

# **Límites y potencialidades de la difusión de software libre en un país en desarrollo. El caso de la Argentina.**

**Verónica Robert<sup>1</sup>**

## **1 Introducción:**

En los últimos años, luego de que se generalizara su uso y producción y de que se instalara la reflexión sobre el tema en ámbitos tecnológicos, ha surgido el interés de economistas por el fenómeno del software libre (SL) o software de fuente abierta (free software u open source software). Fenómeno porque para la economía se presenta como un hecho anómalo que pareciera desafiar las leyes que esta disciplina considera centrales, entre ellas la racionalidad de los individuos objetivada en la búsqueda de la maximización de la utilidad.

Sin embargo, la difusión que ha adoptado en el último tiempo, así como el involucramiento de grandes actores (tales como grandes empresas productoras de software y hardware) estrictamente guiado por la lógica del mercado, condujo a que muchos autores dejaran de lado su perplejidad y empezaran a intentar explicar y justificar no ya el intrínquis, sino qué posición adoptar frente a esta nueva forma de producción, distribución, y organización del trabajo en un área relativamente nueva como la informática.

Los numerosos trabajos que han tratado la problemática del SL demuestran que se trata de un fenómeno complejo y por lo tanto las formas de abordarlo son diversas. Muchos autores han encarado su estudio desde una visión positiva y se han centrado en dar explicación a la cuestión de los incentivos y la racionalidad; otros se han volcado por una línea más normativa a partir de la comparación y competencia entre los dos modelos de desarrollo y distribución de software (libre / fuente abierta y propietario).

El siguiente trabajo, si bien analiza los argumentos centrales de ambas corrientes, se enmarca en la segunda, contribuyendo a: (i) la discusión sobre las motivaciones, (ii) las implicancias del fenómeno sobre la evolución del sector y (iii) la identificación de los elementos que influyen sobre la difusión del SL. El objetivo central de esta investigación es analizar el fenómeno desde la circulación de conocimiento y la creación de competencias, haciendo especial hincapié en las ventajas de su difusión en países en desarrollo. Para ello, analiza las posibilidades de la difusión del SL en la Argentina a partir de datos originales recopilados a través de una encuesta electrónica a trabajadores informáticos argentinos.

Partiendo de la base los estudios realizados sobre las características de del sector de software y servicios informáticos (SSI) en la Argentina, y sobre la problemática del SL, las hipótesis centrales del trabajo son que: (i) las motivaciones para participar en desarrollos libres no están asociadas a retribuciones más o menos concretas sino a la

---

<sup>1</sup> Este trabajo fue realizado en el marco de la Tesis de Maestría en Economía y Desarrollo Industrial de la Universidad Nacional de General Sarmiento.

construcción de un espacio cuasi-público del que los participantes obtiene beneficios indirectos e directos, (ii) estos agentes constituyen una comunidad epistémica en el sentido de que el conocimiento que circula al interior de la red no es un bien de libre acceso, sino que este conocimiento codificado y tácito que circula en la red les permite diferenciarse, construir capacidades y ventajas competitivas; (iii) en este sentido la difusión de SL en el ambiente local contribuiría al desarrollo de capacidades y por lo tanto al crecimiento del sector con elementos locales propios e idiosincrásicos, es decir determinados grados de autonomía en la elección del sendero tecnológico; (iv) no obstante las características actuales –en términos de capacidades y vinculaciones- de los agentes sobre los que recae la decisión de adopción de SL (los trabajadores informáticos) así como la trayectoria tecnológica de las organizaciones para las que trabajan, limitan una difusión más profunda del mismo.

La información empírica se analiza bajo la hipótesis de que la difusión de SL en la argentina depende de la receptividad -capacidad de los agentes de ingresar a la comunidad epistémica- y la conectividad -que considera los vínculos con otros agentes y cómo esto influye sobre su decisión de adopción- de los trabajadores informáticos, mientras que ambas dimensiones se retroalimentan positivamente. Sin olvidar que la difusión se produce en un ambiente con economías de escala en la adopción, costos crecientes de migración (switch), posibilidades de lock-in, etc; y sus efectos sobre las estructuras de mercado.

El trabajo concluye con una breve discusión sobre el tipo de políticas a aplicar en con el objetivo de desarrollar competencias tecnológicas e incrementar la difusión de SL.

## **2 Breve historia del SL y algunas definiciones, o ¿dónde radica el enigma del SL?**

En las primeras décadas desde el nacimiento de la informática, el software estuvo fuertemente ligado al hardware sobre el que se ejecutaba y era considerado un valor agregado a éste más que un producto en sí mismo: los grandes inversores en el desarrollo de software eran los productores de hardware donde iba a correr. La estrategia comercial de estas compañías pioneras radicaba justamente en el establecimiento de plataformas propietarias, verticalmente integradas desde el hardware hasta software, que les permitían detentar poder de mercado sobre los productos asociados a la plataforma e incrementar los costos de migración. Las explicaciones económicas del éxito de una plataforma enfocaron la problemática tanto del lado de la oferta como de la demanda, haciendo especial hincapié en las elecciones estratégicas en el marco de economías de red, economías de escala, sponsoreo, barreras crecientes de salida y aprendizaje (Arthur 1989, Metcalfe 1988, Katz y Shapiro 1968).

Sin embargo, circunstancias más o menos fortuitas, sumadas a la inestabilidad inherente del modelo de plataforma propietaria vertical condujeron a que se transitara hacia un modelo donde el control pasó del hardware al software. Por una parte, el éxito en practicar ingeniería reversa sobre la PC IBM, y el fallo que otorgó legitimidad a la producción de clones PC, condujo a la desarticulación del monopolio del hardware y a la caída de los precios de las computadoras- Lo que funcionó como un importante impulso la industria del software, que contó con un mercado más amplio que el que proponían los desarrollos específicos para un tipo determinado de computadoras. Por otra parte, Unix,

DOS y Windows aparecieron en el mercado como sistemas operativos independientes del hardware y desplazaron del control de mercado a los productores de hardware al constituirse como una nueva “capa” en la comunicación entre el usuario y la máquina. De tal forma, los nuevos productos de software pasaron de montarse directamente sobre el sistema operativo sin tener necesidad de comunicación directa con el hardware (West 2003).

Hacia de la segunda mitad de los años 70 las empresas tecnológicas empezaron a entender el software como un bien comercializable, y, en un tiempo record, Microsoft, el paradigma de empresa productora de software, pasó de ser un grupo de entusiastas universitarios a una de las empresas más exitosas de los últimos tiempos.

La compañía AT&T que había desarrollado Unix, no pudo lanzarse al mercado de productos informáticos por un viejo fallo anti-trust de 1956 en su contra que se lo impedía. Por lo tanto durante 15 años Unix estuvo confinado al uso interno, a la investigación académica en universidades y a un número muy reducido de compañías y vendedores de hardware que habían adquirido licencias sobre el código fuente. Sin embargo, más adelante, consiguió comercializar Unix masivamente a través de una empresa separada que fue vendida a Novell y luego a SCO (West 2003). El movimiento del software libre comenzó como una reacción a la política de AT&T de vender Unix y restringir el acceso de la comunidad académica a su código fuente; y contra la incipiente comercialización de software en general.

Habitualmente se considera el manifiesto escrito en 1983 por el entonces programador del prestigioso laboratorio de inteligencia artificial del MIT, Richard Stallman, como el punto de partida de todo el movimiento. Stallman redactó luego la *GNU General Public License* (GPL), una manera de licenciar software en la que el titular del derecho de autor le otorga al usuario la facultad de modificar el código fuente del programa, a condición que, de hacer públicos productos derivados del original, también haga públicos los fuentes de los nuevos programas licenciados bajo la misma GPL. Stallman inició entonces el proyecto GNU, que consistía en construir un sistema informático completo cuyos componentes estuvieran todos bajo el gobierno de la GPL. Fundó también la *Free Software Foundation* (FSF), institución sin fines de lucro encargada de llevar adelante el proyecto GNU y de proveer un marco legal para el desarrollo de software libre (fsf.org)

No obstante el crecimiento del SL dentro de la FSF estaba confinado a los expertos y no cobró notoriedad hasta que en 1991 Linus Torvalds, un estudiante finlandés, comenzara a escribir un kernel (núcleo del sistema operativo) compatible con los kernel Unix propietarios. Este kernel se llamó Linux y llenó el vacío más grande del proyecto GNU, dando origen al sistema conocido comúnmente como Linux o también GNU/Linux. Linux es sin duda la cara más conocida del software libre por el público general, y atrajo sobre sí la atención de los medios y de las grandes empresas que no había logrado la FSF. Luego, con la aparición de las "distribuciones"<sup>2</sup> y de avanzados entornos gráficos de

---

<sup>2</sup> Versiones de GNU/Linux con multitud de aplicaciones empaquetadas de forma de facilitar su instalación y uso; las más conocidas Red Hat, SUSE, Mandrake y Debian.

escritorio<sup>3</sup>, Linux terminó de salir del ámbito del hobby y la academia para competir fuertemente en el mercado de servidores y tímidamente en el de las computadoras personales. En 1997, Eric Raymond, en su influyente colección de ensayos ‘La catedral y el bazar’, analizó el modelo de desarrollo del SL y sus posibilidades a futuro dentro de la industria tecnológica. Los últimos años han visto tanto el apoyo decidido de ‘pesos pesados’ en la IT, como IBM, Sun, HP y Apple, como el surgimiento de jóvenes empresas dedicadas enteramente al software libre, como Red Hat o VA Linux.

### *Definiciones*

En el corazón de la definición de SL está la facultad que se le otorga al usuario de usar sin restricciones el software, incluyendo las posibilidades de inspección y modificación su código fuente. Estos derechos se especifican en la licencia, que se establece como un contrato entre el tenedor de los derechos de propiedad intelectual y el usuario del software, y pueden ser más o menos permisivos. Por lo tanto, cada licencia de software libre o de fuente abierta implica una definición sobre qué se considera como SL o de fuente abierta, y esto último está inevitablemente asociado a su propia historia y a las posiciones políticas y filosóficas de los movimientos que promueven su desarrollo y difusión, pero también crecientemente a las condiciones de mercado<sup>4</sup>.

Podemos hablar dos corrientes relativamente enfrentadas dentro de esta comunidad. Una es la del “software libre”, que prioriza sobre todo el aspecto ideológico del código abierto, cuyo ideal es cubrir todas las necesidades informáticas de individuos y empresas sin necesidad de software cerrado o propietario. El referente más claro de esta postura es el propio Richard Stallman. La otra es la del “open source”, centrada más en los beneficios tecnológicos y económicos del código abierto. Esta postura, generalmente identificada con Eric Raymond, es obviamente la que más agrada a las grandes empresas que apoyan el SL pero tienen aun un modelo de negocios basado en el software propietario.

En cuanto a las licencias de SL existe dos grades paradigmas que se condicen con cada una de las dos corrientes antes mencionadas. Por una parte, las más cercanas al modelo de la GPL y por otra las del tipo BSD. El paradigma GPL está asociado al movimiento SL, mientras que las licencias tipo BSD asumen una postura más cercana a las ideas del “open source”. La gran diferencia entre ambas es que este último tipo permite a terceros no sólo la modificación y redistribución, sino también la modificación y redistribución en forma de software propietario, no permitido por la GPL. El modelo GPL, obliga a que las modificaciones distribuidas de software bajo GPL sean GPL también, por lo que se dice que este tipo de licencia es “viral” ya que contagia a los programas de computadoras que se desprenden de ella. Asimismo, este modelo es coherente con el objetivo de la FSF (autora de la licencias GPL) de cubrir todas las necesidades informáticas con software libre.

---

<sup>3</sup> Tales como Gnome y KDE, que incorporaban los sistemas de ventanas.

<sup>4</sup> Es común encontrar SL bajo licencias mixtas que contemplan tanto la posición de SL como la de OSS que discutiremos más adelante. Estas licencias mixtas lograron popularidad porque las incorporar ambas posiciones amplía su público apuntado a su principal cometido: que el software sea utilizado por el mayor número de usuarios posible.

A pesar de las diferencias planteadas, ambas corrientes insisten en diferenciar gratuidad de libertad aunque también plantean un cuestionamiento sobre el modelo de negocio basado en licencias propietarias que, según estas corrientes, estaría limitando la libertad de los usuarios de conocer el funcionamiento de programa, al tiempo que impediría desarrollos de software más veloces y de mejor calidad al restringir el acceso del usuario al código fuente.

Ya sea este SL u open source el sentido del ser del software abierto -en oposición al cerrado- radica en una idea muy simple: cuando los programadores pueden leer, redistribuir y modificar el código fuente de un programa de computación, el programa evoluciona. Lo pueden mejorar, adaptar y reparar sus errores a una velocidad sorprendentemente superior a la que se podría llegar a través del método convencional de desarrollo de software (Raymond 1999). De acuerdo con el movimiento “open source” este proceso evolutivo veloz produciría mejor software que el modelo cerrado tradicional, en el que sólo unos pocos programadores pueden ver la fuente y mientras que el resto está limitado a “usar ciegamente una opaca bola de bits” (opensource.org 2004)<sup>5</sup>.

### **3 Qué tienen para decir los economistas**

La literatura que intenta abordar el fenómeno del software libre desde una perspectiva económica, tiene dos vertientes distintas. Una de ellas aborda la cuestión desde una perspectiva positiva y la otra normativa.

La preocupación central de la corriente positiva pasa por buscar una explicación al problema de por qué agentes racionales se involucren en estos proyectos de desarrollo que implican un compromiso sostenido e intenso con una actividad no contractual y no remunerativa. Básicamente busca resolver el dilema de por qué estas personas renuncian a la posibilidad de lucrar económicamente ejerciendo el monopolio que otorgan los derechos de propiedad intelectual de su creación y asumen el producto de su trabajo como de propiedad colectiva y abierta a nuevos cambios y contribuciones (Lerner y Tirole 2002). En este sentido, el problema de la literatura económica positiva sobre SL se aboca a la cuestión de las motivaciones.

La segunda asume este “capricho” (Dalle y Jullien 2003) como un modelo alternativo de organización del trabajo (Benkler 2001) y producción de software, y analiza las ventajas y desventajas del uso y generalización del SL para, primero, el desarrollo tecnológico del sector, y segundo, sus implicancias en el desarrollo económico y social.

Dentro de la primera corriente se desataca el artículo de Lerner y Tirol (2002). Este trabajo, a partir de tres estudios de caso (Apache, Perl y Sendmail), intenta explicar por qué los programadores contribuyen a la producción de un bien público. Utilizando

---

<sup>5</sup> A pesar de que en este trabajo utilizemos las siglas SL (software libre) lo hacemos de forma genérica para referirnos indistintamente al software libre como al de fuente abierta. Para los objetivos de este estudio tal diferencia no es relevante ya que ponemos énfasis en la comparación de las posibilidades de aprendizaje del modelo abierto o libre en oposición al cerrado.

elementos de la economía laboral, particularmente la literatura sobre “career concerns”<sup>6</sup>, explica algunas características de los proyectos de desarrollo de SL. Los autores también utilizan la noción de “user-driven innovation” de la teoría de la innovación (Rosenberg 1976) para describir cómo el rol de los usuarios puede acelerar el proceso de innovación. En este sentido, los programadores serían usuarios y desarrolladores al mismo tiempo por lo que ajustarían el producto a sus necesidades específicas<sup>7</sup>.

El trabajo parte de la premisa de que los programadores maximizan racionalmente sus funciones de utilidad en el momento de decidir qué contribución hacer al SL. De tal forma justifican la decisión de participar en desarrollos de SL sobre la base de incentivos acerca del futuro de su carrera profesional, gratificación del ego, etc. Por otra parte, a partir del concepto de “user-driven innovation” explica la importancia para los programadores de introducir mejoras sobre sus herramientas de trabajo. Efectivamente, estos autores destacan el hecho de que los programas de software libre de mayor difusión son aquellos cuyos usuarios son programadores expertos y difícilmente se encuentren disponibles programas libres competitivos para usuarios ajenos al mundo de la informática<sup>8</sup>.

Si bien estos elementos teóricos aportaron a la comprensión del fenómeno, resultaron insuficientes al no lograr una explicación general para sus múltiples facetas. Por ejemplo, el hecho de que cada vez se vea más los programas libres para usuarios finales preparados para una instalación fácil que no requiera conocimientos de informática, o que grandes empresas de software y hardware participen en el desarrollo de fuente abierta y/o “abran” código de sus programas propietarios, incluso la existencia de empresas dedicadas al desarrollo de software libre compitiendo en el mercado. En este último sentido, los trabajos dentro de la corriente positiva deberán explicar las motivaciones de los grandes actores y de las empresas más pequeñas que dedican su actividad al desarrollo de SL y conciliarlas con las motivaciones de los programadores individuales, ya que a simple vista pueden apreciarse las diferencias.

En la misma línea que el artículo de Lerner y Tirol, estaría el artículo de Baldwin (2002) que busca las “transacciones” en la economía del SL -o sea qué obtienen a cambio los programadores que dedican tiempo al desarrollo de un bien público-. El trabajo de Hann, et al (2002), por ejemplo, afirma que los retornos (monetarios) por la participación en proyectos libres tardan, pero llegan. También dentro de la línea del “career concern”, de acuerdo con estos autores, las contribuciones al SL se traducen en mayor prestigio y éste en mejores oportunidades laborales y mayores salarios. Por su parte, el trabajo de Varian (2001), demuestra que un sistema que se sustente en la provisión puramente voluntaria nunca puede ser confiable, por los inconvenientes de la conducta “free rider”, con lo que,

---

<sup>6</sup> La bibliografía sobre career concerns surge como una rama dentro de la teoría de los incentivos que se intentado desarrollar desde la organización industrial. Los trabajos sobre career concerns sugieren que muchos de los incentivos en organizaciones no se presentan a través de incentivos explícitos en contratos formales sino implícitos y vinculados con la preocupación por la carrera profesional (Dewatripont et al. 1999).

<sup>7</sup> El trabajo de von Hippel (2002) también aplica este concepto al desarrollo de SL.

<sup>8</sup> Si bien en un comienzo puede identificarse este patrón, cada vez son más los programas libres para el público general.

a partir de este aporte, se puede afirmar: (i) que el fenómeno de SL se afirma en algo más que un capricho o simple altruismo o (ii) que el modelo del bazar estaría condenado al fracaso. Por último, el trabajo de Gilles Saint-Paul (2002), alejándose de la postura estrictamente positiva, analiza el efecto de las innovaciones no propietarias sobre el desarrollo económico. De acuerdo al autor del trabajo éstas serían contraproducentes ya que predominaría un efecto “desplazamiento” de la innovación filantrópica sobre la propietaria, lo que implicaría un menor ajuste a las necesidades del usuario final. Al mismo tiempo, los desarrollos filantrópicos implicarían un menor bienestar social por la duplicación de esfuerzos ya que en muchos casos se trataría de trasladar invenciones propietarias a nuevos esquemas de derecho de propiedad intelectual. Frecuentemente se discute qué tan innovador es el SL; es decir, es más habitual encontrar programas propietarios para los que no haya una contra parte libre o abierta, que el caso opuesto. Esto abre la discusión a cerca del rol innovador del SL dentro de la producción de software. Al respecto se puede destacar que el modelo del bazar implica ante todo una innovación de proceso en la media en que se trata de una forma radicalmente distinta de desarrollo de software.

Los trabajos que se enmarcan en el segundo grupo discuten la competencia entre ambos modelos de producción y desarrollo del software. En este sentido las discusiones pueden ser enmarcadas en diferentes esferas o niveles. Entre ellos el nivel (i) tecnológico-organizacional, (ii) económico, (iii) institucional, (iv) de políticas públicas, entre otras.

El nivel técnico o tecnológico y organizacional, en el que se discute la posibilidad de que uno de los modelos sea técnicamente superior, en este sentido frecuentemente se argumenta que la forma de organización del trabajo de modelo del bazar conduce a productos finales más confiables (Raymond 1999; Benkler 2002). En el nivel económico se analiza la relación costo beneficio de la utilización de SL, en este caso los trabajos frecuentemente recogen las especificidades del mercado local de SSI, ya que la variabilidad de los costos estará asociada a la disponibilidad de trabajadores capacitados. Dentro del nivel institucional se discute la relevancia del SL como institución económica; Dalle y Jullien (2003) sugieren que el SL apunta a un nuevo modelo económico e institucional de comunidades creativas o epistémicas, que puede ser comparado con las patentes o la corriente “open science” en la comunidad científica. Otro nivel de discusión se refiere al rol del Estado frente al fenómeno, en este sentido son diversos los argumentos por lo cuales es Estado debería adoptar una actitud activa frente al SL (ver infra) estos van desde la administración de los datos públicos a la determinación de estándares pasando por cuestiones tales como el control de la competencia en el mercado de SSI hasta la promoción de la actividad por sus efectos dinámicos sobre el aprendizaje y la generación de competencias.

#### **4 Esquema de incentivos**

Sobre la cuestión de los incentivos ha rondado mucha de la literatura sobre el SL, - algunos de estos artículos ya los hemos mencionado. Dentro del enfoque económico sobre el SL la cuestión de los incentivos resulta clave para resolver su dilema. Es decir: por qué se hacen contribuciones a la construcción de un bien público sin aparente retribución alguna por el trabajo aportado.

Sin lugar a dudas, la cuestión de los incentivos supone determinada hipótesis de racionalidad que gobierne las conductas de los agentes. La bibliografía sobre incentivos, particularmente tratada desde el institucionalismo norteamericano frecuentemente no hace explícito su supuesto acerca de la racionalidad de los agentes. Por ejemplo el trabajo ya mencionado de Lerner y Tirole no se aleja de la hipótesis con racionalidad optimizadora e instrumental. De tal forma, su percepción sobre la conducta de los agentes se circunscribe a lo que esta perspectiva les permite ver. Agentes que maximizan intertemporalmente sus oportunidades de trabajo sujetos a una restricción de salarios presentes y futuros, eligen participar de la esfera del SL pensando en futuro profesional y guiados por una lógica estrictamente mercantil.

Por otra parte, los artículos auto reflexivos del movimiento SL/OSS (Stallman, Raymond) permiten asumir que la hipótesis de racionalidad que gobierna la conducta de estos agentes trascienden a la maximizadora. En este sentido, por ejemplo, los valores que se construyen en la comunidad, basados en el respeto mutuo por el trabajo del otro, la posibilidad de participar compartiendo el trabajo propio, intercambiando experiencias y sugerencias, alejan a los incentivos que influyen directamente sobre desarrollo personal e implica otros objetivos que trascienden una visión individual.

La auto denominación de *comunidad* por parte de estos grupos implica la importancia que le otorgan al sentido de pertenencia. Incluso la rivalidad entre los movimientos SL y OSS, involucra cuestiones que trascienden a las técnicas para instalarse en el terreno de lo político e incluso filosófico. En tal sentido, resulta incorrecto afirmar que la motivación puede circunscribirse a una cuestión material, aunque también resulta insuficiente una explicación netamente altruista. Sobre todo a partir de la fuerte difusión del fenómeno y la aparición de grandes actores económicos en escena.

Otra manera de entender los incentivos de los programadores para realizar aportes al desarrollo de programas libres es preguntando directamente por qué lo hacen. El estudio FLOSS (2002) aporta datos al respecto que son coherentes con otros hallazgos empíricos (BCG 2000, Relevamiento UNSG-SADIO 2004). Algunas de las conclusiones del estudio FLOSS acerca de las motivaciones son que:

- El 78,9% de los desarrolladores de programas libres o abiertos afirma que dentro de sus motivaciones principales para comenzar a desarrollar código libre ocupó un lugar central la de aprender y desarrollar nuevas habilidades, mientras que el 40,8% comenzó con esta tarea para compartir su conocimiento. Solo el 4,4% empezó a desarrollar SL con el objetivo de hacer dinero<sup>9</sup>.
- El 70,5% afirma que entre sus motivaciones para continuar a desarrollando código libre estuvo aprender y desarrollar nuevas habilidades, mientras que el 67,2% lo hace para compartir sus conocimientos. Mientras que el porcentaje de encuestados que continuó por motivos monetarios es el 12,3%.

---

<sup>9</sup> Cabe mencionar que los encuestados podría marcar hasta cuatro motivaciones principales, por ello los porcentajes no suman 100%.



- A pesar de esto, más del 50% de los encuestados gana dinero de alguna forma a partir del desarrollo de SL.

El trabajo realizado por el BCG basado en una encuesta a cerca de 500 desarrolladores identifica cuatro grupos diferentes diferenciados de desarrolladores que se diferencian entre sí por las motivaciones principales de cada caso. De acuerdo con este trabajo los grupos serían:

- Los creyentes (believers) cuya principal motivación es que según ellos el software debería ser libre. Este grupo está compuesto por el 33% del panel y tienen como segunda motivación principal desarrollar capacidades.
- Los que buscan mejorar sus capacidades (skill enhancers), este grupo está compuesto por 21% de la muestra (a pesar de que la razón mejorar las capacidades fue señalada como una motivación principal por el 43% del muestra). La segunda motivación principal de este grupo es el estímulo intelectual de participar en estos desarrollos.
- Los que buscan divertirse (fun seeking), 25% de la muestra, entre ellos las motivaciones centrales son la estimulación intelectual, porque resulta divertido, para trabajar en equipo y para mejorar sus capacidades. Cabe destacar que dentro de la encuesta la estimulación intelectual como motivación principal recibió el 43% de las respuestas.
- Los que hacen como una tarea profesional (professionals) son el 21% de la muestra, y sus motivaciones son hacerlo porque es trabajo, por el estatus profesional y por el estatus que les significa trabajar en la esfera del software libre. El 30% del panel señaló como motivación principal que se trataba de un trabajo.

Aquí nuevamente puede señalarse la importancia de la auto capacitación a la hora de decidir participar en el desarrollo de SL. Tanto el grupo de los skill enhancers como de los professionals aparece la cuestión de la formación como algo central, mientras que en los otros dos grupos si bien no es central, aparece como segunda motivación principal. Esto muestra por una parte que las motivaciones no son únicas y predecibles sino diversas y que responden a cuestiones vinculadas con el interés personal pero también con cuestiones políticas como prima en el grupo de los believers. Por otra parte, muestra la vinculación entre el desarrollo de capacidades y el desarrollo de SL.

Por último algunos números de la encuesta realizada por la UNSG-SADIO a cerca de 200 trabajadores informáticos argentinos reafirman estas observaciones.

- Entre las personas que desarrollan SL desarrollar habilidades apareció como motivación principal (primeras tres) el 21% de las veces.
- En el segundo y tercer puesto estuvo el compartir conocimiento y mejorar las oportunidades laborales que fueron mencionadas 12% de las veces cada caso.

- Por otra parte el 71% de los encuestados que desarrollaban SL, afirmaron que ganaban dinero de su actividad en el campo del SL, directamente (al 52% de los que desarrollan SL, les pagan por hacerlo o por proveer servicios asociados) o indirectamente (el 19% de este grupo).

El supuesto de este estudio sobre las motivaciones se encuentra a mitad de camino entre el altruismo y el capricho y la búsqueda del beneficio económico. El error que cometen las hipótesis acerca de las motivaciones guiadas por el altruismo y el beneficio económico se hallan en considerar al SL como un bien público. Los reservorios de código libre o simplemente abierto que pueden encontrarse en la web no son un bien público porque no verifican las dos cualidades que definen a este tipo de bienes: consumo no rival y no excluible. Se trata de un bien cuyo consumo es no rival, ya que no sufre desgaste alguno con el uso y la copia digital implica su reproducción casi sin límites a costos ínfimos. Sin embargo no está tan claro que su consumo sea no excluible. Si bien en términos hipotéticos cualquier persona (con acceso a la red) tiene a su disposición el inmenso reservorio de código abierto y libre, las posibilidades de hacer un uso efectivo de este material están limitadas a las personas que tengan la capacidad para manipular, adaptar, compilar y poner en funcionamiento los programas para computadoras que ese código encierra. Evidentemente no todos tenemos las capacidades para hacer esto posible y por lo tanto una barrera delimitada por las competencias distancia el uso efectivo de este supuesto bien público.

En este sentido, habría que reformular la idea de que el SL sea un bien público por la de que se trata de un bien club<sup>10</sup>, en donde para acceder al club hace falta contar con los conocimientos necesarios. La visión de la comunidad de desarrolladores de SL es compatible con esta noción de bien club.

De esta forma, los aportes que realizan al corpus de código pueden ser leídos como una contribución a la construcción un espacio cuasi público desde el cual pueden extraer y devolver y hacer uso económico del mismo otorgándole dinamismo a conjunto. Quienes pertenecen al club son quienes están capacitados para hacer un uso económico de este material y por lo tanto probablemente (aunque no obligatoriamente) retribuyan (Zeitlyn 2003) con aportes personales. Sin embargo, la dinámica de aportes y usos no está liderada por una lógica de reciprocidad, en realidad entran en juego cuestiones tales como compartir conocimientos y habilidades, obtener reconocimiento personal, la posibilidad de realizar desarrollos conjuntos, etc. No existe una tasa de cambio entre horas de programación invertidas en SL y la cantidad de código abierto del que se pueda hacer uso; o entre horas de debugging de código ajeno y aprendizaje y desarrollo de habilidades o gratificación del ego. Ni siquiera una tasa de sustitución en una supuesta función de utilidad del hacker. Por una parte, las encuestas demuestran que es difícil establecer un orden de prioridades, por otra, la trascendencia de la existencia del 'otro' y de la comunidad en la labor anulan posibles especulaciones individuales del tipo costo beneficio.

---

<sup>10</sup> Hawkings (2002) habla del SL como un bien cuasi público.

En este sentido puede hablarse de una comunidad epistémica (David y Forey 2002) en donde entre los miembros existe una importante actividad de creación y reproducción de conocimiento, articulada a través de mecanismos específicos<sup>11</sup> para difundir el conocimiento que se genera.

los miembros intercambian conocimiento pero no siguiendo una lógica de maximización de utilidad sino de contribuir al crecimiento de la comunidad y su diferenciación con el resto. Persiguen la consolidación de la comunidad porque ahí encontrarán el mayor beneficio colectivo, aunque este no sea compatible con el individual. En este sentido el conocimiento transforma en un bien

Algunos de Raymond como la Cultivando por la Noosfera, la Catedral y el bazar o Una breve historia de la cultura hacker<sup>12</sup>, que analizan las motivaciones de los desarrolladores voluntarios de SL, son coherentes con esta la idea de una valoración mediata de su trabajo, la importancia de la reputación, el comportamiento de la comunidad y la existencia de reglas y pautas dentro de ella.

En especial el primer trabajo mencionado, Cultivando la Noosfera, donde el autor analiza la cultura hacker comparándola con las relaciones donáticas o de intercambio puro. De acuerdo con Raymond “queda muy claro que la sociedad hacker de código abierto es de hecho una cultura de regalos” (relación donática), dentro de ella, no hay seria escasez de necesidades de sobrevivencia y el software es compartido gratuitamente. Según este autor, esto crea una situación de abundancia en la que la única medida de éxito es la reputación entre sus pares<sup>13</sup>. En este sentido, define a la Noosfera como el territorio de las ideas o el espacio de los proyectos software, donde los hackers cultivan sus propios proyectos y contribuyen en los desarrollos de otros. La palabra propiedad en este espacio contiene un significado por completo diferentes y se refiere más a la tutela del proyecto que a la propiedad física del mismo. Sobre esta base el autor habla de las reglas que regulan este espacio, como se hacen las contribuciones, como se establecen los roles y como se seleccionan los proyectos para participar. En definitiva habla de las reglas implícitas del club.

## **5 Bajo qué condiciones es deseable la difusión de SL en la Argentina**

Muchas veces hablar de SL en la Argentina parece ciencia-ficción, particularmente cuando se habla de su producción. A pesar de que se trate de un bien cuasi-público el porcentaje de contribuciones realizadas por programadores argentinos es ínfimo. Así lo demuestran las encuestas internacionales a desarrolladores, como la encuesta de UNGS-

---

<sup>11</sup> David y Fofey hablan de elevada utilización las de nuevas tecnologías de la información y comunicación.

<sup>12</sup> Cabe aclarar que a pesar de que hoy el apelativo hacker contenga un significado peyorativo y con él suela designarse a las personas que rompen sistemas de seguridad, el autor lo utilizar en el viejo sentido –ampliamente difundido dentro de los círculos más técnicos- de persona de gran capacidad intelectual y de alto reconocimiento entre sus pares. De aquí en más el término hacker será utilizado con este significado.

<sup>13</sup> Incluso afirma que “puede que no trabajes por reputación, pero la reputación es un pago real, con consecuencias si haces el trabajo bien”, es decir la reputación implica trabajos mejor remunerados y más oportunidades laborales producto del mejor posicionamiento en el ambiente.

SADIO a trabajadores informáticos en donde sólo alrededor del 10% del panel manifestó haber hecho contribuciones al SL. Sin embargo cuando el énfasis es puesto no sobre el desarrollo sino sobre la difusión, el estudio del fenómeno a nivel local cobra mayor significado.

Las implicancias sobre la definición de los derechos de propiedad en una sociedad de la información, la repercusión de esto sobre las estructuras de mercado en los sectores de nuevas tecnologías y sobre las posibilidades de aprendizajes y desarrollo de capacidades en los individuos y en las organizaciones locales, redefinen la importancia del fenómeno para la realidad local. Específicamente a partir del posicionamiento de los países en desarrollo frente la actual revolución tecnológica en donde la informática (junto con otras actividades, como la biotecnología) juega un rol central y en donde los países desarrollados se han ubicado como proveedores monopólicos en estos nuevos sectores. En este sentido la competencia entre plataformas abiertas y cerradas<sup>14</sup> conduce a enriquecer la discusión sobre cómo encarar la construcción de una sociedad de la información y el conocimiento (Katz 2003) en los países menos adelantados y sobre las perspectivas y diferentes formas de transferencia tecnológica desde los países desarrollados a los países en desarrollo.

Hablar de las motivaciones sobre el desarrollo de SL en los párrafos anteriores nos permitió aclarar que el fenómeno del SL no es marginal sino que ya ocupa un lugar importante en el sector y que no parece ser pasajero sino que se instala sobre bases que tienen a afirmarse cada vez más. A partir de este análisis nos proponemos en los párrafos que siguen estudiar las posibles ventajas y desventajas de la difusión de SL y el nivel que ha alcanzado en el ámbito local.

Las ventajas de la difusión del SL fueron analizadas principalmente por la corriente de software de fuente abierta, quienes pusieron mayor énfasis en los beneficios de esta forma de producción y distribución del software (Raymond 2000). También realizaron aportes sobre las ventajas de la difusión del SL desde la economía, particularmente los vinculados con la economía de innovación e institucional (Dalle y Julien 2003, Lecocq y Demil 2002). Otras contribuciones interesantes pueden encontrarse en la labor legislativa -leyes y discusiones parlamentarias- de algunos países que se han interesado en la promoción del fenómeno desde la esfera estatal (Villanueva Núñez 2001, IDA 2002, FLOSS 2002).

Frecuentemente cuando se habla de Software Libre se confunde la libertad con la gratuidad a partir de la ambivalencia de la palabra inglesa 'free'. Lo primero que cabe aclarar cuando se explican los beneficios económicos del SL es que éste no es por definición gratis. Es posible y deseable, desde un punto de vista de la sustentabilidad del modelo, la utilidad comercial del código libre. Como ya hemos explicado, las distribuciones de software libre se comercian aunque evidentemente, a precios sensiblemente más reducidos que las licencias de software propietario. Esto es así porque habitualmente los modelos de negocios basados en SL se fundamentan en la prestación de un servicio más que en la venta de licencias. Así como el SL plantea una competencia

---

<sup>14</sup> Cabe destacar que no se trata de una posición tan dicotómica sino que hay puntos intermedios y de convivencia que hacen que la evaluación entre las alternativas posibles no sea tan sencilla (West 2003).

en la forma de desarrollar software: la catedral y el bazar; también plantea una competencia en los modelos de negocios de la comercialización de software: uno basado en el licenciamiento de la propiedad intelectual (software propietario) y otro en la prestación de una gama servicios conexos al desarrollo de software, desde asesoramiento hasta adaptación y realización de desarrollos propios, instalación, training, soporte, etc.

Diferentes trabajos han estudiado los modelos de negocios basados en SL (FLOSS 2002; Hawkins 2002, Pal y Madanmohan 2002, Valduriez 2002) , un punto común de todos ellos es que los modelos se basan en que el SL es un bien cuasi-público, es decir en la asimetría en conocimiento (contar con las capacidades para realizar las adaptaciones necesarias e dejar funcionando el software en las máquinas de los usuarios finales) y en información (conocer la oferta de SL) entre los proveedores de SL y los usuarios finales.

A primera vista parecería que el uso de SL implica necesariamente un ahorro con respecto al uso de software propietario, aunque no implique gratuidad dado que no es necesario pagar licencias. Sin embargo, esto puede ser una percepción engañosa, ya que el licenciamiento del software constituye sólo una porción reducida del costo total que implica una tecnología. Este costo total, conocido como *Total Cost of Ownership* (TCO) incluye costos de mano de obra calificada, de administración, de hardware, etc, no es fácil de calcular, y los estudios suelen ser intencionalmente parciales hacia uno u otro lado (estimaciones de este tipo pueden encontrarse en las discusiones parlamentarias sobre aplicaciones de SL en el ámbito público). Podemos aceptar que el TCO del SL y del software propietario son lo suficientemente similares como para no permitir generalizaciones considerando el contexto de las economías industrializadas. Sin embargo, la situación es muy distinta en los países emergentes, especialmente aquellos que han logrado un buen nivel educativo y cierto desarrollo tecnológico (entre ellos la Argentina). En estos países, al estar las licencias de software propietario cotizadas en dólares, y la mano de obra calificada en moneda local, se vuelve más ventajoso invertir en SL. Otro beneficio es que los dividendos quedan en la economía local, en lugar de ser girados a los países centrales donde están las casas matrices de las grandes productoras.

El SL representa un claro beneficio económico para la industria tecnológica considerada a nivel global. Esto es porque el SL minimiza el trabajo repetido innecesariamente; es decir, permite a la industria no tener que inventar la rueda en cada proyecto. Históricamente, a medida que la industria tecnológica avanza, las innovaciones se consolidan como un monopolio sobre la plataforma tecnológica (el caso paradigmático es Microsoft) o en forma de estándares abiertos. Sin embargo, para una empresa, la dependencia que implica la inversión en una tecnología monopólica es un riesgo que aumenta con el tiempo. Por ejemplo: antes de Internet, varios proyectos de construir redes globales de computadoras fracasaron por confiar en protocolos propietarios que pronto se encontraron con problemas de interoperabilidad. Fueron necesarios estándares públicos (TCP/IP, HTTP, etc) para lograr la amplia distribución que tiene Internet hoy en día. Por eso, la enorme inversión económica en digitalización hace cada vez más necesario software de base de código abierto, y podemos observar cómo el SL tiene muchísimo más peso en las áreas de infraestructura informática (sistemas operativos, software de comunicación, servidores) que en las áreas más nuevas y cercanas al usuario (aplicaciones de escritorio, juegos, etc).

No resulta evidente suponer un juego de un único ganador entre los modelos de SL y software propietario. Incluso resulta difícil de creer en algún momento que el modelo de SL reemplace totalmente al modelo de propietario; sin embargo, es muy posible que siga ganando terreno en áreas donde se valore la inversión a largo plazo en una plataforma (consideración muy pertinente al sector público).

La recomendación de utilizar SL en el sector público se está haciendo sentir cada vez más en varios países. Los más avanzados al respecto son sin duda los países de la Unión Europea. El grupo IDA (*Interchange of Data between Administrations*), perteneciente a la Comisión Europea, bajo la dirección de Patrice-Emmanuel Schmitz preparó un informe preliminar sobre el uso de software Open Source en el sector público (IDA 2001). En él se analizan las principales ventajas que presenta el uso del software de código abierto, y numerosos ejemplos de utilización en instituciones públicas de Francia, Alemania, Bélgica y España, entre otros países. Este trabajo destaca cuatro puntos centrales sobre los cuales se justifica la utilización de SL en ámbitos públicos<sup>15</sup>: (i) el libre acceso del ciudadano a la información pública, (ii) la perennidad de los datos público, (iii) la seguridad de los ciudadanos y de Estado y (iv) Ventajas económicas y de promoción al desarrollo local.

En cuanto al primer punto cabe mencionar que el Estado es depositario de una gran cantidad de información provista por sus ciudadanos, quienes son dueños de dicha información. En un Estado democrático es necesario garantizar el acceso de la ciudadanía a estos datos, por lo que es imprescindible que el acceso no esté sujeto a ninguna restricción imponible por un proveedor único. Los formatos de datos abiertos, es decir, formas de almacenamiento que siguen normas y procedimientos públicamente disponibles garantizan que el Estado o cualquier entidad interesada pueda generar software capaz de leer, interpretar y procesar esta información. De esta forma, con formatos abiertos el Estado está en condiciones de garantizar el acceso a dicha información ya que no se encuentra sujeto a las restricciones de acceso que plantea un único oferente.

En segundo lugar, juntamente con la forma en que se almacena la información, es preciso que toda la estructura informática que utiliza el Estado sea inmune a las condiciones monopólicas de una empresa privada o a las vicisitudes que puedan afectarla. El mantenimiento de las muy variadas plataformas que utiliza el sector público no puede depender de lo que en el mercado se considere "obsoleto". Utilizando SL, el proveedor del software no es el único en condiciones de brindar mantenimiento.

En lo referente a las cuestiones de seguridad, a medida que avanza la digitalización de los procesos burocráticos, los alcances y responsabilidades del software que utiliza el Estado se vuelven cada vez más críticos. Esta criticidad impone la necesidad de que sea posible auditar el funcionamiento del software; esta auditoría sólo es posible si se dispone del código fuente del software utilizado. Existen numerosos ejemplos de software propietario

---

<sup>15</sup> Las ventajas del SL postuladas en este estudio coinciden en gran medida con los fundamentos de los proyectos de ley que se están proponiendo hoy en nuestro país y con la exposición del congresista peruano Edgardo Villanueva Núñez en respuesta a la carta de Microsoft Perú.

que, se descubrió, contenía accesos no documentados ("backdoors") que le permitían al proveedor (o a todo aquel que conociera el secreto) acceder ilegítimamente a los datos del cliente<sup>16</sup>.

Dos beneficios adicionales de la utilización del software libre son de índole económica: bajar el gasto en licencias y promover la industria local. Estos puntos son los más sujetos a debate. El primero se refiere a la reducción de los costos a partir de lo ya comentado referido a los beneficios económicos. En el segundo se refiere al desarrollo de la industria local a partir de las demandas del Estado de servicios asociados a la implementación y mantenimiento SL a empresas locales. En la actualidad las necesidades informáticas del Estado son satisfechas por empresas extranjeras, por desarrollos propios y en algunos casos por provisiones de empresas locales. A pesar de que actualmente se esté implementando plataformas abiertas en el Estado aún predominan las propietarias.

Por último, como ya hemos mencionado, el uso del SL está relacionado con la creación y circulación del conocimiento, y esto representa una ventaja adicional de su difusión. A diferencia del esquema propietario, permite generar y amplificar procesos de creación de conocimiento en tanto la tecnología no se presenta como un paquete cerrado y existen posibilidades de modificación, compatibilización, adaptación, y, por lo tanto, aprender de tecnologías ya desarrolladas y abiertas. Esto admite el desarrollo de capacidades por medio de ensayo y error, que se complementa a través de la pertenencia al grupo o comunidad de desarrolladores SL.

Estas auténticas comunidades epistémicas trabajan en red de modo que el proceso de creación no está limitado a una persona sino que pertenece al grupo. De tal manera, se inicia un proceso de aprendizaje colectivo dinamizado por el learning by doing –cuando el programador se enfrenta al código escrito por otras personas para modificarlo y adaptarlo a sus propias necesidades- y por el learning by interacting –cuando el programador se pone en contacto con otros programadores de la comunidad para participar de procesos de desarrollo conjuntos. La participación de foros de discusión que instrumenta la comunidad de desarrolladores permite compartir experiencias coordinar acciones conjuntas de desarrollo y de investigación.

Sin embargo esto no implica un ingreso irrestricto, de hecho se han identificado como motivaciones en la participación en proyectos de desarrollo libres, la construcción de un espacio cuasi-público donde el conocimiento tácito y un umbral de mínimo de capacidades funcionan como barreras a la entrada. En este sentido, el fenómeno puede ser descrito desde la noción de comunidades epistémicas en donde, por una parte, el conocimiento no es un bien público en el sentido que la exclusión es posible y por otra, la creación y circulación de conocimiento es intensa dentro de la comunidad pero opaca para los que permanecen fuera de ella. Al mismo tiempo, la noción de circulación de conocimiento al interior de la red no es una metáfora sino que implica un espacio concreto a través de Internet y el uso intensivo de las nuevas tecnologías. En este sentido,

---

<sup>16</sup> Los proyectos de la ciudad de La Plata y de la Ciudad de Buenos Aires citan, por ejemplo, el resonado caso de la clave NSA (National Security Agency, organismo de inteligencia estadounidense) descubierta casualmente en el sistema operativo Windows NT, cuyo propósito aun se desconoce.

ingresar a la comunidad implica participar de estos procesos de creación de conocimientos, y sus consecuencias son por un lado poder hacer uso del código ya desarrollado pero también aprender del mismo. Por lo tanto una última ventaja de la difusión del SL está relacionada con las posibilidades de aprendizaje, lo que es particularmente importante en países en desarrollo.

## **6 De qué depende la difusión de software libre. Características y límites de la difusión de software libre en la Argentina.**

En los apartados anteriores, apelando a las nociones de bienes club y comunidades epistémicas, se ha explicado porqué la difusión de SL no es automática sino que implica la decisión y capacidad de los agentes de ingresar a la comunidad (ya sea como productores o simples usuarios). De acuerdo con nuestra hipótesis, la posibilidad de ingreso depende de la capacidad de absorción, del nivel de conectividad de los agentes y la relación entre ambas dimensiones. A su vez, la presencia de economías de red, sponsoreo, costos crecientes de salida complican las decisiones de adopción incorporando dimensiones estratégicas. Vinculada con la elección de un sendero tecnológico que a priori no se puede afirmar que será el dominante en el futuro y en donde los costos de migración son elevados y crecientes.

A continuación, luego de realizar un análisis descriptivo de los datos de la encuesta, testaremos la hipótesis de relación entre capacidades tecnológicas, uso de redes y grado de difusión de SL. Para ello se utilizará la técnica estadística de análisis factorial.

### **6.1 Características de la difusión: un análisis descriptivo**

De los datos de la encuesta a 181 trabajadores informáticos surgen algunas conclusiones interesantes sobre el nivel actual de difusión de software libre y las características del mismo. Los rasgos fundamentales son: (i) el grado de difusión es elevado, sin embargo el uso que se hace es relativamente pobre; (ii) el desarrollo local de SL es muy limitado y está asociado a motivos muy personales vinculados en primera instancia con el desarrollo de capacidades y en segunda instancia con motivaciones políticas o ideológicas (aunque con gran heterogeneidad de respuestas); (iii) la decisión de utilización de software libre recae sobre las personas y no se trata de una decisión de la organización donde se desempeñan estos trabajadores, lo que resta sistematicidad al uso de SL; (iv) la difusión está asociada a las capacidades de los trabajadores, al dominio de determinados lenguajes y tecnologías y al grado de utilización de redes formales e informales.

#### **6.1.1 Descripción de la difusión del SL entre los trabajadores informáticos argentino**

Del total de encuestados el 75% hace algún uso de SL, aunque solo el 34% puede clasificarse como un uso medio o alto. La decisión de utilizar software libre recae sobre sí mismo en un alto porcentaje (30% de los que utilizan). En el 49% de los casos la decisión recae sobre la empresa/institución para la que trabajan basada en una decisión comercial en tan solo el 7% de los casos, en una decisión económica en el 20% y en una decisión técnica en el 22%.



Cabe destacar el bajo nivel de decisión de adopción del software libre por parte de demandas de los usuarios finales del software o como consecuencia de la estructura existente. Esto da la pauta de que es muy fuerte el uso de SL por motivos personales y cuando se trata una decisión de la organización ésta debe fundamentalmente a las ventajas económicas seguida por las ventajas técnicas. Algunas experiencias de adopción de SL por parte de empresas Argentinas se explica por esto mismo: trabajadores del área de sistemas que abogan por el uso de SL plantean los beneficios para la empresa posibles de su adopción, en algunos casos los desacuerdos con otras áreas (como compras) impiden la migración de lleve a cabo, en otros casos (generalmente donde prima el interés económico)

Esta característica permite justificar la utilización del trabajador informático como unidad de análisis del proceso de difusión de SL en la argentina.

Los motivos por los cuales utilizan SL son nuevamente, el bajo costo y entre los motivos técnicos destacan la seguridad, la estabilidad y la funcionalidad del SL.

El porcentaje de personas encuestadas que desarrollan software libre es 12,2%. y los motivos esta vinculados en primer lugar a desarrollar habilidades y compartir conocimiento y, en segunda instancia, porque consideran que no deberían recaer derechos de propiedad intelectual sobre el software. Sin embargo, el numero de respuestas es muy bajo dado el bajo porcentaje de personas encuestadas que desarrollan SL.

Por último cabe mencionar los motivos por lo cuales no se utiliza SL que la encuesta pudo identificar. Entre estos motivos sobresalen la falta de compatibilidad, los motivos políticos y el desconocimiento (21%, 21% y 15% de los que no utilizan, respectivamente)

## **6.2 Análisis de clusters relaciones entre capacidades y difusión**

Para analizar el grado de difusión de SL en función de las capacidades de los trabajadores informáticos y grado y forma de utilización de redes, se apeló al método estadístico llamado *análisis factorial*. Específicamente se utilizó la técnica de *análisis factorial de correspondencias múltiples* (AFCM) que permite agrupar a los individuos de acuerdo a patrones comunes en variables discretas o nominales. Con el AFCM es posible sintetizar las asociaciones existentes entre las diferentes modalidades de las variables que componen la matriz de datos y así obtener un conjunto limitado de clases compuestas por individuos que presentan una alta homogeneidad intragrupo y una elevada heterogeneidad extragrupo. Este método opera mediante la reducción de la dimensionalidad del fenómeno estudiado, conformando ejes factoriales cuya determinación permite concentrar el análisis en aquellas variables y modalidades que más aportan a explicar la problemática abordada, brindando, así, una visión manejable de la misma (Roitter, 1991 y Crivisqui, 1993).

A partir del análisis factorial, puede realizarse otro tipo de análisis conocido como *análisis de cluster* que agrupa los individuos con características similares. El análisis de Cluster construye grupos minimizándolas distancias euclidianas calculadas con las coordenadas de cada individuo en ejes factoriales que surgen del AFCM, este estudio

construyó los grupos con información correspondiente a los diez primeros ejes factoriales.

Para poder determinar la relación entre difusión de SL y las competencias tecnológicas, este estudio se agruparon los trabajadores informáticos en función de las herramientas de desarrollo que dominan. A partir de estos grupos fue posible identificar trayectorias tecnológicas sobre las cuales se analizó el uso de SL y de redes estos grupos. De tal forma se pudo establecer una relación entre la trayectoria tecnológica, el uso de redes y la difusión de SL lo que contribuyen a entender sobre qué bases se difunde el SL en la Argentina y cuáles son sus limitaciones.

En primer lugar, se analizó las competencias en función de las herramientas que estos trabajadores dominan. Esto permitió que los trabajadores fueran agrupados en cinco categorías en las que se puede identificar cierta trayectoria tecnológica y de aprendizaje en un *paradigma* específico de lenguajes y tecnologías. Este tipo de agrupación resulta central porque permite diferenciar las capacidades en función del *tipo* de herramientas que dominan. Luego se construyeron indicadores de difusión de SL y de uso de redes. A partir de estos indicadores y las trayectorias definidas en el cluster se probó la hipótesis de relación entre competencias, uso de redes y difusión de SL.

#### 6.2.1 Clasificación de acuerdo a las herramientas

Partiendo de la base de que las herramientas de trabajo juegan un rol central en la construcción de capacidades de los trabajadores<sup>17</sup>, se han construido cinco grupos de trabajadores en función de las herramientas (lenguajes y tecnologías) que dominan. Evidentemente el dominio de herramientas está lejos de explicar por sí mismo las competencias tecnológicas de los trabajadores informáticos. Yoguel et al 2004, advierte que las capacidades de estos trabajadores se construyen de una forma sistémica, donde el dominio de herramientas constituye una variable objetiva para medir estas capacidades, pero en su determinación intervienen además dimensiones como la educación formal, la formación en el trabajo, la utilización de redes, entre otros. No obstante, para este estudio se ha optado por agrupar a los trabajadores de acuerdo al dominio de herramientas en la medida en que a partir de ellas es posible identificar el tipo de sendero tecnológicos que han recorrido los trabajadores y cómo éste define sus capacidades.

En tal sentido, el conjunto de variables activas seleccionadas para la construcción de estos grupos son los lenguajes y tecnologías que dominan los trabajadores informáticos. De la lista de lenguajes y tecnologías sobre los que fueron consultados solamente no fueron considerados como variables activas aquellas con una tasa de respuesta inferior al 3%. De este análisis surgieron cinco grupos: (i) sin perfil específico; (ii) perfil

---

<sup>17</sup> Micheli Jordy (2003) señala las herramientas de trabajo utilizadas de modo efectivo y creativo, así como sus conocimientos y habilidades y la manipulación de datos, palabras, representaciones orales y visuales son más importantes que las credenciales acerca de su nivel y campo de estudios a la hora de estimar las capacidades de estos trabajadores.

Mainframes y minicomputadoras; (iii) perfil plataforma cerrada; (iv) perfil plataforma abierta y (v) perfil académico.

El primer grupo resultó ser el más numeroso (48% del panel) esta representado por un conjunto de personas sin un perfil definido en término del tipo de herramientas que domina. Básicamente se definió por el no dominio de las herramientas más complejas y novedosas del grupo de herramientas sobre las que se los interrogó. Este grupo esta constituido por el 48% de la personas entrevistadas y algunas de sus características relevantes son: (i) más de 20 años de experiencia laboral, lo que estaría indicando que se trata de un grupo con edades que entre 35 y 50 años; (ii) la rotación es relativamente baja (hasta un máximo de dos veces en cinco años) (iii) trabajan predominantemente en el sector público. En el grupo existe una leve predominancia del género femenino en comparación con la participación del genero en el total del panel. Cabe aclarar que lo integrantes de este grupo se caracterizan fundamentalmente por no haber seguido un sendero definido en la construcción de sus competencias, es decir, a diferencia de otros actores no ha logrado especializarse en un conjunto de herramientas comunes. No obstante, que la mayor parte de los entrevistados caiga en este grupo no es una debilidad del estudio, ya que la hipótesis que plantea es que determinado sendero tecnológico se asocia a un uso particular de SL. La ausencia de sendero definido en todo caso puede entenderse como la falta de especialización, lo que no implica que no puedan ser asociados a más de una de las trayectorias que serán definidas más abajo o que estén actualmente construyendo su propio sendero.

El segundo grupo constituido por el 12% del panel presentó especial afinidad por las herramientas tales como Cobol, Cliper y Xbase entre los lenguajes y AS 400 y mainframes entre las tecnologías. Por ello fue definido como el grupo Mainframes y minicomputadoras. Este grupo también se caracteriza por ser el de mayor experiencia laboral y mayor edad, mientras que la rotación laboral es significativamente menor que el promedio para el panel. Trabajan mayoritariamente en organizaciones cuya actividad principal no es la provisión de SSI, perteneciendo al sector financiero o Estado y trabajan en relación de dependencia. En cuanto su formación, se caracteriza por haberla adquirido por afuera del sistema educativo formal, predominan los cursos de formación y la formación en el trabajo.

Al tercer grupo se lo denominó plataforma cerradas o propietarias por las características de las herramientas que dominaban y significan el 15% del total. Entre los lenguajes pueden mencionarse por caracterizar al grupo son Visual Basic y ASP, mientras que entre las tecnologías son Punto Net, COM, DCOM, MTS. Este conjunto de herramientas tienen la particularidad de ser tecnologías propietarias y monoplatafora, ya que corren solo en Microsoft Windows. En cuanto a otras características del grupo pueden mencionarse que se trata de personas entre 20 y 30 años de edad y con menos de 10 años de experiencia laboral. La rotación laboral para el grupo es relativamente elevada. Resulta interesante señalar que le grupo no se caracteriza por el sector para el que trabaja. Las personas que componen el grupo se desempeña tanto en empresas cuya actividad principal es la informática como en otro tipo de organizaciones. Tampoco se detecta un patrón sectorial en la demanda, lo que estaría indicando la transversalidad de esta clase de herramientas. En cuanto a su educación formal predominan las licenciaturas en sistemas. El grupo se caracteriza por considerar a los cursos de capacitación como una de las fuentes

principales en su formación profesional, al tiempo que se destaca, a diferencia del resto, por haber obtenido certificaciones<sup>18</sup>.

El cuarto grupo que se caracteriza por la utilización de herramientas asociadas a las plataformas abiertas, por lo que se le dio ese nombre, representa el 15% del panel. Entre los lenguajes que dominan los miembros de este grupo se destacan: C, C++, Java, Unixshell, Python y Perl. Y entre las tecnologías el grupo se destacó por dominar programación multitier, plataforma web, XML, todas ellas relacionadas con tecnologías de comunicación del tipo intranet/internet. Y UML, Programación orientada a objetos y Extreme Programming, todas ellas relacionadas con el paradigma de objetos, relacionado con las tecnologías de comunicación. Una cualidad fundamental de este tipo de herramientas es ser relativamente nuevas. Otras características del grupo son la experiencia laboral menor a 20 años, con edades entre 20 y 30, y la muy elevada rotación (tres veces en cinco años), lo que está asociado al dinamismo asociado con estas tecnologías. Asimismo, trabajan mayoritariamente en empresas cuya actividad principal es la informática. En cuanto a la educación formal, el grupo se caracteriza por la formación universitaria completa en la carrera de licenciatura en ciencias de la computación. No obstante, este grupo se define porque sus miembros consideran que sus capacidades las obtuvieron de forma autodidacta.

El último grupo (9% del total del panel) se destacó por el dominio de herramientas de mayor interés académico que de mercado. Incluso, de la encuesta surge que no hace uso de ellas en su trabajo actual. Esto indicaría que estos trabajadores han desarrollado capacidades que si bien no aplican directamente a su trabajo amplían su competencias. Los lenguajes que dominan por lo que se caracteriza el grupo son Lisp, Smalltalk, C, C++, Python, Assembler, Haskell, Perl. Entre otras características del grupo pueden mencionarse la experiencia laboral menor a 10 años, la rotación laboral elevada (dos veces en cinco años). Este grupo se caracteriza porque sus integrantes se desempeñan mayoritariamente en instituciones académicas. Asimismo destaca la presencia de personas con estudios de postgrado en informática, así como los licenciados en ciencias de la computación. A pesar del elevado nivel de educación formal, los miembros de este grupo, al igual que en el caso anterior, se caracterizan por considerar que sus capacidades las obtuvieron de forma autodidacta.

### 6.2.2 Relación entre difusión de SL uso de redes y competencias

Una vez definidas las trayectorias tecnológicas se analizó la relación entre éstas y el grado de utilización de SL. En este caso se detectó relación entre ambas variables. Más allá del grado de utilización también se estudió el tipo uso que cada grupo hace del SL, es decir, sobre quién recae la decisión de utilización de SL, cuales son las razones por las que eligen SL, si además de la utilización realiza desarrollos de SL y sobre qué motivaciones y entre quienes no lo utilizan por qué eligen plataformas propietarias. De

---

<sup>18</sup> En informática, grandes empresas tales como Microsoft, IBM, Oracle extienden títulos de certificación a partir del dictado de cursos sobre las herramientas y tecnologías que estas mismas empresas desarrollan y comercializan.

esta forma se logró una descripción más completa del tipo de difusión que se logró en cada caso.

### Relación entre cluster de herramientas y el grado de utilización de SL

Cluster de acuerdo a las herramientas que dominan	Grado de utilización de SL				
	Ninguno	Alguno	Bajo	Medio	Alto
Sin perfil específico	34,1%*	17,00%	23,90%	12,50%	12,50%
Mainframes y minicomputadoras	36,40%*	9,10%	36,40%	13,60%	4,5%*
Plataformas cerradas	22,20%	14,80%	44,4%*	14,80%	3,7%*
Plataformas abiertas	3,6%***	17,90%	14,30%	21,40%	42,9%***
Herramientas de interés académico	6,3%*	18,80%	18,80%	25,00%	31,3%*
Total	25,40%	16,00%	26,50%	15,50%	16,60%

Fuente. Elaboración propia en base Encuesta SADIO-UNGS sobre las competencias tecnológicas de los trabajadores informáticos. Nota: la probabilidad de aceptar la hipótesis nula de ausencia de asociación es 0,001%; \* Prueba Z significativa al 10%; \*\*\* Prueba Z significativa al 1%.

El primer grupo se caracteriza por no utilizar SL, es decir el porcentaje de individuos del grupo que no utiliza SL es significativamente superior al promedio de no-utilización del panel. A pesar de esto, como puede apreciarse en el cuadro previo un porcentaje elevado del grupo utiliza SL con diferente grado, en estos casos la decisión de utilización recae sobre la organización para la que se desempeña, más que sobre una decisión personal, bajo un criterio de reducción de costos. Entre los que no utilizan SL el motivo es fundamentalmente por política de la empresa, aunque también señalaron la falta de compatibilidad así como la falta de conocimiento de soluciones SL que pudieran aplicar a su tarea. Por último, el grupo también se destaca por no desarrollar SL.

En el segundo grupo también se caracterizó por la no-utilización de SL, los motivos fueron por política de la empresa y por la falta de compatibilidad. Entre los miembros del grupo que utilizan SL se destacaron los motivos referidos al bajo costo y la estabilidad. Este último es coherente con su trayectoria tecnológica ya que por trabajar en ambientes de alta disponibilidad otorgan una gran valoración a estabilidad que resulta fundamental. En estos casos la decisión de utilización de SL recae sobre la empresa para la que trabaja bajo criterios económicos (menores costos) pero también técnicos. Este grupo también se caracterizó por no desarrollar SL.

En lo que respecta al tercer grupo, se caracteriza por una utilización de SL baja, más que nula a diferencia de los casos anteriores. Entre las razones que condujeron a la elección de SL destacan el bajo costo y la seguridad.

En el cuarto grupo la utilización de SL es elevada. El uso de SL en el ambiente laboral responde a una decisión técnica por parte de la empresa donde trabaja, aunque también la decisión recae directamente sobre los trabajadores. Acorde con esto último, los motivos centrales de utilización de SL son la funcionalidad y la disponibilidad del código fuente. Se puede caracterizar a este grupo por su vocación por la informática<sup>19</sup> y las nuevas

<sup>19</sup> El tipo de carrera que siguieron mayoritariamente (Ciencias de la computación) tiene una aproximación a la informática desde una perspectiva científica y académica. Otras carreras ofrecen un acercamiento a la informática desde los negocios.

tecnologías. El acceso a internet les ha permitido desarrollarse profesionalmente de ahí su espíritu autodidacta. El SL se les presenta como una nueva tecnología completamente abierta con la que pueden desarrollar nuevas capacidades: la disponibilidad del código fuente se vuelve para este grupo una cualidad sumamente atractiva. Este grupo se caracteriza y se diferencia de los anteriores en que un porcentaje significativamente elevado desarrolla SL además de utilizarlo. Las motivaciones para ello ésta relacionadas con compartir conocimiento y desarrollar habilidades. Lo que una vez más es coherente con su espíritu autodidacta.

En el último grupo grado de utilización de SL es también elevado. No obstante se diferencia del anterior en que en este caso, el uso de SL en el ambiente laboral es una decisión personal ya que en organización donde se desempeña no hay requerimientos específicos al respecto. También se diferencia en que los motivos centrales de utilización de SL se destacan las razones políticas así como el deseo de participación de la comunidad. Al igual que el grupo anterior se caracteriza por desarrollar SL, entre las motivaciones, los integrantes de este grupo mencionan las posibilidades de compartir conocimiento y desarrollar habilidades, pero también participar en la escena del SL. De tal forma, una diferencia central entre este grupo y el anterior es que le otorga a la problemática del SL una dimensión política mientras que el grupo cuatro se limita al problema técnico.

### **Relación entre cluster de herramientas y desarrollo de SL**

Cluster de a cuerdo a las herramientas que dominan	Desarrolla software libre	
	Si	No
Sin perfil específico	10,50%	89,50%
Mainframes y minicomputadoras	0%**	100,00%
Plataformas cerradas	9,50%	90,50%
Plataformas abiertas	29,6%*	70,40%
Herramientas de interés académico	40%***	60,00%
Total	16,40%	83,60%

Fuente. Elaboración propia en base Encuesta SADIO-UNGS sobre las competencias tecnológicas de los trabajadores informáticos. Nota: la probabilidad de aceptar la hipótesis nula de ausencia de asociación es 0,06%; \* Prueba Z significativa al 10%; \*\*\* Prueba Z significativa al 1%.

En lo que se refiere a la relación entre difusión de SL y utilización de redes también se detectó relación. Tal como puede apreciarse en la tabla siguiente, a menor grado de utilización de redes menor utilización de SL. La utilización de redes virtuales se evaluó a partir del uso de herramientas de comunicación y pertenencia a comunidades a través de Internet. La relación positiva no parece sorprendente si se sabe que Internet es el principal medio de difusión del SL.

### **Relación entre el grado de utilización de redes virtuales y de SL**

Grado de utilización de redes virtuales	Grado de utilización de SL				
	Ninguno	Alguno	Bajo	Medio	Alto
Muy Bajo	46,2%*	0,00%	23,10%	23,10%	7,70%
Bajo	22,10%	27,9%***	29,40%	8,80%	11,80%
Medio	32,10%	10,70%	26,80%	12,50%	17,90%
Alto	13,90%	11,10%	25,00%	27,8%**	22,20%
Muy Alto	25,00%	0,00%	12,50%	25,00%	37,5%*
Total	25,40%	16,00%	26,50%	15,50%	16,60%

Fuente. Elaboración propia en base Encuesta SADIO-UNGS sobre las competencias tecnológicas de los trabajadores informáticos. Nota: la probabilidad de aceptar la hipótesis nula de ausencia de asociación es 0,029%; \* Prueba Z significativa al 10%; \*\* Prueba Z significativa al 5%; \*\*\* Prueba Z significativa al 1%.

## 7 Conclusiones

La conclusión fuerte de este trabajo es la relación entre la creación de competencias y el uso y desarrollo de SL. Esta relación puede y debe ser entendida como una relación bi-direccional. En este sentido, la difusión de SL genera competencias en los trabajadores informáticos sin embargo en necesario un umbral mínimo para que el proceso se inicie.

De acuerdo con los datos empíricos analizados, el uso de SL está asociado determinadas trayectorias tecnológicas definidas por el tipo de herramientas que los trabajadores informáticos dominan. Eso indica que

Una dinámica similar puede describirse en el uso de redes virtuales. Tanto el desarrollo como la utilización de SL conduce hacia una mayor y más compleja utilización de las redes virtuales; y a su vez esta mayor participación contribuya a la generación de competencias, no obstante, en este caso también es necesario un umbral mínimo inicial en la utilización de redes y en las competencias para que este proceso de creación y circulación de conocimiento se inicie. En este sentido cobra un lugar central entender al conocimiento que circula por estas redes como un bien cuasi-público.

En este sentido las opciones de política para profundizar la difusión del SL, que se justifica en la creación de competencias, resultan por completo diferentes a las que se apela habitualmente desde las organizaciones de promueven el uso y difusión. Estas políticas generalmente son del tipo top-down. Es decir, asumen que para lograr una mayor difusión del SL tienen que imponerse estándares y su uso en ámbitos públicos. El análisis de este trabajo conduce a pensar que este tipo de políticas podría encontrar serias limitaciones en su aplicación debido a la falta de competencias o la falta de compatibilidad entre las trayectorias de las capacidades tecnológicas de los trabajadores informáticos y la trayectoria tecnológica propia del SL.

Más allá de la pertinencia de opciones de políticas del tipo top-down, de este estudio se desprende la necesidad de complementarlas con políticas del tipo bottom-up. Este tipo de políticas debería apuntar a la generación de competencias compatibles con las tecnologías libres así como incentivar al uso de redes y herramientas de comunicación. En este sentido, el objetivo final sería contribuir a la creación de las condiciones para que el grado de adopción de SL sea mayor, pero no ya por una imposición vertical sino por las propias decisiones de los agentes.

## Bibliografía

Arthur, W. B. (1988), "Competing Technologies: An Overview" En Dosi G, C. Freeman R. Nelson, G. Silverberg and L. Soete (eds.), Technical Change and Economic Theory, Pinter, London

Arthur, W. B. (1989) "Competing Technologies, Increasing Returns and Lock-in by Historical Events" *Economic Journal*, 99, 116-131, 1989.

Baldwin C.Y. y Clark K.B., (2002) *Empirics of the Bazaar: Discussion June*. Presentado en la conferencia

Benkler, Yochai (2002) *Coase's Penguin, or, Linux and the Nature of the Firm*. *Yale Law Journal*, 112, Winter 2002–2003.

Bessen, James (2002) *Open Source Software: Free Provision of Complex Public Goods*. *Research on Innovation*

Covarrubias, Isaías M. (2002) *Enfoque Sistémico e Individualismo Metodológico: una aproximación* Universidad Centro Occidental Lisandro Alvarado Barquisimeto. Venezuela.

Chudnovsky, Daniel, López, Andrés y Melitsko, Silvana *El sector de software y servicios informáticos (SSI) en la Argentina: Situación actual y perspectivas de desarrollo* CENIT DT 27/Julio de 2001.

Cowan, Robin, Jonardzand, Nicolas y Zimmermannx, Jean-Benoit (2004) *On the Creation of Networks and Knowledge* MERIT INFOMOMICS 2004-010.

Crivisqui, 1993

Dalle, Jean-Michel y David, Paul M. (2003) *The Allocation of Software Development Resources In 'Open Source' Production Mode* SIEPR Discussion Paper No. 02-27 March 2003

Dalle, J.M. y Jullien N. (2001) *Open-Source Vs. Proprietary Software*. Mimeo <http://opensource.mit.edu/papers>

Dalle, Jean-Michel y Jullien, Nicolas (2003) *Turning Fads Into institution*. *Policy Research* 32 (2003)

David, P. A. and Foray, D., (1994), *Percolation Structures, Markov Random Fields and the Economics of EDI Standard Diffusion*, en Global Telecommunications Strategies and Technological Changes Pogorel (Editor). North-Holland, Amsterdam.

David, Paul & Foray, Dominique (2002). *Economic Fundamentals of the knowledge society*. SIEPR Discussion Paper no. 01-14.

Dewatripont, M. Jewitt, I. y Tirole, J. (1999) *The Economics of Career Concerns, Part I: Comparing Information Structure*. *Review of Economic Studies*. Vol. 66 1 (January)

Frisch H.L., Hammersley J.M., (1963), *Percolation Process and Related Topics*, *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, n° 4.

FLOSS (2002) *Free/Libre and Open Source Software: Survey and Study FLOSS: FINAL REPORT* International Institute of Infonomics University of Maastricht, The Netherlands Berlecon Research GMBH Berlin, Germany Junio 2002

Hann, et al (2002) *Delayed Returns to Open Source Participation: An Empirical Analysis of the Apache HTTP Server Project* Mimeo para la conferencia de Toulouse

Hawkings (2002) *The economics of open source for a competitive firm. Why give it away for free?* <http://opensource.mit.edu/papers>



IDA (2001) (Interchange of Data between Administrations, European Commission, DG Enterprise) Study into the use of Open Source Software in the Public Sector Part 2 Use of Open Source in Europe June 2001.

Katz, J. y Hilbert, M. (2003) Los caminos hacia una sociedad de la información en América Latina y el Caribe. Cepal Alfaomega. Bogotá.

Katz y Shapiro (1968) Technology Adoption in Presence of Networks externalities Journal of Politics Economics 94 822-41 1986.

Laville, Frédéric (1999) Les voies de la rationalité procédurale. Artículo basado en la Tesis doctoral titulada Les voies de la rationalité procédurale. de l'École des Hautes Études en Sciences Sociales. Paris Enero 1999

Lecocq, X. y Demil (2002) Open standard : role of externalities and impact on the industry structure. MIT SL PAPER

Lerner, Josh y Tirole, Jean (2000) The Simple Economics Of Open Source NBER Working Paper Series, Working Paper 7600 <http://www.nber.org/papers/w7600> Marzo 2000.

Metcalfe (1989) The diffusion of innovation: an Interpretative Survey. En Dosi G, C. Freeman R. Nelson, G. Silverberg and L. Soete (eds.), Technical Change and Economic Theory, Pinter, London

Micheli, Jordy (2003), El trabajo de digitofactura en la economía postindustrial en Espacios globales: espacios del capitalismo, Ed. Carmen Bueno (en prensa). México.

Nemirovsky y Yoguel (2001), Dynamics of High-Technology Firms in the Silicon Valley, Working paper No 01-03 in DRUID Aalborg University, Department of Business Studies

Opensource.org 2004 <http://www.opensource.com>

Pal, Nilendu y T.R. Madanmohan (2002) Competing on Open Source: Strategies and Practise. <http://opensource.mit.edu/papers>

Roitter, 1991

Rossi C y Bonaccorssi, (2004) Comparing Motivations of individual programmers and firms to take part in the open source movement. From community to business. Sant' Anna School of Advanced Studies. Pissa, Italia.

Raymond, Eric S. (1999a) The Cathedral and the Bazaar. En The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, O'Reilly & Associates, Inc., Sebastopol, CA.

Raymond, Eric S. (1999b) Homesteading the Noosphere. En The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, O'Reilly & Associates, Inc., Sebastopol, CA.

Raymond, Eric S. (1999c) Brief History of Hackedom. En The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, O'Reilly & Associates, Inc., Sebastopol, CA.

Stallman R. (1983) FSF Manifiesto. [www.fsf.org](http://www.fsf.org)

Valduriez 2002 Business models for OSS. Presentado en la Conferencia de Toulouse.

Villanueva Núñez 2002 Correspondencia entre el Señor J. A. González, Gerente General de Microsoft del Perú y E. D. Villanueva Núñez, congresista peruano. <http://www.gnu.org.pe/rescon.html>

Von Bertalanffy, L. (1987) Historia y situación de la teoría general de sistemas. Ed. Alianza Universidad. Madrid.

Von Hippel, Eric (2002) Open source software projects as user innovation networks MIT Sloan School of Management

Yoguel, G.; Borello, J.; Erbes, A.; Robert, V.; Roitter, S. (2004) Competencias tecnológicas de los trabajadores informáticos argentinos. Más allá de las restricciones de demanda y oferta. Littec e-papers.

Yoguel, G., Novick M. y Fuchs, M. (2003) Desarrollo de Redes de Conocimiento. El Caso de Lifa, Laboratorio de Investigación y Formación en Informática Avanzada de la Universidad Nacional de la Plata. En Nuevas Tecnologías de Información y Comunicación. Los Límites en la Economía del Conocimiento. F. Boscherini, M. Novick y G. Yoguel (Comps.) Miño Dávila 2003.

Yoguel G, Novick M, Milesi D, Roitter S y Borello J (2003), Información y conocimiento: la difusión de TICs en la industria manufacturera argentina Información y conocimiento: la difusión de TICs en la industria manufacturera argentina, Revista de la CEPAL, Santiago de Chile, de próxima aparición.

Zeitlyn, David (2003) Gift economies in the development of open source software: anthropological reflections. Policy Research N° 32.