

**Grupo 8: Descentralización productiva y tercerización:
sus efectos sobre los procesos y relaciones de trabajo.**

Los procesos productivos de software en la Ciudad de Buenos Aires: Una tipología exhaustiva.

Zukerfeld, Mariano

marianozukerfeld@gmail.com

Yansen, Guillermina

guiyaestaba@yahoo.com.ar

Segura, Agustín

segura.a@gmail.com

UBA-PRI R10-279¹.

Introducción

En los últimos años el llamado “sector del software y servicios informáticos” (SSI) ha sido nombrado reiteradamente a la hora de señalar prioridades para las políticas públicas en la Argentina. Así, por ejemplo, en el año 2004 se sancionaron las leyes 25.856 y 25.922, dando status industrial a la actividad, generando beneficios impositivos y crediticios y creando un fondo para financiar I+D en el rubro (Fonsoft). Sin embargo, las indagaciones respecto de las principales variables económicas relativas al sector no resultan aún satisfactorias. Por caso, las estimaciones respecto de la cantidad de unidades productivas y trabajadores incluidos en ellas no son coincidentes ni claras. En parte, ello obedece a que los altos niveles de descentralización y tercerización, la enorme heterogeneidad (incluyendo modalidades ajenas a los mecanismos mercantiles) y la considerable informalidad de los procesos productivos de software han conspirado en contra de su tipologización sistemática y exhaustiva. Más aún, en la literatura específica la producción de software en grandes empresas (que es la

¹ El presente trabajo forma parte de una investigación más amplia titulada y enmarcada por el Proyecto de Reconocimiento Institucional de la Facultad de Ciencias Sociales de la UBA, R10-279. Los integrantes del equipo de investigación son Lucila Dughera, Hugo Ferpozzi, Nahuel Mura, Agustín Segura, Guillermina Yansen y Mariano Zukerfeld. Más información sobre esta investigación puede encontrarse en <http://trabajoinformacional.wordpress.com/>

única que se ha medido de manera consistente) se asume como equivalente a la totalidad del universo de la producción de software, que es, a todas luces, mucho más amplio.

Este trabajo intenta, consecuentemente i) proponer una categorización amplia que ha sido respaldada empíricamente¹ y ii) bosquejar algunas de las modalidades productivas menos conocidas. El principal resultado de esta ponencia es el de la tipologización que se explica a continuación y que estructura las secciones de la misma. Tal tipologización puede ser de utilidad al menos i) para futuros estudios cuantitativos que busquen mensurar cada uno de los estratos ii) para distinguir qué rasgos caracterizan a los procesos productivos de software en general y cuáles son sólo válidos para una modalidad específica iii) para elaborar políticas públicas que atiendan a las particularidades de cada uno de los procesos productivos de software enumerados .

Esta ponencia está estructurada como sigue. En la primera sección presentamos nuestra propuesta de un esquema exhaustivo y sistemático de los procesos productivos de software. Las secciones siguientes utilizan nuestro material empírico para ejemplificar esos distintos procesos. Así, la segunda sección contiene una aproximación a cuatro Procesos Productivos de software *mercantiles*, ordenados en función de su tamaño: a) Microempresas b) PYMES c) Empresas Grandes d) Empresas Multinacionales. La tercera sección, en cambio, describe cuatro procesos productivos de software *no mercantiles*: e) Producción No Laboral (Software Libre y Producción Colaborativa) f) Producción Estatal g) Producción Académica h) Producción en organizaciones sin fines de lucro. Finalmente, presentamos nuestras conclusiones.

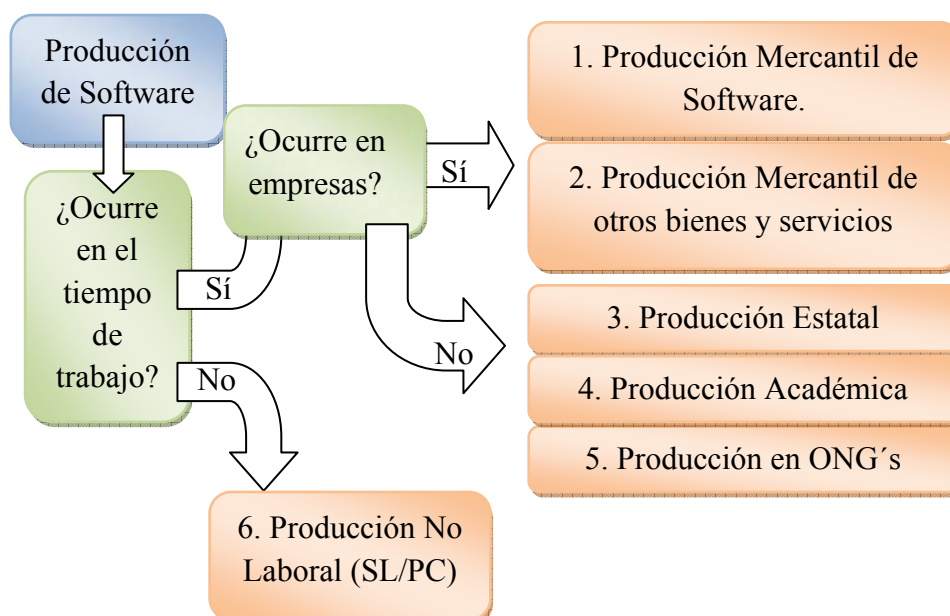
I.

Los tipos de Procesos Productivos de Software: una propuesta.

Partiendo de lo anterior podemos guiarnos por una serie de preguntas para circunscribir los distintos procesos productivos. La primera de ellas es la relativa a si el software se produce como parte de una actividad laboral o no. En el segundo caso, estamos ante lo que llamamos ***Producción No Laboral***. En la práctica este tipo de producción se asemeja a la de Software Libre o a la producción Colaborativa SL/PC. En efecto, casi toda la producción de este tipo es de software no privativo y se realiza en redes de prosumidores que tienen una serie de rasgos específicos (Vidal, 2000;

Bauwens, 2006). Sin embargo, debe quedar de manifiesto que el criterio definitorio es que la producción ocurra por fuera de la jornada de trabajoⁱⁱ. Por el contrario, si la producción de software ocurre como parte del trabajo, estamos ante alguna de las múltiples formas de la Producción Laboral. Dentro de ella cabe preguntarse si se trata de un proceso productivo mercantil o no. Esto es, ¿ocurre en una empresa o en otra clase de organismo? Si no es un proceso mercantil, estamos ante tres posibilidades: **Producción estatal** que, previsiblemente, **Producción académica**, que apunta al software elaborado en los ámbitos universitarios: desde aquél elaborado como parte de procesos de investigación hasta el que sirve para gestionar la actividad académica; **Producción en ONG's** que refiere a la producción de software dentro de las organizaciones sin fines de lucro. Si, por el contrario, se trata de una producción que mercantil tenemos dos opciones. La primera es que estemos ante la **Producción mercantil de otros bienes y servicios**. Esta categoría refiere a la producción de software in house, esto es, a la que ocurre en firmas cuya principal fuente de ingresos proviene de la venta de bienes y servicios distintos del software y los servicios informáticos. Finalmente, la categoría más importante es la de las **Producción Mercantil de software**: la de las empresas específicamente dedicadas a la elaboración, implementación, mantenimiento, adaptación y soporte técnico de software y servicios informáticos. Esta categoría, naturalmente, merece ser desagregada.

Gráfico nro. I
Seis tipos de Procesos Productivos de Software



Fuente: Elaboración propia

Optamos por lidiar con la enorme heterogeneidad de esta categoría descomponiéndola del siguiente modo: primero dividimos entre los procesos productivos que son autónomos y los que son heterónomos. En los **Autónomos**, las decisiones respecto de qué y cómo producir se toman en la unidad productiva misma. Los separamos en función de la cantidad de empleados (y ayudándonos en algún caso con el nivel de facturación). Hay que señalar, siguiendo al OEDE y otras instituciones, que los cortes entre distintos tamaños varían en cada actividad, y que en los procesos productivos de software empresas que parecerían minúsculas en otras ramas pueden ser medianas o grandes. Así, consideramos usualmente como *Microempresas* a las que ven involucrados en sus procesos productivos hasta cuatro miembros. *Pequeñas y medianas* empresas serán aquellas que tienen, y aquí nos separamos del criterio del OEDE, menos de 40 empleados. En nuestro trabajo concreto, la distinción entre pequeñas y medianas no resultó significativa. Consideramos *Grandes* a aquellas unidades productivas que cuentan con una cantidad mayor de empleados. Naturalmente, es difícil establecer la barrera divisoria entre empresas medianas y grandes de manera abstracta. Pero, otra vez,

es relativamente simple hacerlo respecto de los casos concretos que encontramos en el campo, y en nuestra investigación las empresas que rondaban las cuatro decenas de miembros se nos presentaron como radicalmente distintas de las que contaban con una quincena. Puede objetarse que esta división no contempla a las empresas radicalmente mayores, y mezcla bajo la denominación de grandes a una modesta empresa local con un pulpo internacional que puede contar con miles de empleados. La respuesta a esta objeción, en nuestra opinión, viene dada por la otra categoría utilizada: la de los procesos productivos *Heterónomos*. En ellos las decisiones respecto de qué y cómo producir están supeditadas a otras unidades productivas. El caso más evidente es cuando esa decisión recae en una casa matriz. Allí estamos frente a una *Filial de Multinacional*). Una segunda posibilidad es que las decisiones se tomen de manera razonablemente horizontal entre un conjunto de unidades productivas que han establecido, en condiciones de relativa igualdad, vínculos para elaborar determinado producto. A esa modalidad la denominamos en *Producción en red*. Una última posibilidad es que, de manera regular, la unidad productiva en cuestión, tenga un demandante y decisor externo pero casi exclusivo de su producción. A esta situación de tercerización permanente le llamamos *Producción Escalonada*. Este esquema es, ciertamente, perfectible. Pero introducir algunas precisiones obligaría a complejizarlo más allá de los niveles que resultan prácticos para este artículoⁱⁱⁱ.

Con todo, aunque nuestro trabajo de campo cubrió todas las modalidades señaladas, por motivos de espacio aquí sólo presentaremos ocho de ellas. Por un lado, las mercantiles. En la sección siguiente se ejemplifican los tres tipos de procesos autónomos y el principal de los heterónomos, el de las empresas multinacionales. A continuación se analizan los procesos no mercantiles. Partiendo de la producción que ocurre en el tiempo no laboral, se sigue por la que ocurre en el estado, la academia y finalmente, las ONG. Quedan excluidos, entonces, los procesos mercantiles in House, cosa que constituye una seria limitación de este trabajo-, la producción escalonada y la producción en red. Futuras ponencias saldarán esta deuda.

II

Producción Mercantil de Software:

a) Microempresas:

Las microempresas son un subsector que, en función de los casos ocurridos en nuestra muestra intencional, resultó sumamente interesante. Hay dos rasgos homogéneos respecto del *origen* de estas unidades productivas: i) en todos los casos surgen cuando sus miembros se independizan por propia voluntad de unidades productivas mayores. Esto es, a diferencia de otras actividades, aquí en ningún caso emergen con posterioridad a un despido o a la quiebra de la unidad productiva previa. Están motivadas por una búsqueda de autonomía en el manejo de los tiempos y las decisiones respecto de la producción. En menor medida, aparece como una razón la desintermediación en términos económicos. ii) también, en todos los casos relevados, hay un flujo de demanda previo a la creación de la empresa, demanda efectiva que justifica el salto de los emprendedores. Esto es, no se trata de emprendedores que se lanzan a una inversión de riesgo, a una aventura incierta. Sino de exempleados que conservan vínculos con uno o más clientes de sus empleos anteriores y que montan sus emprendimientos sobre la base de esa seguridad económica. A su vez, dado que la inversión necesaria (algunas computadoras que ya se poseían, una página web que ellos saben desarrollar) es poco costosa, las barreras a la entrada en el mercado son muy bajas.

Más allá del impulso inicial, también encontramos resultados homogéneos respecto de la *demand*a posterior. Todos tienen un requerimiento de sus productos y servicios igual o mayor al que pueden brindar. Ahora bien ¿qué hacen estas empresas? ¿qué tipo de productos y servicios ofrecen? Unánimemente, predomina el desarrollo de *herramientas web*. Desde aplicaciones elementales a complejas plataformas e incluso hosting, pero siempre a través de Internet.

La situación de no tener altos *costos* que afrontar coloca a los microempresarios en una situación que les resulta, en todos los casos, ampliamente satisfactoria. De manera más particular, el costo más grande que afrontan algunas de estas unidades productivas es el del alquiler de sus oficinas y los servicios concomitantes, que en ningún caso superaba los \$2000 por mes. Esto merece dos comentarios importantes: i) estas empresas no pagan licencias de *propiedad intelectual* de ninguna índole. Reurren, de manera complementaria, al software libre y al software privativo ilegal en función de sus trayectorias 'path dependence': utilizan los lenguajes, aplicaciones y sistemas

operativos que por algún motivo aprendieron a manejar, y no los que eligieron racionalmente en función de cierta perspectiva filosófica o de un análisis económico. Es decir, en ningún caso hubo una reflexión previa a la adopción. En cambio, con posterioridad a haber tenido la obligación de, por caso, programar en php en un empleo anterior, si emergen los discursos favorables al software libre. ii) estas empresas no tienen trabajadores registrados y, por ende, no asumen cargas previsionales ni de ningún otro orden. En las unidades productivas con mayor nivel de ingresos de este segmento, se empieza a discutir la necesidad de “blanquear” la situación impositiva y laboral, pero la ecuación de costos y beneficios no resulta del todo convincente para los emprendedores.

La ausencia de registro laboral también viene dada por el hecho de en estas empresas predomina un esquema organizativo peculiar: *los socios son más que los empleados*. Vimos, reiteradamente, esquemas de tres o cuatro socios que contratan a un empleado o dos, que ofician de auxiliares, pero nunca de técnicos especializados. Por supuesto, en otros casos no hay empleados, sino sólo socios. Esto es indisociable del origen afectivo del vínculo entre los dueños de la empresa: compañeros de trabajo, amigos de un amigo, etc., pero en ningún caso media una mutua selección profesional. En cambio, la racionalización sí aparece para los empleados que se toman posteriormente.

Aunque podría suponerse que en estas unidades la división del trabajo es nula, esto no suele ser así. En procesos productivos muy elementales, una división típica es la que separa a quien se ocupa del diseño visual de las páginas frente a quien(es) se concentran en la programación misma. En casos algo más complejos, emerge la división entre, por ejemplo, un encargado de la parte comercial, uno o dos enfocados en el diseño del producto y otros concentrados en la codificación. A diferencia de los temas de propiedad intelectual, las decisiones relativas a la división del trabajo se toman de manera conciente y planificada.

El *ingreso* de estas empresas rondaba (hacia mediados de 2010) entre los USD 3000 y los USD 60.000 anuales per cápita. Nótese que si la primera cifra se compara con los valores de ventas por trabajador de las grandes empresas -tal cual los distingue el OPSSI (2010:10)-, vemos que algunas microempresas tienen niveles de productividad

muy superiores a los más altos de las firmas locales y multinacionales-. Por supuesto, ese punto merece mayor discusión, pero nuestros datos indican con claridad que *no puede asumirse inmediatamente que la cantidad de riqueza por empleado que generan las empresas de mayor tamaño sea superior a la de las microempresas.*

Es interesante, sin embargo, que ninguna de estas empresas tiene una política de precios racionalizada. Aunque la dificultad de estimar las horas que demandará un trabajo es un problema para todos los PPSW, en este caso las variaciones son mucho mayores. Por supuesto, los márgenes de rentabilidad varían según el tipo de cliente y, especialmente, cuando se trata de exportaciones.

Esto nos lleva al hecho, quizás llamativo, de que casi todas estas empresas exportan alguna proporción de sus productos y servicios, aunque, claro, esta proporción es sumamente variable. Más aún, también es usual que se produzca la subcontratación de recursos humanos y bienes de capital situados en el extranjero. No sólo encontramos la sub-contratación de programadores de la India –a través de páginas y modalidades de pago específicas que eluden toda instancia impositiva- sino también de capacidad de almacenamiento en servidores norteamericanos.

El nivel de educación formal de los socios y empleados de estas empresas es el menor del conjunto de los PPSW que analizamos: no suele haber egresados universitarios, aunque casi todos los trabajadores han iniciado algún tipo de educación superior. Finalmente, hay que destacar que en este tipo de procesos la presencia femenina sólo existe en las empresas unipersonales llevadas adelante por mujeres. Es decir, no hay ninguna combinación de sexos en estos procesos productivos.

b) Empresas pequeñas y medianas

Las empresas pequeñas y medianas conforman un grupo mucho más heterogéneo que el de las microempresas. Por lo general, son unidades productivas que están en un estado de transición –que quizás se mantenga indefinidamente-, por lo que en ellas conviven culturas organizacionales diversas y aún contradictorias. Para este trabajo, tomamos como ejemplo a una empresa -que llamaremos Tiresias- y a uno de sus socios gerentes – RA-.

Al igual que en el caso de muchas microempresas, los orígenes de Tiresias se remontan a la progresiva independización de quién fuera uno de sus fundadores respecto de sus empleadores previos. A partir de cierto momento sus trabajos independientes tomaron un volumen y una regularidad suficientes como para montar una pequeña estructura. Junto con un par de socios originales, RA reclutó a dos programadores e instaló la sede de la empresa en una habitación del departamento que funcionaba como su hogar en ese momento. Esa vida como microempresa (con otro nombre que el actual) terminó unos años después cuando la firma se mudó a una oficina independiente en el microcentro porteño, ya con más empleados. Aunque ha tenido oscilaciones importantes, la firma cuenta con entre 7 y 10 trabajadores. Algunos de ellos tienen situaciones más estables que otros dentro de Tiresias. De hecho, uno de los aspectos en los que este proceso productivo ha ido variando más es el relativo a la cantidad de trabajo que se terceriza. Así, en el 2009 la empresa decidió achicarse y aumentar la proporción del personal contratado de manera eventual. Respecto de los rasgos de los trabajadores que contrata, RA acentúa, por un lado, la sociabilidad para todos los puestos. Por otro, y para algunos puestos en particular, recalca la pasión por la tecnología. En ningún caso le resultan relevantes las credenciales académicas, pero cierta experiencia previa si puede resultarle valiosa para ciertas tareas. Los salarios mensuales de Tiresias, hacia fines de 2010, oscilaban entre los USD1000 y los USD 1500, con un promedio de USD 1100. Todos los empleados y socios presentes y pasados de Tiresias fueron hombres, con la sola excepción de una secretaria. Consultado sobre el porqué de esta proporción de género, RA amaga vincularlo con el mundo académico, pero luego duda y no ofrece respuestas claras.

La división del trabajo no es muy compleja: RA y su socio tratan con los clientes, haciendo la preventa y el diseño de alto nivel, mientras los programadores fijos y eventuales se ocupan del diseño de bajo nivel, codificación y testeado de los sistemas. Sin embargo, a diferencia de los titulares de las empresas micro, RA se sabe incapaz de realizar aquellas tareas de las que no se ocupa. Esto es, no sólo las ha dejado porque le gustan menos, o le resulta más rentable contratar a alguien para hacerlas: la misma renovación de lenguajes y aplicaciones lo ha llevado a quedar parcialmente relegado de la programación, y a concentrarse más en la actividad gerencial. De hecho, esto se

complementa con su perfil general, en el que la informática es un puro medio de vida: en caso de no obtener ingresos con esta actividad el entrevistado no dedicaría (ni dedica) ninguna porción de su tiempo libre a la producción de software. De algún modo, y a diferencia de las microempresas, aquí ya empieza a emerger el perfil del emprendedor que tiene con la computación más una convivencia pacífica que un amor encendido. RA ha analizado otros negocios en los que desenvolverse y no descarta hacerlo a futuro.

Contrariamente a otros aspectos que alejan a Tiresias de las firmas de mayor porte, respecto de los plazos y entregas la empresa se presenta mucho más racionalizada: utilizan el método Scrum, centrado en planificaciones y división de tareas semanales, con reuniones de control diarias y muy breves.

Tiresias no integra ninguna cámara empresarial ni RA parece preocupado por hacerlo. Previsiblemente, sus empleados tampoco tienen ningún contacto con el mundo sindical. La empresa no ha tenido vínculo con las políticas públicas ni le presta atención a las políticas sectoriales a la hora de tomar decisiones de inversión o de otro tipo.

En relación a los *outputs*, Tiresias produce software a medida: sistemas de gestión y sistemas web, para empresas por lo general grandes. Uno de los elementos que da una pauta de los límites de Tiresias como Pyme es el de la carencia actual de certificaciones ISO, CMM o similares. Cuando se le pregunta a RA por qué carecen de ellas, éste responde con cierta ambigüedad: “Porque lleva tiempo gestionarlo, básicamente. Y evidentemente no nos resultó fundamental con los clientes que tenemos y los procesos que tenemos. Igual, estuvo en nuestra cabeza; está”.

Respecto de los softwares utilizados como insumos, aparecen datos interesantes, aunque en modo alguno sorprendentes. En términos de sistemas operativos, se usan distintas versiones de Windows “por costumbre”. Respecto de los lenguajes, dos de los principales (Visual Basic, C#) se utilizan por la misma razón *path dependence*: RA aprendió a utilizar el primero y el segundo funciona como actualización parcial de aquél. Sin embargo, para los sistemas web se utilizan decisivamente JavaScript y Php, ambos de licencias libres. En este caso, el criterio de adopción fue que “dentro de lo que es web, son los más difundidos, los que tienen más librerías de código disponibles, más información y también podés conseguir la cantidad mayor cantidad de recursos y al

mejor costo.” Finalmente, para las bases de datos se utiliza SQL, tanto en sus versiones privativa (SQL Server de Oracle) como libre (MySQL). En líneas generales, lo que la economía de la innovación llama base de conocimientos y externalidades de redes parecen ser los factores clave para la adopción de sistemas y lenguajes. Dejando de lado los lenguajes y sistemas operativos, las aplicaciones de programación que se utilizan son VisualStudio.net, Notepad++, SQLYog, SQLServer 2005. La elección de estas últimas está muy ligada a los lenguajes utilizados. No aparece acentuada una evaluación de la eficiencia per sé, ni, mucho menos, un análisis respecto de los costos de las licencias.

De hecho, en Tiresias se utiliza un 50% de software de código abierto o libre y un 50% de software privativo ilegal. Esto es, no han pagado por ninguno de los softwares que utilizan, sean estos sistemas, lenguajes o aplicaciones: “O son libres y en otros casos son truchas. No pagamos”. Esta estrategia de una empresa que vende software pero no está dispuesta a pagar por ningún software dista de ser novedosa. En otros trabajos hemos tratado de mostrar que todos los sectores productores de conocimiento pasan por una instancia como esa (Zuckerfeld, 2010: Conclusión). Aquí nos sirve, también, para marcar otro de los límites de este tipo de proceso productivo. Posiblemente el salto hacia las certificaciones, el beneficio de las políticas públicas o la pretensión a cámaras empresariales, en fin, la transición hacia convertirse en una empresa grande, suponga alguna modificación en la política de propiedad intelectual de este tipo de Pymes.

c) Empresas grandes

Para caracterizar este tipo de proceso productivo nos focalizaremos en una de las empresas estudiadas, que llamaremos E. E surge a hacia fines de los años 80 y principio de los 90, partir de un proyecto muy pequeño impulsado por dos socios y un programador argentinos, quienes fundamentalmente comenzaron a desarrollar un producto a raíz de los ingresos generados por sus respectivos trabajos en relación de dependencia en una empresa mayor. El insumo inicial fue el tiempo de trabajo extra-laboral para poder desarrollar un producto de software empaquetado. Hoy la empresa cuenta hoy con 40 empleados y no desarrolla uno sino distintos tipos de software.

La historia de E se vio truncada y favorecida por los distintos contextos macroeconómicos del país, teniendo su primera posibilidad de crecimiento fundamentalmente de la mano de una empresa transnacional (a mediados de los 90), la cual les proveyó la demanda inicial necesaria. Su segunda posibilidad y gran retroceso posterior, surgió del crecimiento del mercado de Internet. Si en la primera instancia, E debió asociarse a otra empresa para lograr instalar su producto (E proveía el software y las capacitaciones y la otra empresa el aspecto comunicacional), sólo fue a partir del desarrollo de Internet, que E pudo comenzar su producción efectivamente autónoma. Al mismo tiempo, E modificó el tipo de productos y debió complejizar su proceso, creciendo en infraestructura recursos humanos para poder funcionar 7x24hs.

Sobre este punto es importante destacar que el *costo* de este tipo de empresas es muy alto en términos de infraestructura y licencias. Todas las herramientas y sistemas con los que trabaja la empresa son privativos y, además de poseer servidores propios, costean un servicio de *housing* en una empresa multinacional.

A su vez, E no trabaja con herramientas y sistemas libres puesto que no posee la base de conocimientos necesaria para hacerlo (como ser un área dedicada a explorar y adaptar esos sistemas). De haber algún inconveniente con los softwares utilizados como insumos, E necesita que la empresa proveedora de la herramienta se haga cargo y resuelva en un lapso muy acotado de tiempo el inconveniente suscitado. Por otro lado, por la misma razón, necesita tercerizar para replicar sus *backups* en servidores privados.

Respecto de los *outputs*, esto es, el tipo de software que E produce, observamos que se trata de software estandarizados cuyos principales clientes son corporativos: tanto empresas nacionales y multinacionales como el Estado. En este sentido, E cuenta con dos o tres productos principales y luego los adapta en función de los clientes específicos. Es por ello, que el trabajo de programación está en desarrollo constante (de los 40 empleados, 20 se dedican al desarrollo). Su principal ingreso proviene de la venta del software, mientras que las consultorías sólo están previstas en una etapa inicial como parte de la venta del producto y no como fuente paralela y constante de ingreso. En este sentido, la empresa E no podría, aunque quisiera, (al menos en esta etapa de su evolución) basar sus ingresos en consultorías permanentes puesto que no lograría vender esos servicios a un precio competitivo. De esta forma, si bien la empresa cuenta

con software de diversa complejidad, el principal desafío una vez desarrollado el producto es conseguir la clientela que se adapte al mismo, y no, a la inversa como en el caso de desarrollo de software a medida.

Por otro lado, el hecho de trabajar con clientes corporativos transnacionales, obligó a la empresa a certificar primeramente a alguno de sus productos y, luego, la metodología de trabajo y los estándares de calidad. Así, E no sólo debió realizar un fuerte gasto en consultoría para ordenar y estabilizar su proceso productivo (además del pago relativamente menor de las certificaciones que rondan los USD 2000), sino que una vez anudada en la exportación de sus productos, hubo de incorporar trabajadores permanentes que hicieran el seguimiento de las diversas normas de calidad y certificaciones.

En este sentido, E comparte con algunas empresas medianas y con las empresas multinacionales el usar una metodología de trabajo Scrum (dada en gran parte por las pautas de las certificaciones) que racionaliza el proceso de trabajo. La diferencia, claro, vendrá dada por la cantidad de trabajadores y extensiones divergentes en tiempo respecto de los equipos de trabajo y los proyectos. Así, para desarrollar un nuevo producto, E trabaja con equipos de cinco o seis personas y elabora el proyecto inicial en un lapso de entre uno y tres meses según la complejidad del producto. Una vez desarrollados los lineamientos generales, los programadores se dividen por módulos que trabajan fundamentalmente en forma secuencial.

La empresa E se encuentra atenta a las políticas públicas respecto del sector de software. Sin embargo, a diferencia de otros procesos productivos más pequeños en donde el financiamiento es el principal móvil para ello, la empresa aprovecha las capacitaciones brindadas por el sector público. Respecto del aspecto económico, se ha visto beneficiada por la Ley de Software en razón de las exenciones impositivas; pero, nuevamente, no en lo relativo a algún tipo de financiamiento.

Un último elemento a destacar respecto de este tipo de proceso productivo, que lo diferencia de las empresas multinacionales y lo acerca a aquellos procesos de menor complejidad, es el relativo a los mecanismos de disciplina y control sobre el conjunto de sus trabajadores. En este proceso productivo se observa una mayor presencia de mecanismos de vigilancia, aunque también tienen alguna incidencia las modalidades

postdisciplinarias. De un lado, la observación directa y la utilización de los espacios, la atención puesta en los horarios, en los tiempos al interior de la jornada laboral y en las acciones realizadas en la computadora nos hablan de una presencia fuerte de la disciplina. De otro lado, existen también evaluaciones respecto de la productividad que se enfocan en el resultado anual de los trabajadores.

d) Empresas multinacionales

Como fue indicado, los procesos productivos de software en empresas multinacionales son los más grandes de nuestra tipología, pero, además, cuentan con la particularidad de ser heterónomos. Esto es, las decisiones acerca de la producción se toman fundamentalmente en una casa matriz y no en la unidad productiva instalada en la Ciudad de Buenos Aires.

Como es esperable, pues, los procesos productivos de software en este tipo de empresas presentan la mayor complejidad observada: de manera general la división del trabajo está muy desarrollada, existen complejos procedimientos que incluyen burocracias administrativas para cada una de las tareas. Los outputs son en su mayoría a medida y la cantidad de trabajadores puede variar entre 100 y 300.

Este último dato depende del tamaño de la empresa, pero también de la diversidad de procesos productivos que integran una misma firma. En efecto, en ningún caso estas se limitan a producir únicamente software, sino que complementan sus ingresos a través de la venta de servicios de diferentes tipos: administración de bases de datos, alquileres de servidores, servicios de telefonía e Internet y consultorías, entre otros.

La mayoría de estas empresas poseen sede y capital de origen norteamericano, y se instalaron en la Argentina entre los años 70 y 80). Si bien cuentan con clientes en Argentina (casi exclusivamente clientes corporativos), el mayor ingreso del proceso productivo de software proviene de las exportaciones de software a medida.

A su vez, un estudio de la CESSI, que relevaba el nivel de facturación de las empresas agrupadas según la cantidad de trabajadores, confirmaba el alto grado de correlación entre empleo y facturación (OPSSI, 2010: 5). En este sentido, obviando los

distintos rubros y distintas fuentes de ingreso de las empresas multinacionales, es clara la ventaja competitiva que poseen respecto de los otros tipos de proceso productivo.

En este tipo de empresas se utiliza un software en donde cada trabajador debe cargar diaria o semanalmente la cantidad de horas trabajadas. Esta planilla es cargada de manera informal y dialogando con los líderes de proyecto. Así, aun cuando no estamos en condiciones de caracterizar la forma en que estas empresas calculan el precio de sus productos, un empleado nos cuenta que esta planilla en general es utilizada para colaborar con el cálculo estimativo del costo de cada producto. De cualquier forma, si el cliente es muy importante, esta planilla es cargada con una cantidad de horas por trabajador que excede a las efectivas.

Los mayores *gastos* en este tipo de procesos se dan, por un lado, por los costos laborales. Este no es un dato menor, puesto que en este tipo de empresas todo el poder de fuego está puesto en gestionar, mantener y capacitar a los trabajadores del software. Es en este tipo de procesos productivos donde se gesta el estereotipo de “los mimados del mercado laboral”. Por otro lado, por los costos de *infraestructura* (edilicia, artefactos, generadores eléctricos de gran potencia, etc.). Las erogaciones en *licencias* no son, en términos relativos, tan altos cabría esperar. Naturalmente, estas empresas no utilizan software ilegal e incluso poseen “auditorías” para chequear que los trabajadores no hayan descargado ningún programa irregular. Sin embargo, cuentan con importantes insumos de software libre. Siendo que el output de su proceso productivo es usualmente privativo, es interesante que estas empresas utilicen en algunos casos hasta un 50 % de software libre, obviamente ahorrándose el pago de licencias. A su vez, claro, se reservan todos los derechos sobre sus productos a través de contratos explícitos que firman los programadores al momento del ingreso a la empresa.

Respecto de los rasgos demográficos de esos trabajadores, digamos que estos promedian los 25 años de edad; solamente entre los eslabones superiores (líderes de proyecto en adelante) pueden encontrarse trabajadores de más de 30 años. Asimismo, estos procesos son los que cuentan con mayor cantidad de mujeres en relación al resto de la tipología que presenta este trabajo. Sin embargo ellas no se distribuyen de manera homogénea entre los distintos roles y, de hecho, están mayormente excluidas del área de programación (Vid. Yansen, Mura y Zukerfeld, 2011).

Como ya adelantamos, *la división del trabajo* está muy desarrollada. El proceso productivo se fragmenta, en primer término, en equipos que poseen entre 10 y 30 trabajadores. En su interior, encontramos analistas de requerimientos de alto y bajo nivel, diseñadores, desarrolladores y testers. Estos equipos están coordinados por un líder de proyecto y este, a su vez, por un *mánager*, quien administra más de un proyecto al mismo tiempo y es quien está conectado con el exterior del proceso. El líder de proyecto es quien se encarga de “administrar los recursos”. Es quien asigna las tareas según las capacidades de cada uno de sus trabajadores, según el tipo de cliente, el tipo de proyecto y los sistemas y herramientas que deben utilizarse. En las empresas multinacionales, todo responde a “metodologías de trabajo” que vienen dadas desde la casa matriz según las cuales los procedimientos deben responder a un determinado estándar de calidad y eficacia. Entre ellas se incluyen los requerimientos relativos a las normas y certificaciones internacionales. Así, la coordinación entre las distintas tareas se da a través de procedimientos escritos que indican qué y cómo hacer en cada situación, reuniones semanales, mails colectivos y sistemas de ticketing que también derivan de estas metodologías de trabajo. De esta forma, todas las acciones que van siendo realizadas a lo largo y ancho del proceso quedan registradas.

Dentro de un equipo puede haber tareas muy rutinarias y tareas más dinámicas. Consecuentemente, el grado de satisfacción con el trabajo viene dado ante todo por el tipo de proyecto en el que los trabajadores están. Al mismo tiempo, estas metodologías de trabajo cuentan con cierto grado de flexibilidad. Esto es, las grandes líneas están dadas, pero los trabajadores pueden proponer modificaciones.

En este sentido, cabe precisar cómo se realizan los cambios en los puestos de trabajo y en conexión, en los salarios. Según las habilidades técnicas, “skills” o “expertise” del trabajador, este puede ser cambiado de proyecto, pero no de rango. En efecto, la posibilidad de ascender en la escala jerárquica o lo que llaman “listas de posiciones” dentro del equipo, es muy inusual. Nuestros entrevistados nos cuentan que cuando se oferta un puesto de trabajo cada empleado puede presentarse pero debe iniciar las entrevistas y evaluaciones como si estuviera presentándose por primera vez a la empresa.

En consonancia, el salario está determinado por lo que se da en llamar el grado de *seniority*. Según un estudio del CESSI el salario de un *junior* en promedio se ubicaba, hacia fines del 2009, en torno a los USD 990, el salario de un *semi senior* en torno a los USD 1260 y el de un *senior* en USD 1650. Naturalmente, son salarios mayores que los del promedio de los otros procesos productivos.

A qué responde el nivel de *seniority*, no es algo del todo claro, pero de nuestras entrevistas surge que tiene que ver fundamentalmente con el nivel de complejidad de los sistemas y herramientas que domina cada empleado al momento del ingreso a la empresa. Así, en muchos casos esto pareciera estar determinado desde el inicio de la contratación, siendo muy improbable que la antigüedad en un puesto influya para el crecimiento del grado de *seniority*. A su vez, nuestros entrevistados afirman que la antigüedad tampoco es un tema tenido en cuenta en el salario. De hecho, un “recurso” puede entrar a la empresa cobrando un mejor salario que otro recurso que está en un rango mayor y trabajando hace más tiempo. Algunos de nuestros entrevistados aseguraron que la forma más eficaz de acrecentar el salario es cambiar de empresa.

Las empresas, conscientes de este “problema de recursos”, establecen restricciones de diverso tipo para intentar evitar que sus trabajadores las abandonen. Por un lado, obligan a la permanencia durante un cierto tiempo si el recurso está siendo capacitado por la empresa; por otro, establecen acuerdos interempresariales, impidiendo que una empresa contrate a un recurso que fuera de otra, por ejemplos, dentro de los seis meses subsiguientes luego de que el trabajador hubiera renunciado a su empresa.

Como sugerimos más arriba, dentro de la empresa el sistema de capacitaciones y certificaciones es muy importante. Como todo aquí, está estrictamente organizado. Cada empleado debe elegir una carrera dentro de la empresa, que puede orientarse a una especialización técnica o bien a una especialización en el área de *management*, e ir tomando cursos en esa dirección en la medida que el ritmo de trabajo se lo permita. Si bien esto no está predeterminado por el contrato de trabajo, los trabajadores nos cuentan que es un “compromiso” que el empleado asume y que está bien visto que lo vaya efectivizando.

Un elemento a destacar en este tipo de procesos productivos es la importancia que tiene el discurso empresario en la subjetividad de los trabajadores. Paralelamente a

la individualización de los contratos, existen fuertes incentivos de reconocimiento discursivo. Por ejemplo, algunos los entrevistados asocian el tener una mayor flexibilidad horaria o decisonal debido al desempeño diferencial que ellos habrían demostrado en relación a otros compañeros. En este mismo sentido, todo derecho concedido por la empresa, es entendido por ambas partes como un “beneficio”. Así, por ejemplo, el día de *homeworking* es un beneficio, un plan médico es un beneficio y las capacitaciones también lo son. En el mismo plano de reconocimiento, podemos ubicar las listas de posiciones internas, que no redundan en ascenso jerárquico o salarial y que no necesariamente tienen tampoco relación con el grado de seniority. Un ejemplo claro que se repitió entre nuestros entrevistados es el etiquetamiento de los trabajadores. Así aun en un mismo rango, existen distintos niveles que bien pueden estar dados por un puntaje o por una categoría que semánticamente poco tiene que ver con la actividad. Dentro de un mismo rango, por ejemplo, se puede ser categoría Rythm, Blues o Jazz. ¿Qué significa esto en términos efectivos? Que si un trabajador realizó determinados cursos, y en las evaluaciones anuales logró la realización satisfactoria de una determinada cantidad de tareas asignadas, puede ir ascendiendo en esta escala. ¿Qué gana por ascender en esta escala? En primer lugar, reconocimiento; en segundo, que le sean asignadas tareas un poco más complejas y menos rutinarias. Sin embargo, ni el plus de sueldo ni la jerarquía real tienen correlato con estas escalas.

Todo pareciera indicar que los trabajadores en estos procesos productivos deberían ir profundizando y perfeccionando al máximo su tarea específica y explotando el máximo de sus posibilidades personales, pero que la escala jerárquica y salarial está relativamente predeterminada y carece de un dinamismo equivalente al que se pregona.

Por último, es palpable en estos procesos productivos que, apoyados en parte en este fuerte discurso que indica que “cada recurso es un mundo”, existe un relajamiento de los mecanismos disciplinarios característicos de los procesos productivos de la etapa industrial. La flexibilidad horaria, la poca atención que se pone a las actividades extra-laborales que realizan los trabajadores durante la jornada, el espacio arquitectónico, los escasos o nulos requerimientos relativos a la presentación y el bajo nivel de pautaación de las tareas parecen indicar que los mecanismos preponderantes son de control más que

disciplinarios, poniendo el énfasis en la productividad esperada por cada recurso, más que en el detalle de los quehaceres diarios.

III

Producción no Mercantil de Software

e) La Producción No Laboral: El software Libre y la Producción Colaborativa.

La producción de software libre tiene rasgos que la distinguen radicalmente de las modalidades usuales en las empresas del período industrial. En varios aspectos es la forma más pura, el tipo ideal weberiano, de los procesos productivos de software. Abundan los trabajos sobre sus rasgos en términos organizativos y de licenciamiento, y no tiene sentido volver aquí sobre ellos (Vidal, 2000; Benkler, 2006; Bauwens, 2006; Zukerfeld, 2010). En cambio, podemos comentar un proceso productivo de software libre en particular, el de Debian, en el que participa una de nuestras entrevistadas. Debian es, aún dentro del mundo del SL, un ámbito de vanguardia. Un aspecto particularmente interesante es el hecho de que, a diferencia de lo que ocurre con otras redes de programadores (llamadas usualmente “comunidades”), en Debian las inclusiones y exclusiones están claramente delimitadas. Si bien toda clase de usuarios puede colaborar con el desarrollo de Debian, sólo unos pocos alcanzan el grado de “Debian Developer”, esto es, de miembros plenos de la comunidad. El equivalente a un documento de identidad que certifica tal pertenencia es el correo electrónico con un dominio @debian.org. Para obtener la jerarquía y su aval es necesario, por un lado, haber pasado una cantidad de tiempo más que prudencial realizando aportes al menos a un subgrupo determinado. Pero luego es necesario rendir una especie de examen ante un referente o guía desarrollador que irá evaluando cuando el aspirante puede ser miembro pleno. Es importante destacar que en esa evaluación no sólo se consideran aspectos técnicos generales y de conocimientos específicos sobre Debian, sino también cuestiones relativas a la filosofía del software libre y sus aplicaciones prácticas. Previsiblemente, la titulación formal no es evaluada ni representa una ventaja para este proceso productivo.

Ahora bien ¿qué derechos da el ser un “debian developer”? Además de la posibilidad de coordinar a los usuarios no certificados y de los beneficios surgidos reconocimiento de éstos, hay otro aspecto decisivo. Se trata de la posibilidad de votar respecto de las grandes decisiones relativas a la orientación del proceso productivo. En efecto, y esto nos conduce al tema de la *organización del trabajo*, en Debian los lineamientos que salen de lo cotidiano se toman de manera democrática. Más aún, se cuenta con un sistema que permite rankear, esto es, ponderar las distintas propuestas relativas a una decisión dada. Esta cuestión, que merecería mayor investigación, muestra uno de los caminos posibles para mantener al máximo la horizontalidad en procesos productivos de gran magnitud (son unos mil desarrolladores, más los colaboradores esporádicos) y respecto de cuestiones técnicas sumamente complejas.

Las tareas productivas más cotidianas, en cambio, descansan en el medio centenar de grupos que se ocupan de temas específicos como “Kernel”, “Seguridad”, “Testeo”, etc. Por supuesto, un miembro puede pertenecer a distintos grupos.

Debian funciona, efectivamente, como una comunidad extralaboral. Además de las listas que coordinan actividades productivas, los productores tienen listas específicas relativas a temas sociales. Por ejemplo, si uno de ellos viajará a una ciudad determinada, espera encontrarse con otros miembros de la comunidad que residan allí. Más aún, las DebConf, encuentros cara a cara de toda la comunidad, resultan fundamentales en ese vínculo entre los lazos afectivos y los productivos. De hecho, nuestra entrevistada comenzó a acercarse a Debian por una de esas reuniones. Decidió asistir pensando más en un destino turístico y en conocer gente que en aprender programar. Pero terminó dando sus primeros pasos, con la ayuda de acompañantes ocasionales, en la comprensión de este sistema operativo. Incluso, después, a lo largo de todo su vínculo con la comunidad Debian, la adquisición técnica estuvo estrechamente ligada a su inserción social.

Un último aspecto a mencionar es el relativo a la relación de Debian con el mundo mercantil. Por un lado, la situación del *producto* es clara: se produce sin fines de lucro y se lo distribuye gratuitamente, pero se permiten los usos lucrativos de versiones modificadas, siempre que estos sigan siendo Software Libre. Este hecho es inescindible de que sean las empresas más grandes del mundo las que financien las DebConf. Pero

por otro lado, la relación entre algunas de esas empresas y los usuarios no es tan clara. Más allá de los detalles, dentro de la comunidad hubo un debate importante ante la creciente contratación de los Debian Developers por parte de Canonical, una empresa que desarrolla Ubuntu (otra distribución de Linux). Aunque la utilización de fragmentos de código de Debian para distintas versiones de Ubuntu no generaba conflictos –dado que el código es abundante–, la contratación de programadores sí lo hizo. Y esto por el sencillo hecho de que se perdía el recurso escaso: los mejores programadores. Se los ponía a programar en algo lo suficientemente parecido como para que tuvieran su cuota de producción de SL cumplida pero lo suficientemente distinta como para que no pudiera ser reutilizado por la comunidad de Debian. Consultada por su opinión, nuestra entrevistada mostró, simultáneamente, cierto enojo con la empresa mentada y la firme intención de aceptar si se le propusiera el contrato a ella. En cualquier caso, la relación entre el Software Libre (como producto, como redes de vínculos, como los saberes de sujetos particulares) y el mercado tiene todavía bastante tela para cortar.

f) Producción Estatal

La producción en el interior del Estado se caracteriza por estar formalmente organizada pero no mercantilmente orientada. Esto es: la producción de software se da en departamentos o sectores del Estado y su producción está orientada específicamente a satisfacer las demandas internas de la administración estatal. Así son procesos productivos que en parte se parecen a los de las empresas in House o, a veces, a los procesos heterónomos: están supeditados a las exigencias planteadas por estos organismos, los procesos y actividades que éstos desempeñan, y por ende, a sus autoridades. Estos procesos trabajan atendiendo el desarrollo y mantenimiento de los sistemas y productos, así como resolviendo los pedidos de funcionalidad de los clientes internos o jefes de proyecto.

Sin embargo, en este modelo piramidal, cada bloque cuenta con autonomía relativa. Los equipos de desarrollo y sistemas están divididos en proyectos o unidades con sus respectivas autoridades y división de tareas, a nivel general y a nivel interno. Al interior

de algunos proyectos, cuentan con cierto grado de independencia en cuanto a decisiones puntuales relativas a proyectos.

Existe un alto grado de división y coordinación de tareas y roles de trabajo. No obstante, se reconocen, según las circunstancias, informalidades cotidianas en las decisiones, interacciones y tiempos de trabajo. Dependerá de cuál sea el interlocutor en cuestión: interno o externo, o si comparten la misma autoridad o no.

El output del proceso son softwares de uso interno, tanto de sistemas como de aplicaciones: se destacan que, aún formando parte de un ministerio u organismo en particular, los equipos de desarrollo a su vez produzcan para otros ministerios u organismos cuyas actividades están administrativamente vinculadas –como ser, el compartir una misma base de datos–, así como las posibilidades de replicar el mismo producto de software sobre sistemas operativos y redes homogéneas.

Por lo general no se cuenta con certificaciones de otros organismos. No hay una determinación estricta sobre la elección de los códigos y herramientas de programación, sino que tal elección depende de la preferencia de los líderes de proyectos o coordinaciones departamentales. A veces, la utilización de opciones privativas puede relacionarse con una alianza de negocios tanto del propio líder de proyecto como del departamento en cuestión. Lo mismo vale para la elección de los sistemas operativos, que en los casos relevados emplean Windows, y complementariamente Linux.

La financiación de los proyectos es una cuestión que excede a las entrevistas realizadas pero en general se dispone de recursos económicos y tecnológicos muy satisfactorios. Dada la criticidad de algunos procesos, éstos suelen contar con equipos propios de generación eléctrica.

El piso de los salarios está por encima de los USD 1000 y la antigüedad es un aspecto decisivo para ordenar las variaciones salariales. Las jornadas de trabajo son de seis o siete horas y nuestros entrevistados dedican aproximadamente un cuarto de ese tiempo a la labor efectiva de programar. Dependiendo de la posición en la estructura del proceso o de la unidad productiva (departamento o ministerio), el tiempo restante puede ser utilizado para comunicaciones y reuniones o para actividades ajenas al trabajo.

g) Procesos productivos de software académicos

Son todas aquellas producciones de software que se desarrollan al interior de organizaciones educativas, por lo general universitarias, indistintamente de ser públicas o privadas. Incluye tanto a aquellas que son llevadas adelante por equipos de desarrollo interno, orientadas a proveer soporte informático a las redes de la organización, como a aquellos proyectos de investigación orientados a alcanzar alguna acreditación educativa (título de grado, posgrado, doctorado, etc.).

Se deduce, entonces, que estos tipos de procesos productivos no son mercantiles, pese a que puedan tener aplicaciones económicas prácticas ulteriores, tanto para la organización educativa en como para otras unidades productivas que, llegado el caso, utilicen el resultado de las investigaciones o bien incorporen a los investigadores y sus saberes a su plantel. En todo caso, una apreciación sobre la utilidad práctica de los outputs de estos procesos es que se orientan específicamente a potenciar los procesos de educación y aprendizaje, sea en un sentido focalizado (de investigadores becados en particular) o difuminado (de proveer plataformas informáticas para el conjunto de la comunidad educativa).

Los procesos en cuestión cuentan con divisiones formales del trabajo, o en todo caso, de los roles de los agentes involucrados, según el caso. Suele haber personal de mayor jerarquía supervisando o coordinando los procesos; sean los tutores de tesis o bien los nexos administrativos entre las autoridades institucionales y los equipos en cuestión.

A diferencia de otros procesos productivos, se caracterizan por la marcada incidencia de los grados educativos alcanzados. Teniendo en cuenta que la actividad se da al interior de instituciones educativas, esto no resulta sorprendente. Complementariamente, no se observa la presencia ni demanda significativa de certificaciones habilitantes en productos o lenguajes de programación, o de cursos afines. En fin, en el origen de las técnicas de estos trabajadores se constata i) el papel relevante de la educación universitaria tanto en los saberes teóricos como metodológicos que atañen a la investigación; ii) la presencia de técnicas prácticas adquiridas en otros ámbitos, sean laborales o extra-laborales, o bien de técnicas desarrolladas durante el mismo proceso de producción, sean por transmisión entre pares

o experimentación. Este segundo rasgo es común a la mayoría de los tipos de procesos productivos y de los casos relevados en esta investigación (Dughera, Segura, Yansen y Zukerfeld, 2011), no así la valoración que se le da al aprendizaje formal.

Los trabajadores de estos procesos productivos suelen disfrutar y sentirse muy interesados por la actividad que desarrollan. Sostienen, asimismo, que su actividad es altamente creativa, y en ese sentido, enriquecedora.

La composición de estos grupos de desarrolladores e investigadores se caracteriza, en contraste con muchos otros tipos de procesos productivos de software, por la presencia significativa, en términos proporcionales, de mujeres (Yansen, Mura y Zukerfeld, 2011).

Es interesante destacar el empleo de sistemas operativos y aplicaciones de software libre, en particular, en los casos relevados de producción académica se constató el empleo de Ubuntu y de otras distribuciones de Linux. En ello confluyen varias motivaciones: económicas (menos costos), técnicas (mayor eficiencia) y axiológicas (esquemas valorativos favorables al software libre). Una hipótesis sobre esta dimensión podría señalar que la versatilidad y la flexibilidad de los desarrolladores de estos proyectos, así como su menor orientación a producir software diseñado para usuarios no-especialistas, los hace más proclives a adoptar este tipo de sistemas operativos y sus aplicaciones.

h) Producción de software en Organizaciones sin fines de lucro.

Contamos con dos entrevistas relevantes en este subsector. Nos valemos de ellas para delinear algunas características de estos procesos. Uno de ellos ocurre en una ONG que aquí llamaremos “Agencia”, una agencia de noticias^{iv} para radios comunitarias de América Latina que depende de una ONG mayor que agrupa a las radios comunitarias a nivel mundial. De allí proviene entonces el financiamiento. El otro caso, “Red Música”, es un sitio web creado y financiado inicialmente por particulares para servir como plataforma de interacción y colaboración entre músicos de todo el mundo. Ambos procesos se diferencian respecto del *origen* del software desarrollado (y por lo tanto también la relación con el mismo como producto). En un caso, se trata de una aplicación web de uso interno que funciona para posibilitar o facilitar una producción colaborativa

de noticias para radios comunitarias. En el otro caso, por el contrario, la aplicación desarrollada parte de una iniciativa más amplia, es a la vez medio y fin en sí mismo. Por un lado, proviene de una motivación por la disciplina musical y por ende es un medio para producir música colaborativamente, pero por el otro proviene de una motivación por desarrollar distintos proyectos de software que sirvan en un futuro de base a proyectos que si sean redituables para los creadores.

Naturalmente, de los procesos productivos tipologizados, este es el menos relevante en términos de: *complejidad del proceso de trabajo* (escasa división del trabajo, inexistencia de burocracias administrativas, formas de comunicación informales, decisiones mayormente consensuadas, plazos y tiempos de trabajo relativamente flexibles); *complejidad del producto y los lenguajes utilizados*, resultando básicamente en aplicaciones web; y de la *cantidad de trabajadores* que involucra.

NE es el único desarrollador de Agencia^v y se ocupa durante 4 horas por día de crear, mantener y modificar el panel de control y la página resultante, además de otras eventuales tareas que involucren a la tecnología y los artefactos tecnológicos en general. En el caso de Red de Música, el desarrollo inicial del proyecto involucró a una mayor cantidad de personas, pero aun así la diferencia con la producción estatal, o la de medianas y más grandes empresas es significativa.^{vi} A su vez, hay que mencionar que los trabajadores que suelen participar de alguna forma en estos procesos lo hacen desde sus hogares, con sus propios medios de trabajo y sin contrato mediante.

Coherentemente, es notable que, ambos entrevistados *no programen únicamente para la ONG* respectiva y no la consideren su actividad principal, dado que hay otras que les proveen mayores ingresos^{vii}.

A diferencia de la Agencia, Red de Música, que inicialmente tampoco prestaba atención a las *políticas públicas*, no solamente logró financiar el 50% de la segunda etapa del proyecto con el programa “Buenos Aires Emprender” sino que LM además opina que el Estado es el único capaz de solventar proyectos de este tipo.

En términos de los *costos del producto*, aquí también se establece entonces una diferencia. NE es el único costo de mano de obra para la agencia^{viii} y los costos en licencias son nulos; utilizan un 70% de lenguajes y aplicaciones privativas ilegales y un 30% libres. LM tampoco posee gastos en licencias puesto que utiliza mayormente

Software Libre, pero sí posee una inversión inicial más costosa pagando a desarrolladores y diseñadores en la elaboración inicial. Una vez en marcha, el mantenimiento lo realizan los propios socios y cuentan con colaboradores esporádicos surgidos de la propia Red de Música, de manera que la diferencia con NE se diluye y los costos terminan siendo bajos en el funcionamiento diario.

Otro aspecto relevante de este tipo de proceso productivo son las representaciones que de su actividad tienen los entrevistados, quienes son, además quienes toman las decisiones principales. Ellos valoran la actividad no mercantil y el uso de lenguajes libres, están al tanto en mayor o menor medida de las leyes de propiedad intelectual y dudan también de la factibilidad de apropiación sobre el software. Así, NE no cree haber inventado nada y piensa que no hay necesidad de protección de las aplicaciones web ya que la mayoría circula en Internet.

Si bien sería erróneo establecer una relación inmediata, hay que mencionar rasgos comunes en la educación de los entrevistados. Ambos estudiaron carreras que no finalizaron, no relacionadas al software, sino sociales o humanísticas. Tampoco transitaban por cursos, capacitaciones, o certificaciones, como sí ocurre con los desarrolladores en otros tipos de procesos productivos. Son en cambio, autodidactas en gran medida. Esto a su vez es coherente con el limitado nivel de complejidad de los lenguajes que manejan.

Conclusiones

En esta ponencia hemos partido de señalar la enorme heterogeneidad que tienen los procesos productivos de software. Muchos de ellos ocurren en el tiempo de trabajo, pero otros no. Entre los primeros, la mayoría se da en empresas cuyo principal producto es el software. Pero también se produce software en ámbitos no mercantiles (como el estado y la academia), y en firmas que venden algún otro tipo de bien o servicio. En concreto, propusimos una categorización que distingue seis tipos de procesos productivos de software: 1. Producción Mercantil de Software (subdividida entre procesos Autónomos –micro, pyme y grandes- y heterónomos –filial, en red o escalonada). 2. Producción Mercantil de Otros bienes y servicios 3. Producción Estatal 4. Producción Académica 5. Producción en ONG's 6. Producción No Laboral (Software

Libre y Producción Colaborativa). Esta distinción es el principal aporte de esta ponencia. Sin embargo, para mostrar su utilidad era necesario probarla empíricamente, esto es, mostrar que estas categorías tienen referentes concretos que vale la pena distinguir, que portan rasgos propios. Eso hicimos en el trabajo de campo que antecede a esta ponencia, aunque aquí sólo hayamos podido ejemplificar algunos de esos procesos productivos. El análisis de las microempresas, las pymes y la producción no laboral de Software Libre nos sirvieron para mostrar contrastes y coincidencias respecto de los procesos productivos más estudiados –las grandes empresas locales y multinacionales-. Por ejemplo, el hecho de que algunos de estos procesos productivos menos estudiados no sólo presenten rasgos propios, sino que pudieran ser significativos en términos del empleo, el producto o aún de las cadenas sectoriales –cosas que no podemos afirmar categóricamente pero sí sospechar-, avalaría la necesidad de contar con tipologías exhaustivas como la que presentamos aquí. Más aún, la aceptación de esta heterogeneidad coloca como un compromiso de la agenda económica y sociológica el indagar de manera cuantitativa respecto de la extensión de cada uno de estos procesos. Confiamos en que futuras investigaciones vengan a saldar esta deuda.

Referencias

- BENKLER, Yochai (2006) *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. Yale University Press: Boston.
- FORAY, Dominique (2004) *The Economics of Knowledge*, Massachusetts Institute of Technology, Massachusetts.
- OEDE Observatorio de Empleo y Dinámica Empresarial (2010) *Boletín Anual de Empleo Registrado y Empresas*, Ministerio de Empleo Trabajo y Seguridad Social.
- OPSSI (2010) *Niveles salariales en las empresas de software y servicios informáticos de la República Argentina*, CESSI, disponible en:
<http://www.guiaindustrial.com.ar/cessi/InformeENSJUNIO2010-VersionSint.pdf>
- VIDAL, Miquel (2004)[2000] *Cooperación sin mando: una introducción al software libre* en Gradin, Carlos (compilador) :() { :|& } ;: Internet, hackers y software libre, Editora Fantasma, Bs. As.
- DUGHERA, Lucila; SEGURA, Agustín; YANSEN, Guillermina y ZUKERFELD, Mariano (2011) *Sobre los aprendizajes de los trabajadores informáticos* Ponencia en el SSI de las 35 JAIIO, Córdoba.
- YANSEN, Guillermina; MURA Nahuel y ZUKERFELD, Mariano (2011) *Género tecnología y software*, Ponencia en el ASET 2011, Buenos Aires.

ZUKERFELD, Mariano (2010) Capitalismo y Conocimiento: Materialismo Cognitivo, Propiedad Intelectual y Capitalismo Informacional, Volúmenes I, II y III. Disponible en: capitalismoyconocimiento.wordpress.com

ⁱ El trabajo de campo de esta investigación consistió en la realización de 24 entrevistas en profundidad a productores de software y una entrevista en profundidad a un informante clave. Se realizó entre septiembre y diciembre de 2010 en la Ciudad de Buenos Aires.

ⁱⁱ Si se produce SL como parte de las actividades de una empresa -cosa bastante usual- estamos ante un proceso de los descriptos en la categoría Empresas SW. Por el contrario, si se produce de manera no mercantil un software sin licencias “libres” -menos usual-, estamos en un proceso que queda subsumido por esta categoría.

ⁱⁱⁱ En efecto, una objeción evidente es que la distinción entre PMSW autónoma y heterónoma sigue dos criterios distintos. Y que, por ejemplo, hay microempresas que participan en una producción escalonada. Sin embargo, en términos prácticos, aquí jerarquizamos las decisiones respecto del procesos productivos. En el caso mentado, se trata de una producción escalonada, no importando tanto la cantidad de empleados o facturación de la firma.

^{iv} En toda la agencia trabajan un total de cuatro trabajadores (Además de nuestro entrevistado – programador-, hay un coordinador, dos redactores y luego seis o siete corresponsales de noticias que son colaboradores no remunerados).

^v En todas las ONG que contactamos para intentar una entrevista, la respuesta al explicar la falta de tiempo para realizarla, fue en algunos casos que había una sola persona ocupada del sw específicamente y en otros, que luego del desarrollo inicial, no contaban con un área o encargado de la parte de desarrollo, sino que dedicaban una atención mínima de carga de contenidos.

^{vi} En el proyecto inicial trabajaron los fundadores, cuatro desarrolladores y tres diseñadores. LM nos cuenta que en la red hay miles de colaboradores y que son fundamentales para alimentar al sitio.

^{vii} En el caso de LM, considera que su actividad principal es otro emprendimiento en el que está trabajando con más dedicación en la actualidad y que es de índole comercial. Asimismo, aclara que “Música” no genera ingresos en sí misma, sino que sirve de base a otros proyectos que sí eventualmente lo hacen.

^{viii} Tanto él como el promedio de sus compañeros ganan alrededor de USD 175 por mes.