLOSS ha estado cercenada a pocos tópicos. Por lo general, se ha tratado al FLOSS como un fenómeno anómalo y excepcional, difícil de comprender para la economía. Por ello, el principal tópico de investigación han sido las **motivaciones para involucrarse en esta actividad** (Robert, 2006b). De modo abrumador se han estudiado las motivaciones de los programadores individuales a “trabajar sin un pago directo” como parte de los proyectos OS de la comunidad, y en menor medida, las motivaciones de las empresas a colaborar en la comunidad FLOSS, aportando trabajadores, su tiempo de trabajo y recursos económicos a ello. Su preocupación central ha estado en cómo es posible esta “anomalía” de que distintos tipos de participantes renuncien a la posibilidad de lucrar económicamente través del monopolio que otorgan los derechos de propiedad intelectual de su creación y ofrecen, por el contrario, el producto de su trabajo como de propiedad colectiva.

Estos estudios han avanzado en explorar cuales son las motivaciones individuales para participar, listando: razones reputacionales (Dalle y Jullien, 2003, Lakhani y von Hippel, 2003, Lerner y Tirole, 2000), el aprendizaje y

15- <http://ifipwg213.org/>

mejoramiento de capacidades de programación (Lakhani *et al.*, 2002, Lakhani y von Hippel, 2003, Von Krogh *et al.*, 2003), la búsqueda de conectar necesidades sociales con funcionalidades de un software (Hars y Ou, 2001) , y el propio esparcimiento (Lakhani y von Hippel, 2003).

Por su parte, la investigación de las motivaciones de las empresas ha avanzado en clasificarlas en tres tipos de motivaciones: sociales, tecnológicas y económicas. Algunas investigaciones han hallado una preeminencia de éstas dos últimas sobre las motivaciones sociales (Bonaccorsi y Rossi, 2006, Bonaccorsi y Rossi Lamastra, 2004), mientras otras han encontrado una complementariedad entre los distintos tipos de motivaciones (Stam y van Wendel de Joode, 2007).

Ha habido diversos enfoques para conceptualizar la existencia del FLOSS como un bien público o un bien común. Con todo son aportes teóricos motivados a por qué el FLOSS existe, y no al cómo es, y a su importancia en la estructura económica. La economía sigue sin dar(se) cuenta de la relevancia y participación del FLOSS en el sistema de producción capitalista actual.

Luego, hay toda una serie de avances en la comprensión del funcionamiento económico, productivo e innovativo del FLOSS, que no ha provenido puramente de la economía, sino de un ámbito de lo que podríamos llamar *Estudios del OSS*, compuesto por distintos campos, que van desde la sociología, el *management*, a las ciencias informáticas y, también, interdisciplinariamente, la economía). Más allá de algunas cuestiones que tienen que ver con los aspectos de las características del licenciamiento OS o demostrar la superioridad técnica de los software OS sobre los privativos, a estos estudios debemos los avances en ciencias sociales a lo que es la comprensión económica del FLOSS, que sistematizamos en los siguientes cinco apartados temáticos.^{16 17 18}

16- Muchos estudios destacan como una de las más importantes innovaciones institucionales que ha introducido el FLOSS en el mundo del software ha sido la cuestión del licenciamiento y la propiedad intelectual. Por ello, diversos artículos (Mustonen, 2003, Skidmore, 2007)) exploran, describen y clasifican las licencias OS; y cuestionan directamente el enfoque tradicional de “fallas de mercado” para la innovación (Rossi, 2006).

17- La rivalidad entre el FLOSS y el software propietario ha sido trabajada como productos “competidores”, tanto en términos de calidad del software como de dinámica de competencia tecnológica. Un ejemplo paradigmático son las comparaciones entre Linux y Windows (Dalle y Jullien, 2000, Dalle y Jullien, 2003).

18- Priorizamos en estos cinco ítems las publicaciones, dando una menor importancia a las

a) Funcionamiento económico del FLOSS en general.

Hay un conjunto de libros de gran envergadura que luego de introducirnos a la historia del FLOSS, procuran dar una perspectiva económica global del fenómeno, desde su relación con el crecimiento y el desarrollo, sus dimensiones de oferta y demanda y cuestiones de política pública.

En este grupo deben destacarse el libro de Lerner y Schankerman (2013), *“The Comingled Code: Open Source and Economic Development”*, el manual de Amant y Still (2007) *“Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives”*, el libro de Weber (2004), *“The success of open source”*, y el de Bitzer y Schröder (2006), *The Economics of Open Source Software Development. Éstos constituyen, junto con los trabajos seminales de Richard Stallman y de Eric Raymond, puntos de partida para la investigación y comprensión económica del FLOSS* (Stallman, 1983, Raymond, 1999).

b) La forma de producción de los proyectos de desarrollo colectivo de FLOSS como un proceso de innovación.

La investigación ha logrado esquematizar de modo bastante adecuado el proceso productivo de los proyectos de desarrollo de FLOSS.¹⁹ Los mismos funcionan como un sistema de innovación distribuida caracterizado por:

- La ausencia de una unidad centralizada de toma de decisiones *ex-ante* y la existencia de un diseño y corrección del código concurrente. Esto es facilitado por la modularidad de los software y su código fuente, que permite que distintos grupos de desarrollo esté trabajando simultáneamente en distintos componentes que pueden ser ensamblados para lograr funcionalidades altamente complejas (Bonaccorsi y Rossi, 2003). Esto permite que el proceso de desarrollo se lleve adelante de modo concurrente y explotar una inteligencia colectiva distribuida, con resultados innovativos superiores a los esquemas de *software factory* tradicionales (Kogut y Metiu, 2001).

presentaciones en reuniones científicas. Sin embargo, a continuación, la complementamos con los aportes de dos importantes eventos académicos, que son las *OSS Conferences* y las Jornadas de Software Libre de la JAIIO.

19- Nos concentramos a nivel de proyectos colectivos FLOSS, pues también hay una serie de estudios emparentados con la ingeniería de software que están más emparentados en como son los procesos de desarrollo. Un listado de estos estudios puede consultarse en Aksulu y Wade (2010), aunque de allí podemos destacar a Baldwin y Clark (2006), Koch y Gonzalez-Barahona (2005) y MacCormack *et al.* (2006).

- La integración de los propios usuarios en el proceso productivo, creativo, de desarrollo, corrección y mejoras al software (Hippel y Krogh, 2003).

- La producción colaborativa basada en bienes comunes (*commons-based peer production*). Ésta describe un modo de producción económica donde un número importante de personas trabaja de modo cooperativo y voluntario, usualmente mediado por tecnologías digitales y la Internet (Benkler, 2017). Funcionan ante una aparente ausencia de jerarquías y se establecen comunidades de producción entre pares, que desempeñan funciones tales como la coordinación, la división del trabajo, la búsqueda y aceptación de colaboradores, la capacitación, la creación y cumplimiento de normas, resolución de conflictos y el establecimiento de metas y objetivos. En el FLOSS, a través de esta forma de organización los programadores que van a realizar diversas tareas en la producción se autoseleccionan y distribuyen entre las funciones (Rossi, 2006).

Todo esto conforma un “modelo comunal de innovación” (Lee y Cole, 2003) o un modelo de innovación “privado-colectivo” (Hippel y Krogh, 2003). La efectividad del modelo de producción de los proyectos FLOSS está en el hecho de que el resultado del proceso innovativo no es percibido completamente ni como un “bien público puro”, ni como un “recurso de uso común” (O’Mahony, 2003), sino que posee significativos beneficios que pueden ser apropiados privadamente aunque las innovaciones “se revelen” libre y gratuitamente, sin la necesidad concreta de establecer derechos de propiedad privada. De este modo, preconcepciones de la teoría económica *mainstream*, tales como el comportamiento *free rider*, no parecen tener relevancia empírica. Contrariamente a la idea tradicional que ata incentivos para innovar y propiedad privada, la liberación de una innovación puede no involucrar una pérdida de beneficios para el innovador, sino una ganancia por la difusión de la innovación y por la obtención de efectos de red. El proceso de innovación de las empresas que se involucran se ve impulsado y acelerado por la participación en lo que aparece como un pool de I+D colectiva (Dahlander y Magnusson, 2008), de modo que se reduce significativamente las inversiones requeridas para desarrollar un nuevo software.

c) El funcionamiento, formas de organización y gobernanza de las comunidades FLOSS

El carácter de un sistema de innovación distribuida de los proyectos FLOSS ha llevado a investigar la manera en que las comunidades donde se llevan a cabo

y se produce el software se organizan y funcionan. Este se trata de un tema central y crítico, pues abre un espacio de competencia y cooperación tecnológico-económica, que traspasa al mercado.

Por un lado, nos encontramos con una serie de conceptualizaciones y definiciones alrededor de las interrelaciones entre las Comunidades, sus proyectos de desarrollo y sub comunidades que los componen. Ello en ocasiones es abordado como el estudio de *Ecosistemas OS* (Poo-Caamaño *et al.*, 2016), donde proyectos y comunidades viven simbióticamente y, en ese sentido, sus evoluciones, estructuras y maneras de funcionar dependen mutuamente entre sí; o como *Constelaciones de prácticas* que están enraizadas (*embedded*) en dinámicas existentes de mercado y de flujos monetarios (Berdou, 2010). Así, diversos artículos estilizan las formas en que se inician y luego se desarrollan los proyectos FLOSS (Schaarschmidt *et al.*, 2015), y los clasifican y distinguen entre sí acorde a ello (Capra *et al.*, 2011, Capiluppi *et al.*, 2012). Aparecen así proyectos FLOSS tradicionales (donde no se involucran las empresas), proyectos FLOSS con involucramiento empresarial y proyectos FLOSS liderados por empresas (Capiluppi *et al.*, 2012). Los primeros dos son proyectos comunitarios y los últimos dos cuentan con esponsorio corporativo.

Por otro, están bien asentadas en la literatura las características que posee la estructura organizativa de los proyectos FLOSS. Éstos suelen estructurarse según lo que se denomina como modelo cebolla (*onion model*) (Nakakoji *et al.*, 2002), idealizado en dos grupos de desarrolladores: al centro y de mayor importancia el núcleo o core donde están las principales autoridades, y en sus alrededores una periferia de colaboradores y programadores (Crowston y Howison, 2005, Crowston y Howison, 2006, David y Rullani, 2008, Crowston y Shamshurin, 2017). Hay un cuerpo importante que ha logrado caracterizar cómo es y cómo se forma esta división del trabajo, el rol de los méritos y la dinámica intra-organizacional de la periferia (Rullani y Haefliger, 2013).

La forma de organización de las comunidades y sus proyectos se comprende plenamente a través de investigar su gobernanza. La gobernanza OS se define como los medios de lograr (parcial o totalmente) la dirección, control y coordinación de organizaciones o individuos autónomos para que contribuyan conjuntamente hacia un proyecto de desarrollo FLOSS (Markus, 2007). Esta gobernanza puede ser “espontánea”, recurrir a una serie de instrumentos internos más formales de gobiernos, o depender de terceras partes “externas” a la Comunidad (De Laat, 2007). La mayor parte de la literatura de gobernanza proviene de dos campos

disciplinarias de investigación: las ciencias políticas y el *management* corporativo (Schaarschmidt *et al.*, 2015). Entre los temas que se han investigado en este cuerpo pueden mencionarse (Scacchi y Jensen, 2008, Lee *et al.*, 2017): cuestiones relacionadas a cómo es la autoridad de los procesos de decisión, la asignación de recursos, el liderazgo, el control social y los mecanismos de coordinación, entre otras (Di Tullio y Staples, 2013, O'Mahony y Ferraro, 2007, Berdou, 2010, Schaarschmidt *et al.*, 2015, Shah, 2006, West y O'mahony, 2008).

d) Rol y participación de las empresas

Esta es una de las líneas de la literatura OS más relevante para el estudio que engloba este libro, por el simple hecho de que la empresa constituye el actor principal de las economías capitalistas modernas.

El primer interés dentro de este tema es, bien, ¿de qué manera y porqué se involucran las empresas en las Comunidades FLOSS? Ha surgido en respuesta a ello una diversidad de estudios describiendo o tipificando las formas de involucramiento de las empresas (Jullien y Zimmermann, 2011). Las formas van desde contar con trabajadores de la empresa pagos para colaborar en proyectos FLOSS de la comunidad (Dahlander y Wallin, 2006, Henkel, 2009, Berdou, 2010, Colombo *et al.*, 2013), la propia iniciación de proyectos FLOSS esponsorados por la empresa (Capiluppi *et al.*, 2012), la liberación de código anteriormente cerrado (Henkel, 2004), la “donación” de herramientas de desarrollo desarrolladas *inhouse*, el financiamiento y apoyo a Fundaciones que soportan al FLOSS, hasta la provisión de otro tipo de infraestructuras y recursos (hardware, ancho de banda, hosting, etc.).

Los tipos de involucramiento han sido clasificados de distintas maneras. Una clasificación que califica el grado de involucramiento según el nivel de beneficio mutuo que tienen las empresas en relación a la comunidad las tipifica en tres clases: relaciones de tipo simbiótica, de tipo comensalista y de tipo parasitaria (Dahlander y Magnusson, 2005). Otras clasificaciones distinguen a las formas de participación de las empresas en proyectos OS comunitarios (que son liderados por miembros de la comunidad y no han sido iniciados por empresas) en tres modelos, que pueden combinarse para una misma firma en distintos espacios FLOSS: un modelo de aporte de código (*coding*), un modelo de soporte y apoyo a la comunidad (*support*) y un modelo de gestión (*management*) de proyectos (Capra *et al.*, 2011). También la participación de las empresas puede distinguirse si éstas se acoplan a un proyecto iniciado por la comunidad donde hay involucramiento

de firmas (*industry-involved OSS projects*) o si directamente inician sus proyectos y los lideran (*industry-led OSS projects*) (Capiluppi *et al.*, 2012).

Un segundo interés ha estado en (de)mostrar que la participación de las empresas en el FLOSS se trata de una actividad “rentable”. Aquí es muy prolífica toda una literatura que muestra y explicita toda una serie de beneficios que tienen para las empresas actividades que van desde la liberación de software, hasta la colaboración en proyectos comunitarios y el apoyo al FLOSS. Por otro lado, hay un conjunto de investigaciones que se han empeñado en mostrar diversos “modelos de negocio” vinculados al FLOSS (Bonaccorsi *et al.*, 2006, Seppänen *et al.*, 2007, Hecker, 1999, UNU MERIT y Berlecon Research, 2002) que tienen las limitaciones que señalamos en la sección anterior: su incapacidad para dar cuenta de la heterogeneidad de combinación de actividades que habilita el FLOSS en confluencia con la hibridación predominante en las carteras de oferta del software.²⁰

Es esto último uno de los resultados más robustos e interesantes de estos esfuerzos: tanto el predominio de diversos grados de combinaciones entre software OS y privativo en la especialización de las empresas del sector (West, 2003, Bonaccorsi *et al.*, 2006, Lerner y Schankerman, 2013, Munga *et al.*, 2009), como de una amplia diversificación productiva impulsada por el OS (Colombo *et al.*, 2014), aparecen como rasgos característicos de la actividad del software actualmente a nivel global.

Por último, un tercer interés ha estado en la dinámica competitiva que ha abierto el FLOSS entre los líderes tecnológicos de la industria del software. Es lo que se ha dado en llamar *open coo-petition* (Teixeira *et al.*, 2016, Samuelson, 2006), representando el hecho de que grandes actores del sector aparecen colaborando en el ámbito del OS, por ejemplo en un gran proyecto, y rivalizando al mismo tiempo, compitiendo por la dirección tecnológica del mismo (o participando en proyectos competidores en una misma área, como pueden ser dos proyectos de software de infraestructura de computación en la nube tales como OpenStack y CloudStack). De esta manera el FLOSS ha abierto un espacio de cooperación y competencia paralelo al sistema de precios, en tanto que es factible argumentar incluso que los proyectos FLOSS maduros están directamente enraizados o

20- A ello habría que sumarse de que la idea de un “modelo de negocio” falla en capturar los efectos de derrame (*spill-over*) recibidos por una empresa al participar de un desarrollo colaborativo OS, que involucran todos los procesos de aprendizaje propios de una comunidad de práctica y los que ocurren a nivel de la organización (Berdou, 2010).

embebidos en estructuras de los mercados (Berdou, 2010). De esta forma, al dejar de lado de la disciplina económica el estudio del funcionamiento de las Comunidades de producción colaborativa como el FLOSS, el *mainstream* de la economía ha resignado el estudio de un importante componente de los mercados. Especialmente relevante en sectores que son los drivers del cambio tecnológico en las economías capitalistas actuales.

e) Rol y participación de los gobiernos: la necesidad de promoción pública del FLOSS.

Gradualmente el FLOSS ha entrado en la agenda de políticas públicas de gran cantidad de países. Con ello, ha surgido la necesidad de justificar la intervención pública, máxime porque el *mainstream* económico tiende a generar racionalizaciones de la aversión típicamente reaccionaria contra la intervención estatal y precisa argumentos para justificarla cuando se vuelve un aspecto necesario para las economías de mercado.

Así podemos mencionar tres tipos de argumentos dentro de la economía al respecto. En primer lugar, una línea tiende a justificar la acción pública por razones de eficiencia ante la existencia de “fallas de mercado” ocasionadas por las características del software como bien mercantil (Schmidt y Schnitzer, 2002). Los mercados de software poseen características tales como la existencia de efectos de red, grandes economías de escala y un rol ubicuo de la innovación, que requieren correcciones estatales para funcionar “eficientemente”.

Un segundo tipo de argumento relacionado versa en la superioridad técnica (calidad, seguridad, etc.) del FLOSS sobre el software privativo. Ello deriva en que fomentar su provisión se torne socialmente deseable, mientras los mercados por sí solos tenderían a subproducirlos, tal y como la economía ortodoxa concibe un bien público (Varian y Shapiro, 2003).

Finalmente, una línea argumentativa diferente enfatiza la importancia de promover el FLOSS para impulsar el desarrollo. Así, muchísimas investigaciones documentan y fundamentan las ventajas y beneficios de impulsar el FLOSS para economías periféricas. Las razones son variadas: elimina muchas de las barreras a la entrada a esta actividad debido a sus menores costos de ingreso (van Reijswoud y Mulo, 2007) y porque facilita los procesos de innovación dado el carácter “abierto” de los programas, soluciona muchos de los problemas legales de propiedad intelectual vinculados a la “piratería” (Dudley-Sponaugle *et al.*, 2007), reduce la dependencia tecnológica de la administración pública y los gobiernos a

un proveedor de software particular, permite un gran ahorro de divisas por el no pago de licencias extranjeras al uso de software privativo y secundariamente por su potencial para impulsar procesos de aprendizaje y sustitución de importaciones (Moncaut y Robert, 2016). En sí, porque el FLOSS ofrece elementos de autonomía e independencia tecnológica, que habilitan a la construcción de competencias domésticas.

Dentro de los aportes que no forman parte de publicaciones, sino que son presentaciones y comunicaciones en reuniones científicas especializadas, caben destacar dos espacios académicos y reseñar algunas de sus contribuciones, muchas de ellas complementarias a lo ya presentado en esta sección.

El primero de estos espacios académicos con las ponencias de las *Open Source Systems Conferences* de la IFIP.²¹ En este ambiente hay una predilección hacia la explotación cuantitativa de grandes bases de datos de proyectos FLOSS (como el uso de repositorios del código, registros de las contribuciones, del registro de versiones, listas de correo, etc.) y la aplicación de técnicas de big data y data mining para el análisis, además de estudios de casos (en su mayoría, más bien descriptivos y escasamente analíticos). A través de este tipo de metodologías, resulta relevante el trabajo que permite dimensionar la importancia del FLOSS a través de la medición de su crecimiento a partir del cómputo de la evolución de las líneas de código OS escritas y la medida en que aumenta la base global de proyectos FLOSS activos (Deshpande y Riehle, 2008, Wiggins y Crowston, 2010, Bordeleau *et al.*, 2019).²²

Respecto de la faceta económico-productiva del FLOSS, en estas conferencias pueden agruparse los artículos en 3 categorías, que han generado una serie de aportes que complementan la literatura previamente reseñada, especialmente de forma empírica en los siguientes tópicos:

- **El funcionamiento de las Comunidades FLOSS.** Aquí pueden hallarse:

- i) descripciones de la evolución de las estructuras de las comunidades y la intensidad de sus patrones de comunicación e información (Weiss *et al.*, 2006, Capiluppi y Michlmayr, 2007, Lindman, 2007, Kamei *et al.*, 2008, Crowston y Shamshurin, 2017);
- ii) Recuentos de las formas y estrategias de gobernanza que llevan adelante sus líderes de las comunidades como redes de

21- International Federation for Information Processing.

22- La sección siguiente se nutre en buena medida de esta clase de aportes.

interacción socio-técnicas (Jensen y Scacchi, 2010, Mäenpää *et al.*, 2017); y iii) Explicaciones de la importancia del involucramiento de distintas clases de organizaciones para el éxito y funcionamiento de las comunidades, tales como las empresas (Capiluppi *et al.*, 2012), las Universidades, el Gobierno y las Fundaciones (Forrest *et al.*, 2012, Lindman y Hammouda, 2017).

- **El involucramiento de empresas en la producción de FLOSS y en las comunidades OS.** En esta temática predomina la narrativa e ilustración de estudios de casos, la mayoría utilizando fuentes cuantitativas, pero complementadas con recursos cualitativos. Tres grandes aportes deben señalarse: i) Los beneficios concretos que cuentan las empresas por liberar sus desarrollos, (Lindman *et al.*, 2009, Kim *et al.*, 2012, Capra *et al.*, 2009) de usar componentes OS incluso para la producción de software privativo, de participar en las comunidades grandes y también las maneras (tales como el contar con programadores contratados para colaborar, dar infraestructura y personal para los proyectos, etc.) (Berdou, 2006, Martínez-Romo *et al.*, 2008, Lindman *et al.*, 2009, Capra *et al.*, 2009, Melian y Mähring, 2008, Teixeira, 2012); ii) Descripciones de experiencias concretas de primera mano sobre la participación e influencia de grandes corporaciones en muchos desarrollos OS (Aaltonen y Jokinen, 2007, Robles *et al.*, 2007, Jaaksi, 2007, Durand *et al.*, 2012, Lindman *et al.*, 2008) y la manera en que cooperan y compiten grandes actores de la industria del software al interior de proyectos FLOSS de gran dimensión, tales como OpenStack (Teixeira, 2017).²³ iii) Algunas experiencias de pymes productoras de FLOSS: aunque son menores, pueden contarse estudios muy relevantes sobre casos de pymes noruegas que han sido capaces de crear comunidades FLOSS alrededor de sus desarrollos y mantenerse en actividad hasta la actualidad (Hauge y Ziemer, 2009), y la descripción de varios modelos de negocios FLOSS de mipymes africanas (Sowe y McNaughton, 2012).

- **La descripción del involucramiento de otros grandes usuarios y desarrolladores de FLOSS: Administraciones Públicas y Universidades.** En este campo las ponencias son más bien descripciones de experiencias, tanto de organizaciones públicas que utilizan FLOSS y como han realizado las migraciones (Ven y Verelst, 2006, Ven *et al.*, 2006, Brink *et al.*, 2006,

23- Los casos incluyen proyectos concretos, como el propio núcleo de Linux o Debian o experiencias concretas, tales como la de Nokia, Samsung, Facebook, Google o Phillips Medical Systems.

Kuechler *et al.*, 2012, Robles *et al.*, 2016, Conklin, 2006), su uso en cuestiones infraestructurales, tales como la administración del tráfico aéreo o el establecimiento de redes de información de salud pública (Hardy y Bourgoi, 2006, Staring y Titlestad, 2006), y su implementación en el sector educativo en distintos países (Brink *et al.*, 2006, Sowe *et al.*, 2007, Petrinja y Succi, 2012). Claramente estos estudios tienen un sesgo temático hacia las cuestiones de uso e implementación.

Fuera de estas líneas temáticas, caben mencionarse algunos antecedentes sobre las disparidades de género en el sector del software y en el mundo del FLOSS (Kuechler *et al.*, 2012, Robles *et al.*, 2016). Luego, en estas conferencias hay otra serie de ponencias que se alejan de la temática económico-productiva o son más bien de índole técnico o de corte metodológico. Allí entran ponencias que evalúan y presentan diversas características de los software FLOS, de los proyectos OS y su dimensión (tales como la calidad de los programas, formas de testeo de calidad, aplicaciones concretas con distintos fines, los ciclos de vida de los proyectos y sus patrones de forking, uso de métricas para la selección de código, etc.), el análisis de prácticas y modelos de desarrollo vinculadas (como la relación con las metodologías ágiles o el uso de *sprints* como forma de desarrollo, herramientas de gestión de proyectos, otras cuestiones más técnicas como problemas de autenticación recomendaciones para la adopción de librerías; etc.); aspectos legales y de licenciamiento, la enseñanza del FLOSS. Por último hay un cuerpo de ponencias alrededor de cómo investigar al FLOSS haciendo uso de técnicas de big data y de procesamiento de lenguaje natural a través del acceso a repositorios de repositorios, describiendo técnicas para implementar y limpiar bases, qué indicadores usar sobre un proyecto (Sowe *et al.*, 2007, Conklin, 2006, Petrinja y Succi, 2012, Noll *et al.*, 2012, Ribeiro *et al.*, 2018).

El segundo espacio académico relevante de ponencias y comunicaciones especializadas ha sido un evento análogo a las *OSS Conferences*, en Argentina o las Jornadas de Software Libre (JSL) que formaron parte de las JAIIOs (Jornadas Argentinas de Informática), organizadas por la Sociedad Argentina de Informática (SADIO), entre los años 2005 y 2013. Los aportes que pueden destacarse de estas Jornadas se centran en ponencias sobre presentación de experiencias y proyectos, por lo general con escaso trabajo analítico y carácter más bien empírico.

Se destacan las presentaciones de aplicaciones para la Administración Pública, tales como la experiencia de la Municipalidad de Rosario (Giardini y Aguzzi,

2011, Boidi y Giardini, 2006), diversos casos de aplicaciones en áreas tributarias tanto a nivel nacional en AFIP (Reingart, 2012) como a nivel provincial para el caso de la DGR de Misiones (Gonella y Schelske, 2010), así como la evaluación de la implementación de OS en dependencias como el Organismo Regulador del Sistema Nacional de Aeropuertos (ORSNA) (D'Agostino, 2010) o la Dirección Nacional de Vialidad (Sirvente, 2006).

Un punto en el que ha habido avances que han presentados en las JSL son una serie de intentos incipientes en la medición del uso de FLOSS en empresas (Rubбини, 2010; Rubбини, 2009) y en la caracterización por medio de encuestas de los modelos de negocio (Castello *et al.*, 2009). Este último estudio encontró en una muestra chica de empresas argentinas FLOSS una escasa aplicabilidad de los modelos de negocios estilizados por Hecker (1999) y Raymond (1999), lo que verifica los resultados de estudios en el tema que hemos mencionado.

En síntesis, desde diversos campos disciplinares se cuenta con un buen conocimiento de cómo funcionan internamente las Comunidades FLOSS, cómo se clasifican y distinguen sus proyectos, qué motiva a sus intervinientes a participar, cómo se relaciona todo este campo con el mundo corporativo, cuáles son sus lógicas y de manera ilustrativa y anecdótica, cuál es su dimensión económica y productiva.

Maria Alessandra Rossi (2006) plantea que hay buenas razones para sostener que la complejidad de un abordaje económico del OS no está tanto en el hecho de porqué los actores contribuyen libremente a la producción de un bien que en enorme medida estará disponible “para todos”, sino en la complejidad de la estructura institucional del FLOSS y su habilidad para organizacionalmente evolucionar a lo largo del tiempo. Claramente el *mainstream* en economía no cuenta con instrumentales teórico-analíticos adecuados para abordar estas dimensiones. La ortodoxia económica sigue estancada en la “excepcionalidad” del carácter del software como un bien y mantiene una visión del FLOSS como un fenómeno anómalo, como una “falla” del mercado. Pese a ello, la participación de las empresas en el OS parece ser un aspecto bastante común del funcionamiento del mercado de software.

Aunque, en ese sentido, la teoría económica tradicional no incorpore su comportamiento como un aspecto característico de los procesos de competencia y cooperación capitalista, el FLOSS sigue avanzando a lo largo de los años en su importancia y participación en la industria del software. La sección siguiente

procura dar una idea cuantitativa de esta dimensión. Ello lo hacemos en dos pasos: primeramente en base a diversas fuentes secundarias tratamos de ofrecer diversas medidas cuantitativas de la importancia del FLOSS en la industria global del software (sección 2.1), mientras a través de un relevamiento realizado por nuestro equipo de investigación en empresas del sector de software de la Argentina, procuramos dar algunos indicios de su participación en el sector productivo de nuestro país (sección 2.2).

2. El FLOSS en la industria del Software

2.1. La importancia del FLOSS en la industria global del sector de software

Si bien es difícil la cuantificación de la dimensión del FLOSS en la producción mundial de software y servicios informáticos debido a la escasa información estadística que distinga su participación, pueden mencionarse algunos datos.

Estimaciones para inicios de la década de los 2000's sostenían que el 56% de las 2500 empresas estadounidenses más grandes utilizaban algún tipo de FLOSS, especialmente en sus servidores web (Bonaccorsi y Rossi, 2003). En base a una encuesta en 15 países (Lerner y Schankerman, 2013) hallaban que cerca del 40% de las firmas de software contribuían en proyectos FLOSS. Un relevamiento realizado en España a 1932 empresas de software (ESFA-SI 2011), revelaba que el 39% de ellas habían comercializado productos bajo licencias FLOSS en 2010 y que el 18% de sus ingresos podían atribuirse a este tipo de actividad (CENATIC, 2011).^{24 25}

En base a una encuesta realizada en 2007 a unas 570 empresas de software de Noruega, se encontró que cerca del 47% de ellas integraba componentes OS en sus productos desarrollados, que el 16% tenían empleados que colaboraban en proyectos OSS de la comunidad y un 5% contaban incluso con proyectos OSS propios (Hauge *et al.*, 2008). También en los países nórdicos, Nikula y Jantunen (2005) muestran que el 44% de las empresas de software finlandesas incorporaban OSS en su oferta de negocios. Acorde a un estudio de Gartner del

24- La encuesta se realizó en Brasil, Chile, China, Francia, Grecia, India, Israel, Kenia, México, Polonia, Rusia, Sudáfrica, Singapur, Tailandia y Turquía; y la muestra completa comprende 1894 respuestas positivas.

25- Éste fue realizado por la disuelta fundación Centro de Excelencia de Software de Fuentes Abiertas, a través de su Observatorio Nacional del Software de Fuentes Abiertas (ONSFA). A su trabajo se deben dos de los instrumentos de encuestas más destacados para identificar a la actividad empresarial FLOSS (CENATIC, 2011, CENATIC, 2009).

año 2008, alrededor del 85% de las empresas de software utilizan algún tipo de FLOSS en sus procesos de desarrollo (Zuckerfeld, 2012).

Una línea de estudios argumenta que tratar de estimar el volumen de ingresos provenientes de productos FLOSS y su participación en la industria, así como análogamente la proporción de empresas OS en el sector de software, subestima el crecimiento del FLOSS. ¿Por qué? Básicamente, porque el software OS actualmente forma parte de muchos (sino la mayoría) de productos de software propietarios de código cerrado, como insumo en sus procesos de desarrollo. Por tanto, medir los ingresos que generan los software puramente OS subestima la dimensión productiva y económica que éste tiene en la industria del software. Esta gama de estudios ha preferido medir el crecimiento total del OS a través del crecimiento en la cantidad de líneas de código y de proyectos.

A nivel de proyectos, desde la literatura de Estudios OS la evidencia empírica sobre la expansión del OS aparece como tan abrumadora, que las investigaciones se han abocado a discutir si el ritmo en que el OS crece es exponencial, cuadrático o superlineal.²⁶ Deshpande y Riehle (2008) han estudiado el carácter del crecimiento en cantidad de líneas agregadas y en cantidad de proyectos agregados mensualmente sobre una muestra de más de 5.100 proyectos OS activos y populares para un período de 10 años (1995 y 2006). Este estudio muestra que el crecimiento de la cantidad de líneas y de proyectos que se *agregan* por mes, crecen más bien de modo exponencial, de forma que la cantidad total de ambos se duplicaría en aproximadamente unos 14 meses.²⁷

Según Open Hub, a mayo de 2019, de la información disponible existían alrededor de 52.800 proyectos OS activos en el mundo con diverso nivel de intensidad en su dinamismo.²⁸ La participación de las empresas en estos proyectos

26- Por ejemplo, Succi *et al.* (2001) encuentra una tasa de crecimiento lineal para dos proyectos OS (Apache y GNU compiler compiler) y Robles *et al.* (2005) verifican que los proyectos NetBSD, FreeBSD, OpenBSD y otros 18 siguen un patrón de crecimiento lineal. Ambos estudios, junto con Godfrey y Tu (2001) confirman que el kernel de Linux crece superlinealmente. Por último, Koch (2005) sobre la base de 4047 proyectos muestra que un modelo de crecimiento cuadrático se adecua mejor a su crecimiento que uno lineal.

27- Se toman proyectos sobre cierto umbral de in-links a las webs de los proyectos. Se trata de una muestra que representa cerca de un 28% del total de proyectos OS activos a nivel mundial. A 2007, de los 150.000 proyectos registrados en sourceforge se estimaba que unos 18.000 proyectos estaban activos.

28- De un total de cerca de 218.440 proyectos OS para los que se cuenta información sobre sus niveles de actividad. Además, existen otros 246.135 proyectos de los que no se registran datos, por

de desarrollo colectivo es relevante. Capra *et al.* (2009) han encontrado sobre una muestra de 1300 proyectos OS alojados en sourceforge.net, en casi un 31% de ellos una o más empresas se han involucrado en el desarrollo de los mismos, tanto a través del apoyo a los mismos, como escribiendo código y, aunque en una menor medida, coordinando y gestionando los proyectos.²⁹

2.2. *El uso y producción de FLOSS en el sector de software de la Argentina*

En Argentina el gran crecimiento del FLOSS no ha pasado desapercibido. Ya a mediados de los 2000s, Rubbini (2009) señalaba la existencia en el país de una centena de oferentes de servicios basados en FLOSS y más de 5.000 desarrolladores especializados en esta área distribuidos en todo el país. Actualmente se evidencia una tendencia reciente es la formación de emprendimientos con modelos de negocios basados en FLOSS. Las experiencias en este sentido han dado lugar a las asociaciones como la Cámara Argentina de Empresas de Software Libre (CADESOL) y la Federación Argentina de Cooperativas de Trabajo de Tecnología, Innovación y Conocimiento (FACTTIC). Pocos trabajos se han concentrado en el papel del FLOSS en el espacio productivo local (Robert, 2006a, Robert, 2006b) y sus potencialidades en términos de un desarrollo integrado nacional (Heinz, 2006). Además del ámbito productivo, en la actualidad observamos experiencias que dan cuenta de una difusión del software libre en la administración pública, programas educativos, universidades y organizaciones del tercer sector (Zanotti, 2015).

Contamos con dos antecedentes de relevamientos tecnológicos como parte de la participación en proyectos de investigación con otros equipos del resto del país, que tienen ciertas limitaciones para nuestro objetivo, pero que sirven como unos indicios pioneros. El primero de ellos brindó información sobre el uso o el desarrollo de OS para el año 2011 en empresas de software.³⁰ Así, este

lo que no es posible saber si están activos o no, totalizando la base de Open Hub a mayo de 2019 unos 494.575 proyectos OS creados globalmente.

29- Y es preciso destacar de que Capra *et al.* (2009) consideran en su trabajo solamente proyectos OS “comunitarios”. Esto es, no analizan los proyectos OS comerciales, que son liderados y lanzados directamente por una empresa comercial.

30- Esta encuesta se realizó en el marco del Proyecto de Investigación dirigido por Gabriel Yoguel (UNGS) “Capacidades de absorción y conectividad en sistemas productivos y de innovación locales” de la Fundación Carolina (id. 386317), durante 2011 en CABA, Conurbano Bonaerense, Córdoba y Santa Fe y cubre el período 2008-2010.

relevamiento señala en base a una muestra de 257 empresas del sector, que más del 60% de las empresas de software utiliza o aporta al desarrollo en código abierto. De éstas, casi el 97% lo utiliza en sus procesos productivos y alrededor de dos tercios desarrollan módulos o programas completos FLOSS (Morero y Borrastero, 2015). Sin embargo, este diseño de encuesta no permite distinguir el uso, de la producción de FLOSS.

El segundo de ellos, discrimina la información sobre uso y sobre producción de programas OS. Fue una encuesta tecnológica realizada entre 2016 y 2017 alrededor de proyectos de investigación coordinados por investigadores de la UNGS, y realizada conjuntamente con investigadores de la UNC, UNMP, UNSAM, UNS, UNICEN y UTN-FRCU. Esta encuesta fue realizada a 183 empresas de software de la Argentina.³¹ Con respecto al uso de OS, los resultados son bastante contundentes: 9 de cada 10 empresas de software de la Argentina utilizaban alguna herramienta o insumo OS. Con respecto a la producción, la encuesta indagó sobre la realización de actividades de desarrollo de FLOSS como parte de su oferta. Una proporción menor con respecto al uso desarrollaba software de ese tipo: el 12% de las firmas entrevistadas desarrollaban FLOSS.

Se trató de un primer indicador de referencia, pero hay razones para considerar que este porcentaje subestima la participación del FLOSS en la actividad productiva e ingresos de las empresas de software. La razón de ello es la tremenda importancia de las actividades de servicios dentro de los modelos de negocios alrededor de FLOSS. El montaje de una oferta de servicios alrededor de un software open source, muchas veces genérico, pero libremente disponible, así como la provisión de desarrollos a medida a partir de este tipo de software o de código gratuitamente accesible en la comunidad; son prácticas bastante extendidas en modelos de negocios de pymes de software. Los relevamientos anteriores no han incluido mediciones de oferta de servicios informáticos a partir de FLOSS, solo actividades de desarrollo de productos.

Nuestro equipo de investigación como parte de distintos proyectos de investigación alrededor de la medición de la innovación en la actividad del FLOSS, está llevando adelante desde fines de 2018 y principios de 2019, una tercera onda de estos dos antecedentes, que trata de cubrir las limitaciones de

31- La encuesta se realizó de manera online a través de la plataforma LimeSurvey y fue realizada a empresas de software de CABA, Provincia de Buenos Aires (Bahía Blanca, Mar del Plata y GBA), Provincia de Córdoba, Provincia de Santa Fe y Provincia de Entre Ríos

los anteriores. Esta tercera onda se está relevando a nivel nacional e incorpora un diseño que amplía y completa el abordaje del FLOSS en el sector del software.

Este relevamiento, denominado “*III Encuesta sobre Innovación en el sector de Software de la Argentina 2019*”, tiene un diseño en cuatro módulos. Incorpora a los estándares típicos de medición de la innovación de las encuestas tecnológicas que siguen al Manual de Oslo (lo que habilita a comparabilidad, entre ondas y respecto a otras encuestas), precisiones al abordaje de la especialización productiva con distinciones según la producción y/o comercialización de FLOSS, provisión de servicios a partir de FLOSS, así como de la intensidad de las actividades de liberación de desarrollos de software, la importancia del FLOSS en su actividad comercial, y la utilización de FLOSS para sus procesos de desarrollo.³²

Al momento de redacción de este capítulo, el relevamiento cuenta con 105 respuestas válidas. Esto involucra empresas del sector de software de la Ciudad Autónoma de Buenos Aires (46.7% de la muestra), las Provincias de Buenos Aires (17.1%), de Córdoba (18.1%) y de Santa Fe (18.1 %).³³

En este nuevo relevamiento, al considerar no solamente el desarrollo de programas, sino también la prestación de servicios, los resultados muestran una importante participación del FLOSS en el perfil productivo del sector de software argentino. La evidencia en materia de oferta señala que más del 42% de las empresas del sector de software de la Argentina realizan desarrollo de productos FLOSS o prestan servicios a partir de un FLOSS, tanto si éste ha sido producido internamente como si se trata de un software desarrollado por terceros (p.e.: desarrollado por una comunidad). Esto configura un panorama donde 2 de cada 5 empresas de software del país realizan productos o servicios FLOSS. De éstas, en promedio, alrededor del 55% de sus ventas provienen de estas actividades, en rangos que van desde empresas que cuentan con una participación marginal de sus ingresos desde el OS (sólo 7 casos con ventas OS hasta el 20% de su facturación) hasta firmas que se especializan totalmente en FLOSS o en porcentajes muy

32- Asimismo el diseño posee un módulo de Competencias y Organización del Trabajo reducido, algunas incorporaciones para la medición de la innovación que pueden ser propias del OS, y preguntas de vinculaciones típicas que siguen al Manual de Oslo. Sobre éstas últimas, se ampliaron a actores propios de la actividad del FLOSS, así como se incorporaron preguntas sobre las formas de relacionarse con la comunidad FLOSS.

33- El período de relevamiento culmina en diciembre de 2019, con una muestra objetivo de 180 empresas, dependiendo de la tasa de respuesta. Si ésta es baja respecto de la meta, se descartarán aleatoriamente casos para asegurar la representatividad geográfica nacional.

elevados (18 casos obtienen más del 80% de sus ventas del OS).³⁴

Tabla 1.2. Desarrollo de productos FLOSS y/o Prestación de servicios a partir de un FLOSS

	Cantidad de empresas	Porcentaje
Desarrollan/Prestan	43	42,2 %
No Desarrollan	62	59,0 %
Total	105	

Los tipos de actividades de desarrollo OS y de servicios a partir del FLOSS que realizan tienen frecuencias variadas, como ilustra la Tabla 3 a continuación. Con todo, la importancia de las actividades es relevante: el 18% de las empresas argentinas de software desarrollan programas completos de código abierto, un 32% de ellas realiza desarrollo de módulos o partes de programas OS, el 21% de las empresas del sector producen otro tipo de insumos OS (tales como librerías, bases de datos, etc.) y algo más del 26% herramientas de desarrollo o de infraestructura para uso interno.

Respecto de los servicios, más del 29% de las empresas de software argentina prestan servicios de consultoría tecnológica en base a algún FLOSS, y casi el 25% de ellas presta servicios de consultoría de sistemas (en infraestructura, servidores, redes, etc.) alrededor de un software de código abierto.

34- Recordar de que la muestra está aún en un parcial grado de avance, por tanto estos porcentajes se tratan de un indicador de importancia. Además, se trata de una muestra y no tienen carácter censal. Por lo que representan un indicador, más que una participación de la población del sector.

Tabla 1.3. Tipo de actividad de desarrollo de software y de prestación de servicios que realiza

	Cantidad de Empresas	Porcentaje sobre el Total de Empresas de Software (n=105)	Porcentaje sobre las empresas que proveen servicios o productos FLOSS (n=43)
Desarrollo de programas completos OS	19	18,1 %	44,2 %
Desarrollo de módulos o partes de programa OS	34	32,4 %	79,1 %
Desarrollo de Otros insumos OS (librerías, bases de datos)	22	21,0 %	51,2 %
Desarrollo de Herramientas de desarrollo/infraestructura interno OS	28	26,7 %	65,1 %
Desarrollo de FLOSS embebido en hardware	7	6,7 %	16,3 %
Presta servicios de consultoría tecnológica en base a un FLOSS	31	29,5 %	72,1 %
Presta servicios de consultoría de sistemas en base a un FLOSS	26	24,8 %	60,5 %

Por último, los resultados con respecto al uso de FLOSS en las empresas de software del sector son abrumadores. Como puede apreciarse en la Tabla 4, casi 4 de cada 5 empresas argentinas de software utilizan FLOSS en sus procesos de desarrollo, **independientemente de si venden soluciones propietarias o de código abierto**. Dos tercios de la muestra usan herramientas o productos OS, un 41% de las empresas utilizan código OS en el desarrollo de nuevo código y la mitad de las empresas de software usan algún otro tipo de insumo OS.

Tabla 1.4. Uso de herramientas o insumos open-source en los procesos de desarrollo

	Cantidad de Empresas	Porcentaje sobre el Total de Empresas de Software
Utiliza herramientas/productos OS para el desarrollo de software	70	66,7 %
Utiliza código OS en el desarrollo de código nuevo	43	41,0 %
Utiliza otro tipo de insumo OS (librería, server, bases de datos, etc)	53	50,5 %
Utiliza algún tipo de FLOSS	80	76,2 %
No utiliza FLOSS	25	23,8 %
Total	105	

Estas estadísticas preliminares de nuestra encuesta revelan un uso muy significativo del FLOSS en las empresas de software de la Argentina, así como una participación importante en su oferta productiva. El hecho de que en algo más de cuatro de cada diez empresas del sector al menos parte de sus ventas provengan de ofrecer desarrollos de software de código abierto o de servicios alrededor de un FLOSS, desarrollado por una comunidad o internamente, habla de un fenómeno

que dista de ser excepcional o “anormal” en el comportamiento competitivo de las empresas del mercado local.

3. Reflexiones Finales

La presencia corporativa dentro de la actividad del FLOSS se ha tornado un hecho notorio. Esto es respaldado tanto por estadísticas globales de la industria del software, como por estadísticas locales del sector en la Argentina y estudios teórico-conceptuales sobre el rol económico-productivo del OS. Aunque esta presencia empresarial en el FLOSS se ha manifestado crecientemente, la economía se ha preocupado en estudiar mucho más cómo se involucran los individuos que las empresas en los proyectos OS (Lerner y Schankerman, 2013). Los prejuicios subyacentes al *mainstream* en economía, que asumen la ubicua presencia de la guía de la optimización de la utilidad en el comportamiento social o la dogmática necesidad de los derechos de propiedad privada para el funcionamiento de la actividad económica e innovativa, tienden a abordar al comportamiento de las empresas en el FLOSS como “anormal”, y han diseminado una visión del fenómeno como marginal o excepcional en términos empíricos.

El presente capítulo aporta diversos elementos que contribuyen a la visibilidad cuantitativa del FLOSS, tanto en la industria global del software, como nacional. Asimismo, se han presentado y sistematizado distintos aportes de un cuerpo de literatura que trasciende disciplinarmente a la economía, que permitan conceptualizar al FLOSS en su dimensión productivo-económica no como un hecho excepcional o anómalo, sino como un fenómeno corriente en la competencia capitalista actual en sectores tecnológicos.

Ello abre el camino, o bien constituye un punto de partida, para un programa de investigación más amplio que comience a abordar más directamente formas de cooperación y competencia en el terreno de la producción abierta y colaborativa, y su impacto en el ritmo y calidad de la innovación. Sus prácticas se expanden a otra serie de actividades productivas, que van desde la fabricación de semillas a la de medicamentos, pasando por la investigación genética, la elaboración de enciclopedias, libros y la producción electrónica en plataformas de hardware de código abierto, hasta la decodificación del genoma humano (Dughera *et al.*, 2012, van Zwanenberg *et al.*, 2017). Dado que comprender el rol de las características y ritmo de la innovación en las economías capitalistas resulta un

aspecto central para el diseño de estrategias de desarrollo en economías periféricas como la Argentina, el estudio de las tensiones que el *open source* plantea entre las formas de propiedad privada y colectiva en la generación de innovaciones, son aspectos relevantes para ser estudiados y abordados en este contexto.