

Doctorado en Ciencia Política y de la Administración: Pensar y Gobernar las
Sociedades Complejas
Departamento de Ciencia Política y Derecho Público
Universidad Autónoma de Barcelona

***Dilemas de Acción Colectiva: Instituciones y Cooperación en
Organizaciones Tecnológicas
Redes para la Innovación***

Jorge Luís Salcedo Maldonado
Directora: Eva Anduiza Perea
Enero de 2009

Introducción	4
1. <i>Objeto de la investigación:</i>	4
Interés de la ciencia política en las organizaciones FOSS	6
¿Por qué el caso del software de código libre?	8
2. <i>Conceptos Claves</i>	11
¿Qué entiendo por cooperación y defeción en el contexto FOSS?	11
3. <i>Preguntas de investigación e hipótesis</i>	12
1. Perfil y selección de los casos	15
1.1. <i>Antecedentes de los casos</i>	16
1.2. <i>Los casos</i>	18
1.3. <i>¿Por qué elegir estos casos “exitosos” y no otros “fallidos” dentro de la comunidad de free software y open source?</i>	21
2. Marco de análisis y de desarrollo institucional	24
3. Incentivos y atributos del grupo	31
3.1. <i>Clases de incentivos</i>	31
3.2. <i>Atributos del grupo</i>	35
3.2.1. <i>Homogeneidad</i>	35
3.2.2. <i>Heterogeneidad</i>	36
3.3. <i>Tamaño del grupo</i>	40
4. Características Estructurales	43
4.1. <i>Funciones de producción</i>	43
4.2. <i>Características de los bienes</i>	46
4.3. <i>Internet y la Web</i>	50
5. Características organizativas	56
5.1. <i>¿Cómo lograr la coordinación y coherencia del producto?</i>	56
5.2. <i>¿Si todos opinan diferente en relación a lo que observan como ponerlos de acuerdo?</i>	58
5.3. <i>Cómo filtrar el ruido</i>	61
6. Características Institucionales y Reglas	64
6.1. <i>Tipos de reglas</i>	64
6.2. <i>¿Qué sucede con la propiedad intelectual?</i>	70
Conclusiones	73
BIBLIOGRAFÍA:	79

ÍNDICE DE FIGURAS Y CUADROS.

<i>Figura 1. Mapa de nuevos bienes comunes y sus sectores</i>	9
<i>Figura 2. Marco de análisis institucional y de desarrollo</i>	25
<i>Figura 3. Marco de Análisis Institucional y de desarrollo</i>	28
<i>Figura 4 Tipos de funciones de producción</i>	43
<i>Figura 5 Cuadro del tipo de bienes</i>	49

“ENTONCES ¿Quién o quiénes inventaron el Tlön? Supongo que el plural es inevitable, ya que la hipótesis de un único inventor -algún Leibniz trabajando en la oscuridad y la autorreclusión-, se ha descartado unánimemente. Se conjetura que este 'nuevo mundo' es obra de una sociedad secreta de astrónomos, biólogos, ingenieros, metafísicos, poetas, químicos, algebristas, moralistas, pintores, geómetras, guiados y dirigidos por algún misterioso hombre de genio. Hay muchos hombres adeptos en esas disciplinas diversas, pero pocos son capaces de imaginación; y aún menos son capaces de subordinar la imaginación a un plan riguroso y sistemático. El plan es tan vasto que la contribución de cada autor es infinitesimal”.

Jorge Luís Borges, Tlön, Uqbar, Orbis Tertius (1940)

Introducción

1. Objeto de la investigación:

La presente investigación es acerca de la acción colectiva en el caso conocido como Free Open Source Software (FOSS) desde la perspectiva de la economía política y de su corriente denominada como neoinstitucionalismo económico (Peters, 2003). El marco teórico que utilizo para analizar el caso es el conocido como el Análisis de Desarrollo Institucional (IAD¹, ver Hess y Ostrom, 2007:41) que durante más de 30 años ha sido utilizado por diferentes investigadores en el análisis de recursos comunes. Mi estrategia de investigación consiste en establecer un diálogo entre la teoría y los estudios empíricos acerca del FOSS.

Para las personas no familiarizadas con el tema, FOSS es el conjunto de organizaciones e individuos que con un marco institucional específico están dedicados al desarrollo de software no propietario (Linux, Firefox, Openoffice, Apache...). El desarrollo de este software responde a la necesidad de no depender de un tercero (empresas titulares de las licencias) para que programador

¹ The Institutional Analysis and Development.

y usuario si lo desean tengan la libertad de adaptar, modificar y mejorar el software. En el software privativo (los productos Microsoft son el ejemplo más común) es la empresa la que decide qué cambios realizar y bajo qué criterio de mercado desarrollar las nuevas aplicaciones que ella considere. El software que desarrolla el FOSS se denomina abierto o libre, lo que significa que no restringe el acceso al código fuente del programa informático por mecanismos técnicos o legales en el marco de los derechos de propiedad intelectual. El código fuente es aquella parte del lenguaje del programa necesario para entender las instrucciones que quiere dar el programador original y que permite a un tercero interpretar la cadena de ceros y unos que es el lenguaje binario de base que constituye un programa informático. El código le permite al programador desarrollar diferentes aplicaciones que sean compatibles y poder instalar periféricos (hardware como cámaras, impresoras, etc...) que el ordenador reconozca. También le permite desarrollar nuevas actualizaciones del programa, corregir errores y adaptarlo acorde a sus necesidades o a las de un posible usuario. Además el hecho que el código sea de libre acceso facilita que se mantengan ciertos estándares o lenguajes comunes entre las empresas que desarrollan tanto hardware como software.

Dentro de lo que se conoce como organizaciones FOSS hay dos corrientes principales, la Free Software Foundation y el Open Source. Sus diferencias tienen connotaciones tanto éticas como económicas. La Free Software Foundation liderada por personas como Richard Stallman presenta el uso de este tipo de software como una cuestión fundamentalmente de principios e ideología, en que free no debe entenderse como gratuito sino como libre. La reivindicación es el ser libre de modificar, copiar, adaptar y mejorar la información del programa sin ir a restringir su acceso. Es tal la importancia del software en el mundo en el que vivimos que se debería considerar antiético usar software privativo porque excluye de su uso a quienes no tengan el hardware compatible a las preferencias que definió el creador del programa o los medios para pagar por él en cuanto la opción

de modificar o copiar es incurrir en un delito. La corriente Open Source, entre sus creadores está Bruce Perens y Eric Raymond,² presenta una perspectiva menos idealista y más pragmática al hacer énfasis en la libertad de información y el compartir conocimiento como una garantía para la innovación. Lo más importante es el avance de la tecnología, usar o no software privativo ha de ser una decisión personal no impuesta y guiada por el criterio de la calidad que ofrece el producto más que los valores que dice representar. La postura pragmática de la corriente Open Source facilitó el acercamiento de empresas privadas al fenómeno FOSS. Los casos seleccionados pertenecen a esta última tendencia.

Interés de la ciencia política en las organizaciones FOSS

La acción colectiva es un problema central de la ciencia política con respecto a diferentes áreas como la participación política, la cooperación internacional y la economía política entre otras. Una de las preguntas centrales dentro de la acción colectiva es la motivación de los actores para cooperar: ***¿por qué ciertos actores están dispuestos a dedicar su tiempo y conocimiento a realizar actividades que no necesariamente implican una recompensa material? ¿Qué hace que un grupo esté dispuesto a cooperar ante la ausencia de una autoridad que le obligue a hacerlo?*** El caso FOSS permite estudiar este tipo de incentivos no materiales y evaluar qué condiciones pueden requerirse para favorecer la cooperación.

El caso FOSS además nos cuestiona si la teoría con la que contamos nos sirve para explicar las nuevas realidades que observamos o si es necesario replantear, ampliar y desarrollar conceptos para explicar de manera satisfactoria la acción colectiva y los principios y características que presentan los recursos de uso común (RUC) que surgen en nuevos entornos tecnológicos tales como el Internet

² Bruce Perens es el creador del término Open Source, Raymond es el escritor de uno de los libros de referencia dentro de la comunidad FOSS “La catedral y el Bazar”, 1997.

y la Web³. En ciencia política la cuestión del número de individuos que son parte de un grupo es fundamental para formular hipótesis sobre las posibilidades de éxito de la acción colectiva y gestión de bienes comunes por grupos de N-personas. Hablo de éxito en el sentido de Ostrom (1990), de favorecer el surgimiento de instituciones que permitan a sus miembros alcanzar resultados productivos en situaciones en las que las tentaciones de actuar como un gorrón o de rehuir responsabilidades están presentes.

Que el proceso de desarrollo de este tipo de software involucre un número creciente de individuos entre los que un porcentaje significativo (varia de un proyecto a otro) se encuentre descentralizado geográficamente y contribuya sin recibir por lo menos de manera directa una retribución material (aproximadamente el 50% no recibe retribución material directa, ver estudio de David, Waterman y Arora, 2003) convierte a este caso en un ejemplo que cuestiona una visión clásica de la economía política. Una visión que considera que las dificultades de coordinación, comunicación y cooperación de un grupo aumentan en la medida que el número de participantes se incrementa (Olson,1965, Hardin,1968.) Las siguientes afirmaciones de reconocidas autoridades académicas la reflejan:

“Los grupos grandes o “latentes” no tienen incentivo para actuar con el fin de obtener un bien colectivo porque, por valioso que ese bien pueda ser para el grupo en conjunto, no le ofrece a la persona incentivo alguno para pagar cuotas a una organización que trabaja por los intereses del grupo latente, o para soportar en cualquier otra forma algunos de los costos de la acción colectiva.” (Olson, 1993. p. 61)

³ Es importante no confundir los que significa Internet y la Web; el primero hace referencia a la infraestructura física (cable óptico, redes telefónicas, routers, host, satélites.....) La Web es el gran banco de datos e información, tal vez el más grande de la historia.

“Parecería, entonces que hay cierta verdad en la máxima conservadora según la cual la propiedad de todos es la propiedad de nadie. Nadie valora la riqueza que es gratuita para todos, porque el que es lo suficientemente arriesgado para esperar que llegue el tiempo propicio para su uso, sólo encontrará que ese recurso ya ha sido tomado por otro...” (Gordon, 1954: 124 citado por Ostrom, 2000:27)

¿Por qué el caso del software de código libre?

La ciencia política no puede desconocer la importancia creciente y los cambios que en las últimas décadas los ordenadores y especialmente el Internet han originado en nuestras vidas⁴. El software es el lenguaje por excelencia para poder interactuar con el hardware, navegar por la red y realizar un amplio abanico de actividades. En este sentido considero que entender las instituciones que gobiernan su desarrollo podrá contribuir a mejorar la comprensión teórica y a evaluar las implicaciones políticas y sociales que se presentan con la aparición de este tipo de tecnología.

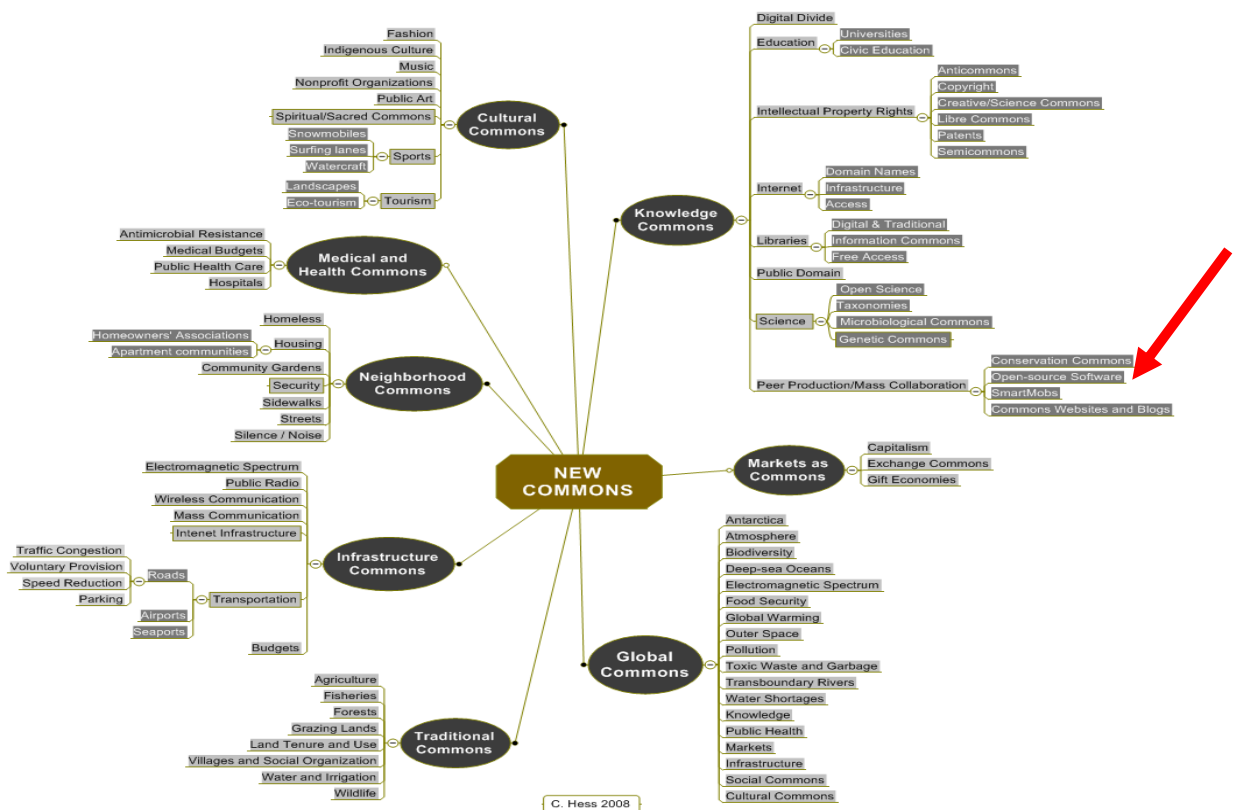
Entonces podría tomar como caso de estudio el software privativo y no sólo el libre. De hecho cerca del 90% de los ordenadores personales según la World Wide Consortium (W3C) para el año 2007 utilizan Windows, esto lo haría un caso de estudio relevante, pero la tendencia en la tecnología no son los ordenadores de escritorio sino el Internet, la Web y la tecnología móvil en las que el software libre tiene una importancia creciente.

En cuanto a la visión clásica de la economía política que considera que mientras no existan incentivos o una autoridad central coercitiva es imposible la acción

⁴ Incluso aquellos sectores de la sociedad que por diferentes motivos fundamentalmente económicos o culturales no tienen acceso a estas tecnologías se enfrentan a un nuevo tipo de exclusión, la denominada brecha digital.

colectiva en grandes grupos, Ostrom (1990) ya en su conocido estudio del Gobierno de los Comunes con un amplio soporte empírico revalúa la postura teórica de Hardin (1968) y Olson (1965). Mi interés entonces se centra en lo que Hess y Ostrom (2007) denominan “recursos comunes no tradicionales” o también llamados “nuevos recursos comunes “(Hess 2008). El mapa que Hess establece de lo que serían nuevos recursos comunes es bastante amplio (bienes culturales, bienes médicos, bienes globales, infraestructura, conocimiento...) y dentro de los bienes comunes de conocimiento ubica el software producido por las organizaciones FOSS que toma una mayor importancia con la difusión de tecnologías como el Internet que también es clasificado como un nuevo bien común.

Figura 1. Mapa de nuevos bienes comunes y sus sectores



Fuente: (Hess, 2008:13)

Hess los considera comunes en un sentido amplio del concepto, en cuanto se encuentran en un esquema de propiedad común y porque es necesario el esfuerzo de más de un individuo para gestionarlos y alcanzar un resultado deseable en el que se podrían presentar dilemas sociales que son situaciones en las que los individuos deben elegir acciones en contextos de interdependencia, en las que cada individuo si decide elegir estrategias con base a un cálculo de maximización material individual al corto plazo sus acciones producen un resultado subóptimo en comparación al que obtendría si cooperara con el colectivo.

Para el caso FOSS este tipo de dilemas se pueden ejemplificar cuando el programa en su versión inicial sólo fuera descargado pero nadie contribuyera en la creación y desarrollo del mismo. En una situación así el bien no presentaría mejoras, los posibles fallos nadie los arreglaría y menos en la eventualidad que alguien hiciera arreglos o mejoras estaría dispuesto a compartirlos. No obstante en la realidad sucede lo contrario y los casos seleccionados parecen ser exitosos ya que hay un conjunto de personas dispuestas a contribuir con la provisión y mejora del bien además de compartir con la comunidad esos desarrollos.

Las características de los bienes comunes y principios⁵ de diseño de comunidades exitosas que Ostrom (2000:69 y 148) formula se desarrollaron principalmente con base en estudios sobre sistemas de recursos naturales. En el FOSS así como en otros “nuevos recursos comunes” este no es el caso, la presente investigación pretende contribuir en el debate (Hess 2008, Ostrom 2007, Schweik, 2007...) acerca de si estos principios y características se aplican o no en situaciones en

⁵ Entre los principios de diseño de los RUC tradicionales de larga duración están los límites claramente definidos, arreglos de elección colectiva, supervisión, sanciones graduadas, mecanismos de resolución de conflictos, entidades anidadas, coherencia en las reglas de apropiación y reconocimiento mínimo de derechos de organización. (Ostrom 2000:148)

las que los principales componentes del bien y el sistema de recursos son información y conocimiento.⁶

2. *Conceptos Claves*

¿Qué entiendo por cooperación y defección en el contexto FOSS?

En el caso de estudio entiendo como cooperación la decisión de contribuir y pertenecer a una de las organizaciones estudiadas. Lo que se manifiesta mediante: 1. La descarga e instalación del respectivo programa con sus correspondientes actualizaciones y reportar así sea de manera automática posibles errores y sugerencias para mejorar el programa 2. Colaborar esporádicamente con el desarrollo de nuevas aplicaciones o adaptaciones usualmente ante un interés puntual. 3. y dedicarse a tiempo completo (cerca de 40 horas a la semana) a corregir y actualizar las versiones existentes compartiendo los aportes con los otros usuarios. Ante éstos niveles de cooperación existirán cuatro perfiles de cooperantes (Crowston, Kevin and Howison, James 2004). Los del primer nivel serán los simples usuarios que si reportan algún problema es porque tienen la función automática activada, el siguiente los usuarios activos que son los que categorizan problemas según la severidad y tópico en el *reporting system* (BUGDP), los desarrollares ocasionales y por último los *core developers* o desarrolladores centrales que son una minoría (Ghost, Glott y Robles, 2002) pero presentan el mayor nivel de compromiso, muchos de ellos son fundadores y ante los méritos acumulados tienen más responsabilidades y más competencias. (Mockus, Herbsleb 2002:318).

⁶ Autores como Tom Davenport considera los datos como bits de información, la información ya sería esos datos organizados en un contexto, el conocimiento sería la asimilación de la información la comprensión de cómo, dónde y cuando decidir utilizarla y con qué propósitos. El conocimiento implica la experiencia, la interpretación y reflexión que permite dar un uso adecuado de la información.

En el contexto FOSS la defección puede entenderse en tres sentidos. Primero cuando ante la falta de claridad en las licencias algún miembro de la organización u otro trata de apropiarse del conocimiento y de las contribuciones del grupo con fines privados. Segundo que al no coincidir con los propósitos y objetivos de la organización un desarrollador o grupo de desarrolladores decide crear un proyecto paralelo lo que se conoce como forking, que es el desarrollo por parte de un segmento de la comunidad de una versión distinta a partir del código fuente matriz lo que en ciertos casos puede llevar a la división de los esfuerzos cooperativos y puede poner en riesgo la compatibilidad y viabilidad de las distintas versiones en término de que la versión naciente difiera mucho de la original y que la cantidad de miembros de la comunidad al dividirse no sean suficientes para mantener el proyecto. Podemos hablar de diferentes tipos de forking, no es lo mismo las aplicaciones y versiones del programa que desarrolla una empresa acorde a las necesidades de un cliente o usuarios al hecho de crear una comunidad aparte que no comparte los valores o procedimientos de la comunidad madre. Es el segundo caso que divide los esfuerzos cooperativos y puede poner en riesgo la viabilidad de los proyectos lo que considero defección. La tercera forma de defección que típico es cuando alguien decide hacer actos de vandalismo como contribuir con un código defectuoso, saturar las listas de mails o los forum para afectar a la organización.

3. Preguntas de investigación e hipótesis

La investigación pretende dar respuesta a las siguientes preguntas:

1. ¿Qué permite que a pesar del tamaño y crecimiento de las redes FOSS la **cooperación y la coordinación** se mantengan?

Para dar respuesta a esta pregunta general considero necesario resolver tres preguntas adicionales que nos permitirán comprender aquellas variables que hacen posible la coordinación y cooperación en estas organizaciones y el tipo de bien que producen.

2. ¿Qué **incentivos** favorecen la **cooperación** en las organizaciones FOSS entre personas de diferentes partes del mundo que no necesariamente comparten un ámbito geográfico u obtienen recompensa material?
3. ¿Qué tipo de **características organizativas y estructurales** presenta el proceso de desarrollo de los proyectos para lograr alcanzar un resultado coherente y funcional?
4. ¿Qué **instituciones** y de manera específica qué mecanismos de resolución de conflictos desarrollan las organizaciones para evitar la **deserción**?

Las cuatro preguntas se encuentran intrínsecamente relacionadas; de la respuesta que les dé podremos comprender el proceso de desarrollo y de gobernanza⁷ de las organizaciones FOSS. Formulo las siguientes hipótesis como respuestas tentativas que he de falsear:

Hipótesis 1: *El grupo de desarrolladores como colectivo comparte ciertos propósitos y metas pero a nivel individual decide cooperar en respuesta a incentivos selectivos tanto materiales como socioculturales.*

⁷La definición que asumo de gobernanza es en relación al funcionamiento institucional esencialmente en la interacción entre distintos niveles sobretodo cuando se producen grandes decisiones que no obedecen a una subordinación jerárquica sino a una integración en red. Mi propósito no es entrar en el debate acerca de lo que diferentes autores entienden por el concepto voy en la línea de gobernanza policéntrica que define Ostrom.

Hipótesis 2: *La libertad de información junto con los mecanismos de comunicación que ofrecen el Internet y la Web han permitido la cooperación para el desarrollo del producto.*

Hipótesis 3: *La cooperación se facilita gracias a la organización anidada (modularización) y a los diferentes niveles de granularidad del trabajo en red.*

Hipótesis 4: *La cooperación en las redes estudiadas ha sido sostenible en el tiempo y ha ido aumentando porque las licencias públicas son una garantía institucional que brinda confianza a los posibles cooperantes.*

1. Perfil y selección de los casos

Las organizaciones FOSS presentan una amplia diversidad de proyectos, según el portal SourceForge.net para el 2006 estaban registrados 200.000, lo que hace de un estudio comparativo pormenorizado de todos los proyectos un propósito muy ambicioso para un solo investigador y que tampoco tenga mayor sentido para los intereses de la investigación. Además muchos de los proyectos ni siquiera están activos; solo lo están aproximadamente el 15%. Concentro mi atención en tres casos o proyectos y me guío por un criterio de representatividad y relevancia. Representatividad en el sentido del número de miembros de la comunidad, porcentaje de participación y tasa de crecimiento dentro del mercado y relevancia en cuanto me sirven para identificar las condiciones que promueven la cooperación en grandes grupos y gestión de bienes comunes.

Los casos son el sistema operativo *Linux*, el buscador *Mozilla Firefox* y el software para servidores *Apache*. Los tres cuentan con grupos crecientes de colaboradores que de distintas maneras y niveles de compromiso contribuyen con el desarrollo de los proyectos. Los casos seleccionados son más cercanos a la tendencia del Open Source.

A pesar de que me centro en estos tres casos no pierdo la perspectiva del fenómeno FOSS en su conjunto. Mi propósito no será profundizar en detalle en la minucia técnica y de desarrollo histórico de los casos seleccionados, tendré en cuenta detalles en la medida que considere me sean útiles para explicar la acción colectiva y la forma como se gestiona la información y el conocimiento entendidos como recurso común. Analizo diferentes estudios que han tenido el presupuesto, el equipo, el tiempo y el acceso a un mayor espectro de casos (ver fuentes), además de las páginas institucionales de los proyectos, páginas de portales dedicados al open source como Sourceforge y entrevistas realizadas por terceros

a los miembros fundadores. Esto me brinda el soporte empírico para contrastar con la teoría de acción colectiva y puntualmente con los estudios de bienes comunes tradicionales.

1.1. Antecedentes de los casos

Todo fenómeno presenta una historia previa y los casos seleccionados no son la excepción. Tal como lo planteo no me interesa hacer una descripción detallada de todos los posibles eventos, hechos y personajes que antecieron y presentan los casos seleccionados lo que estaría bien para un libro de la historia del software, pero me desviaría del propósito de explicar la acción colectiva y la gestión de nuevos RUC en estas comunidades. No obstante considero que antecedentes como el software Unix y situaciones como el dominio alcanzado por Microsoft son hechos que han influenciado de manera determinante el desarrollo de los casos y en esa medida es importante tenerlos en cuenta.

Unix es una referencia obligada dentro del ámbito de la informática en relación a las innovaciones y diversos aportes con los que contribuyó. Los primeros albores de Unix los podemos ubicar a finales de la década de los sesenta y principios de los setenta. Su creador Ken Thomson inicialmente lo llamó UNICS, que significaba *uniplexed information and computing services*. El contexto en el que surge este software está caracterizado por la escasez en lo que se refiere a capacidad de procesamiento y almacenamiento que tenían los ordenadores en esa época. Thomson tenía un hardware limitado y además recordemos que en ese momento los ordenadores eran un artículo de lujo que sólo unos cuantos centros de investigación podían permitirse. La filosofía que guiaba la creación de Unix era la de hacer algo sencillo y pequeño pero que a partir de pequeños elementos en red pudiese llegar a ser un todo de una gran complejidad.

Otra de las grandes contribuciones conceptuales de este programa es la noción de pipe que permite aplicar el criterio de modularización en los programas modernos. Pipe es el mecanismo o canal de comunicación que permite conectar los datos de salida de un programa con los datos de entrada de otro y así permite dividir complejas tareas en una serie de sencillas y simples instrucciones conocidas como módulos. Los módulos son programas autónomos que pueden comunicarse entre sí y que al poder interactuar permiten la creación de procesos mucho más complejos. Los creadores de Unix también se dieron cuenta de la importancia de documentar los procesos de creación de software y más de un software de tal complejidad si se pretende que otros aprendan a manejar el programa. Antes la práctica que prevalecía para aprender era la prueba y el error o contar con la fortuna de conocer a alguien que lo utilizara y estuviese dispuesto a enseñarlo. Los documentos facilitan poder aprender el lenguaje y difundirlo. Unix además contribuyó con el desarrollo del Internet, infraestructura que como hoy sabemos ha favorecido multiplicidad de fenómenos sociales entre éstos el FOSS.

No obstante, Unix no sólo dejó prácticas a seguir y legados positivos para el caso FOSS, también enseñó lo que no debe hacerse. Por ejemplo el no definir de manera adecuada los derechos de propiedad del software. Este hecho llevó a que se dieran toda una serie de disputas legales entre los desarrolladores la más conocida entre AT&T con la Universidad de California al principio de los noventas que lo único que generó fue entorpecer el posible avance del programa. AT&T reivindicaba toda la propiedad del software desconociendo los aportes y mejoras del grupo de Berkeley y de otros programadores. También con Unix se crearon tal proliferación de versiones (Forking) muchas de ellas incompatibles que esto hizo que proliferaran múltiples estándares. Unix no logró crear una gran comunidad unificada que concentrara esfuerzos y permitiera un avance más rápido del programa, los esfuerzos colaborativos se dispersaron, los estándares no eran compatibles, las reglas no eran claras.

Otro gigante de la informática que ha marcado el desarrollo de las organizaciones FOSS es Microsoft. Su modelo es opuesto a lo que profesa el FOSS; Microsoft concentra su atención en el desarrollo de software propietario con código fuente cerrado y es uno de los principales defensores de los derechos de propiedad intelectual en lo que se refiere a patentes como a copyright sobre el software, su lucha es total contra lo que califica como piratería. En menos de diez años hasta los primeros años del 2000 constituyó lo que sería prácticamente un monopolio en lo que corresponde a presencia de su sistema operativo y programas de oficina en los ordenadores de escritorio y a pesar de que en sus inicios despreció el potencial de Internet, su explorador es hasta el día de hoy el más utilizado. La importancia de citar a Microsoft como un antecedente dentro de un estudio acerca del caso FOSS es que la posición dominante de Microsoft lo convirtió en el enemigo a vencer al lograr unir a competidores tradicionales como IBM, Nokia, Sun, HP, Fujitsu (entre los más representativos) en torno al propósito común de desarrollar y apoyar alternativas para competir contra este gigante, lo que se tradujo en contribuciones y vinculación de estas empresas con organizaciones FOSS. El lobby de Microsoft ante diferentes gobiernos y organismos internacionales como la (Organización mundial sobre temas de propiedad intelectual) OMPI por la defensa de temas de propiedad intelectual y al manejar un código de fuente cerrado también hizo que los hackers lo vieran como el enemigo a vencer, esto hizo que programadores colaboraran en proyectos de código abierto que fueran una alternativa a los productos Microsoft. Tener un enemigo común aparentemente favoreció la formación de un sentido de comunidad.

1.2. Los casos

Linux es un sistema operativo que tradicionalmente ha sido utilizado para instalarse en superordenadores ocupando en este nicho del mercado más del

70%⁸, sus primeras versiones requerían una cierta profundización en conocimiento informático en cuanto la interfaz requería conocer lenguaje de programación; hoy han aparecido distintas versiones con una interfaz más amigable para los no informáticos. Entre los personajes más conocidos de Linux está su fundador Linus Torval, que sigue jugando un papel activo y crucial en las modificaciones sustanciales que se le deseen hacer al programa y centra primordialmente su atención en el desarrollo de las versiones beta y Alan Cox desarrollador británico que juega el papel de uno de los lugartenientes de Torval y concentra su atención en la versión estándar del programa.

El desarrollo de Linux se inicia al principio de los noventa hecho que coincide con la crisis de Unix y con el fortalecimiento de Microsoft. No obstante es a mediados de la década que el sector empresarial deja de percibirlo como un grupo de anarquistas radicales dedicados a programar por hobby. El crecimiento de la comunidad y rendimiento del programa hacen que empresas como DEC se vinculen con su desarrollo y surgen las primeras empresas comerciales que venden servicios especializados en Linux como es el caso de Red Hat, a finales de los noventa otras empresas como IBM, HP, Fujitsu también comienzan a contribuir con su desarrollo. Linux en sus distintas versiones y distribuciones se ha consolidado como una alternativa no propietaria que no sólo compite en términos de costes (ahorro en licencias) sino de calidad en comparación de los sistemas operativos propietarios (Windows HPC, Mac OS X) en lo que se refiere a mayor seguridad del producto, estabilidad del mismo y transparencia al estar desarrollado enteramente en código abierto. Organizaciones como la Nasa, Boeing, Oracle, SAP también utilizan y apoyan Linux. Acorde a los intereses de ciertos grupos de usuarios han aparecido distintas distribuciones con grupos dedicados a desarrollar versiones del programa, como Linux para equipos de escritorio. Todos los grupos que manejan Linux tienen características de desarrollo

⁸ <http://www.top500.org/stats/list/31/os> (consulta 11 de noviembre de 2008). De las 500 supercomputadoras en el mundo, 376 emplean alguna versión de Linux como sistema operativo.

común y reconocen valores comunes. En la investigación no me centro en una distribución específica sino en Linux como una gran agrupación.

Mozilla Firefox: Del total de usuarios de navegadores Web aproximadamente un 20% en el mundo y un 30% en Europa utilizan Mozilla Firefox⁹. Este navegador se ha caracterizado por crear nuevas aplicaciones que facilitan la búsqueda en la red como puede ser el desarrollo de las pestañas, mayor seguridad y menos requerimiento de memoria para el equipo que lo ejecute en comparación a otros navegadores. En menos de seis años ha logrado romper el monopolio que ejercía Internet Explorer desarrollado por Microsoft.

Dave Hyatt y Blake Ross crearon la versión 1.0 del navegador la que es puesta en la red en noviembre del 2004. Entre sus antecedentes está su relación con el antiguo explorador propietario Netscape el cual, antes del Internet Explorer y que éste fuera vendido junto con el sistema Operativo Windows, era el explorador más usado. Bajo el nombre de Netscape se intento liderar un proyecto de software libre pero fue un fracaso. La historia de Mozilla como organización se remonta a 1998 cuando su lanzamiento fue hecho por Netscape. La fundación existe desde el 2003 y en el 2005 la junta directiva de la fundación crea como subsidiaria y dependiente de la fundación a la corporación. La decisión se da ante la necesidad de poder gestionar los réditos que generan sus dos principales programas. (El navegador y el gestor de correos) y querer darle un enfoque más comercial. Según el estudio Mockus Audris, Fielding Roy, Herbsleb James D (2002) el grupo cuenta con doce desarrolladores centrales, 486 desarrolladores ocasionales y 6837 usuarios activos.

Apache: Es líder del mercado de los programas de servidores Web, su participación supera a productos de software propietario como el Windows Server.

⁹ <http://www.xitinot.com> (mayo 1 /2008)

Más del 50%¹⁰ de todos los servidores de Internet funcionan con Apache y es un producto de calidad en los criterios que mencioné para Linux.

Bryan Behlenmdorf es uno de sus fundadores y decidió tomar una versión del código desarrollado por el Centro Nacional de Supercomputación y Aplicaciones de la Universidad de Illinois para la primera versión del programa. El Centro decidió abandonar el software y Behlenmdorf junto con otros ocho programadores y ofreciendo su propio servidor como espacio de experimentación decidieron comenzar a desarrollarlo. El programa desde 1995 ha sido todo un éxito. Muchos de sus desarrolladores son empresarios y ofrecen servicios Web a diferentes compañías. A diferencia de Linux no presenta un gran líder visible, las decisiones más trascendentales las toma el comité de los desarrolladores centrales. Al igual que los otros dos casos es un grupo meritocrático pero menos centralizado y más pequeño en comparación a lo que puede ser Mozilla Firefox o Linux. Meritocrático implica que los programadores que hagan los aportes más significativos en términos técnicos, tengan más experiencia y respeten las reglas fijadas por la organización puedan acceder a cargos de mayor responsabilidad. IBM ha tenido un papel clave en su desarrollo al impulsar (financiera, con programadores...) la creación de la fundación Apache que ha permitido promover otros proyectos y desarrollar más rápido el software de servidores. De acuerdo al estudio Mockus Audris, Fielding Roy, Herbsleb James D (2002) este comité está conformado por 25 miembros cuenta además con 400 desarrolladores ocasionales y 3100 usuarios activos.

1.3. ¿Por qué elegir estos casos “exitosos” y no otros “fallidos” dentro de la comunidad de free software y open source?

¹⁰http://news.netcraft.com/archives/2008/03/26/march_2008_web_server_survey.html (mayo 1 /2008)

El concepto de éxito en la presente investigación siguiendo la línea de Ostrom (2000) lo entiendo como el surgimiento de instituciones que permitan a sus miembros alcanzar resultados productivos en situaciones en las que las tentaciones de actuar como un gorrón o de rehuir responsabilidades están presentes. Enfatizo que mi interés son los fenómenos de organización y cooperación en grandes grupos, constatar si la teoría sobre acción colectiva y gestión de RUC tradicionales es acorde a lo que podemos observar en fenómenos como el FOSS. De manera específica los casos seleccionados los asumo como modelos que me permitan comprender la gestión de un recurso común como es la información y el conocimiento en el contexto de la aparición de nuevas tecnologías como el Internet y la Web.

El por qué unos proyectos fallan y otros se mantienen es una cuestión que abordaré en la medida que contribuya en mejorar la explicación pero sin centrar mi atención en ello y desde la perspectiva de las características que se consideran necesarias para que prospere la acción colectiva y entienda la gestión de RUC. Al igual que Ostrom (2000) pero en un contexto diferente difiero de los postulados de Hardin (1968) y considero que es posible la cooperación en grandes grupos bajo ciertas condiciones. Me interesa contribuir en tipificar las condiciones en bienes informacionales y de conocimiento como el caso FOSS.

En este tipo de organizaciones encontramos que los individuos que interactúan producen un tipo de comportamiento colectivo en el que la cooperación prospera. Este proceso se diferencia de otras formas de producción de software en las que prevalecen sistemas jerárquicos de toma de decisiones de arriba hacia abajo en las que un centro lidera las iniciativas de cambio y es el que planifica el camino a seguir. En las organizaciones FOSS la mayoría de iniciativas de transformación del producto surgen desde la base misma de la organización (Botton-up). El diseño, desarrollo, ejecución y puesta a prueba del producto dependen más de la

comunidad en su conjunto que decide desarrollarlo y posteriormente aprobarlo o no.

Las organizaciones FOSS nos hacen reflexionar acerca de la gestión de bienes comunes y específicamente de bienes comunes informacionales como es el software. Al estudiar el proceso de desarrollo del FOSS nos preguntamos acerca de la noción de propiedad en cuanto a pesar de que el soporte de este tipo de organizaciones depende en gran medida de la iniciativa individual voluntaria (el porcentaje varía según el proyecto), una de las características del producto es el libre acceso y uso del mismo sin ser excluyente. Su proceso de desarrollo se presenta como una alternativa a la respuesta dicotómica de privado o estatal.

En las dificultades de excluibilidad se parecen a los bienes públicos y RUC tradicionales lo que lleva a preguntarnos acerca de las instituciones que presentan estas organizaciones para evitar el denominado problema del gorrón o *free rider* en el que un miembro decide intentar apropiarse del esfuerzo colectivo y disfrutar del bien sin contribuir en su mantenimiento o en este caso producción. Tema fundamental no sólo para la economía política sino en general para la gestión de la propiedad colectiva.

2. Marco de análisis y de desarrollo institucional.

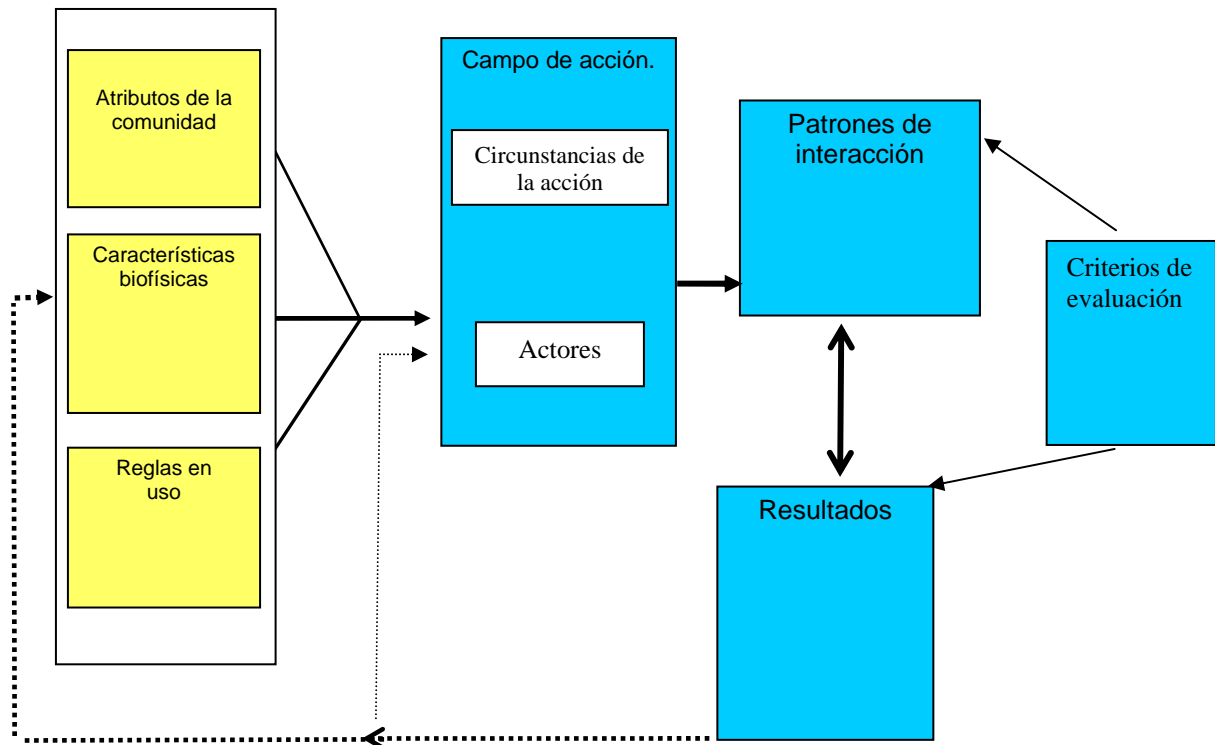
Existen diferentes perspectivas para abordar un fenómeno social. En el presente proyecto se recurre para responder a las preguntas de investigación al marco de Análisis Institucional de Desarrollo (Institutional Analysis and Development –IAD-) que ha sido utilizado durante más de treinta años por diferentes investigadores en la comprensión de sistemas de bienes comunes. El marco de IAD es una herramienta de diagnóstico que facilita el desarrollo de un método comparativo de análisis institucional (Ostrom, Hess 2007:42). Este instrumento contribuye en el intento de responder a la cuestión acerca de cómo los humanos logramos actuar colectivamente, crear comunidades y organizaciones, tomar decisiones y crear reglas que nos permitan preservar un recurso o alcanzar un resultado deseado.

El que el IAD obvie la necesidad de inventar un nuevo marco de análisis para cada una de las diferentes preguntas de investigación relacionadas con el estudio de la acción colectiva permite que haya sido aplicado en una amplia variedad de proyectos de investigación tal como lo citan Hess y Ostrom (2007:42), por ejemplo están los estudios acerca de la comprensión del rol de instituciones en el uso de recursos vitales en sociedades pobres de Agrawal (1999), los estudios comparativos entre diferentes políticas de educación superior de Richardson (2004), estudios acerca de la evolución del sistema bancario, del rol de la información en la gobernanza de recursos forestales, de derechos de propiedad en arreglos comunales en zonas urbanas y el estudio de Schweik en FOSS referencia importante de la presente investigación (Hess y Ostrom 2007:277)

El marco es posible entenderlo como el conjunto de variables independientes necesarias para poder explicar el comportamiento colectivo, está estructurado en un esquema que permite una gran flexibilidad para el investigador en el momento de determinar y entrar a considerar qué factores incluirá. El esquema presentado

por Ostrom y Hess (2007:41-82) considera que el análisis institucional ha de tener en cuenta las siguientes variables:

Figura 2. Marco de análisis institucional y de desarrollo.



Fuente (Hess y Ostrom, 2007:44)

Atributos de la comunidad: Hace referencia a las características que presentan los usuarios y proveedores del bien, con qué información cuentan, qué incentivos los motivan, los valores que comparten y el tamaño de la comunidad.

Características biofísicas en las que se desarrolla la gestión del recurso lo que implica considerar con qué tipo de medios de comunicación y tecnológicos se cuenta. En estudios previos es bien conocido como las condiciones físicas del

entorno y la tecnología con la que se dispone determinan las limitaciones y posibilidades que pueda tener la gestión del recurso común y las características que éste puede presentar.

Reglas en uso: Es el conjunto de normas compartidas acerca de lo que tiene o no permitido realizar un participante en una acción particular.

Campo de acción: Es el resultado de la interacción entre las características biofísicas, los atributos de la comunidad y las reglas en uso de los agentes involucrados en el sistema de recursos. En función de esta interacción los actores decidirán actuar de una manera o de otra. El escenario de acción tiene en cuenta las *circunstancias* en las que se da la acción y los tipos de *actores* que la cometen.

Patrones de interacción: La interacción entre todas las variables anteriormente mencionadas determina ciertos patrones. La forma como los actores interactúen llevará al éxito o posible fracaso del sistema. Dentro de los patrones es importante estudiar las posibles jerarquías que presente el sistema, los valores que guíen las interacciones y el respeto y confianza que exista o no entre sus miembros.

Resultados obtenidos: Son las metas y propósitos de la comunidad, qué ha sucedido con el recurso y si se ha alcanzado la provisión deseada. Algunos análisis institucionales comienzan por los resultados y se preguntan por qué se presenta una situación de sobreexplotación, congestionamiento u otro tipo de dilema. De acuerdo a la situación se analizan las diferentes variables para intentar identificar los factores que puedan estar desencadenando el problema.

Criterios de evaluación: Pueden ser múltiples pero han de estar en relación a las necesidades y propósitos que presenta la comunidad gestora del recurso. Entre los criterios están si se han alcanzado las metas, si el recurso ha logrado ser

preservado o producirse y mantenerse para el caso FOSS, los niveles de cooperación que se presentan, la distribución de los posibles beneficios y la eficiencia del proceso.

Hay tres formas de abordar el marco de análisis cuando estamos realizando una investigación institucional: es posible iniciar con lo que se conoce como el escenario o campo de acción compuesto por los actores y las diversas situaciones o circunstancias en las que se desarrolla la acción, también es posible iniciar evaluando los resultados que presenta el sistema , si hay situaciones de escasez , sobreutilización o bajos niveles de provisión del bien y por último iniciar el análisis por lo que Hess y Ostrom denominan factores exógenos. La presente investigación inicia el análisis por los factores exógenos compuestos por los atributos que presentan la comunidad, las características biofísicas y las reglas de uso. Iniciar por los factores exógenos es más apropiado cuando lo que se pretende es delimitar claramente la naturaleza y características del recurso que se está compartiendo (Hess y Ostrom, 2007)

presenta para el caso FOSS. A continuación abordo las características biofísicas, las cuales divido en características estructurales y en características organizativas. En las características estructurales primero analizo el tipo de bien; para ello presento algunas de las diferentes funciones de producción que un bien puede tener y entro a considerar cuál sería la que presenta los bienes FOSS, incluyo el tema de la función de producción del bien en cuanto de acuerdo a la función que presente se pueden presentar diversos tipos de rendimientos y provisión en relación al número de cooperantes. En este mismo apartado enuncio los tipos de bienes que la economía política clasifica y de acuerdo a las características de los bienes que las organizaciones FOSS producen los intento ubicar de acuerdo a lo que plantea la teoría. Posteriormente entro a considerar las facilidades tecnológicas que presenta el sistema con medios como el Internet y la Web.

En el capítulo de las características organizativas subrayo que la tecnología no es sólo el aspecto instrumental sino humano y primordialmente organizativo, lo que implica mecanismos para gestionar posibles conflictos y favorecer la acción colectiva en grandes grupos. En el capítulo sobre las reglas de uso que los sistemas FOSS presentan expongo el tipo de normas que configuran el sistema y que hacen que éste pueda considerarse abierto o no.

Para el capítulo que corresponde al de los patrones de interacción incluyo la relación entre innovación y producción. Las empresas tradicionales tienen la disyuntiva de mantenerse produciendo lo que tradicionalmente les ha dado resultado pero con la amenaza de quedar rezagados frente a las novedades que produzca la competencia o arriesgarse con algo novedoso pero que no necesariamente será acogido por el mercado y ponga en riesgo la organización. En las comunidades FOSS ¿cómo se produce y gestiona la innovación?, ¿quién decide o no qué hacer? en el recuadro que corresponde a criterios de evaluación no sólo me interesa evaluar los casos FOSS seleccionados acorde a los criterios que correspondan a la gestión por parte de la comunidad, sino lo que me interesa

es indagar acerca de los resultados en términos de las características y principios que las organizaciones y el bien exhiben y si es posible aplicarlos en otros casos diferentes a los seleccionados y fuera del ámbito del software. El recuadro que correspondería al campo de la acción no lo hago explícito ni lo incluyo en mi esquema como un capítulo lo que no quiere decir que no lo tenga en cuenta en diferentes apartados y capítulos. En los capítulos acerca de los atributos de la comunidad, los patrones de interacción y las reglas de uso entro a describir las circunstancias en las que interactúan los actores y enuncio qué y quiénes son los actores que hacen parte del sistema de recursos

Cada uno de lo recuadros será desarrollado como un capítulo de la presente tesina lo que espero me permita avanzar en la falsación de las hipótesis y en dar respuesta a las preguntas de la investigación. Tal como lo planteé ésta es una investigación en proceso, en el presente avance desarrollo la parte izquierda del esquema, los tres primeros recuadros. En la fase final del proyecto profundizaré aún más en estos capítulos y desarrollaré los otros apartes del diagrama.

3. Incentivos y atributos del grupo

Dentro de la corriente conocida como el neoinstitucionalismo económico¹¹ retomo en primera instancia planteamientos del texto de Olson, *La lógica de la Acción Colectiva* (1992 primera edición 1965) que junto a los planteamientos de Hardin (1968) acerca del dilema de los comunes podrían denominarse como teorías de la acción colectiva de primera generación (Ostrom y T.K. Ahn 2003). La teoría enfatiza en las dificultades para cooperar en grandes grupos. y establece que la cooperación sólo es posible ante la existencia de incentivos selectivos (positivos o negativos). Recompensas en el sentido positivo y en el negativo la existencia de una autoridad central sea el Estado o empresa privada que aplique mecanismos de coerción.

3.1. Clases de incentivos

Para Olson (1992) los incentivos corresponden a qué se les aplica u ofrece selectivamente a los individuos según contribuyan o no a procurar el bien colectivo. A pesar de que en su análisis privilegia los incentivos materiales, él considera que los incentivos sociales (honor, status, amistad, aceptación social...) pueden ser poderosos pero su efecto se circunscribe a grupos pequeños (1992:71). No obstante no sólo los incentivos materiales son condición suficiente para la cooperación en grandes grupos o latentes como él los denomina, él plantea que son necesarias características organizativas como una estructura

¹¹ Éste no es una teoría o paradigma en cuanto es resultado de una amalgama de planteamientos (Kalmanovitz, 2003). Retoma aspectos de la microeconomía al basarse en los costes de transacción; tiene en cuenta el derecho, en lo que se refiere al estudio de las normas y reglas que limitan el comportamiento social; aborda aspectos de la teoría de la información (manejo de información imperfecta por parte de agentes en interacción) y presenta una vertiente histórica con base en el trabajo de North (1990). Dentro de este conjunto hay autores neo institucionales más y menos cercanos a los postulados básicos de la teoría neo-clásica: maximización de la utilidad, racionalidad en la elección, escala de valores establecida, manejo de la información y comunicación.

federada que permita la existencia de pequeños grupos donde cada uno tiene una razón para unirse con los demás a fin de formar un grupo más grande o federación que los represente. En los grupos pequeños se facilita la comunicación y el intercambio constante de información entre los miembros y la posibilidad de focalizar los incentivos. En relación a esto podríamos decir en una primera aproximación que la organización en módulos interactivos con tareas específicas que presentan las organizaciones FOSS estudiadas se parecerían a una organización federada en el sentido de Olson, claro que esta afirmación la profundizaré en el apartado de características organizativas. Sin embargo, el mismo Olson cuestiona la viabilidad de las organizaciones federadas ante la dificultad de focalizar los incentivos en cuanto la mayoría de gente no cuenta con el tiempo necesario para relacionarse con un gran número de amigos y conocidos más los costes de administración creciente que se presentan con el aumento en número de miembros de una gran organización.

Los incentivos sociales funcionarán sólo en pequeños grupos o bajo una fuerte presión de los medios masivos de comunicación. Ante la imposibilidad de los encuentros cara a cara (Olson 1992:71) y al no existir presión mediática el impacto de los incentivos sociales será mínimo, lo que no nos permitiría comprender porque un grupo de individuos que no cumplen estas condiciones deciden contribuir voluntariamente en el desarrollo del software abierto sin recibir ninguna compensación material por lo menos directa (cerca del 50% de los miembros de las organizaciones ver encuesta de David, Waterman y Arora, 2003).

Ante esta posición autores como Moe (1980, citado por Molins 1998) se diferencia de la postura de Olson y retoma la importancia de los incentivos sociales al tener en cuenta aspectos de carácter ideológico y valores como el altruismo, la lealtad, la amistad, el prestigio que influyen en la decisión de pertenecer y participar o no en un grupo independientemente al tamaño del mismo lo que coincide con la investigación de David, Waterman y Arora (2003) con los

resultados del estudio de University of Maastricht acerca de comunidades FOSS (2002) y con el estudio de Lakhani, Karim and Wolf (2005) en el cual a pesar de que se le da importancia a aspectos que ellos denominan extrínsecos como son la remuneración, la experiencia, las posibilidades de contratación, consideran que en el momento de cooperar es de mayor importancia aspectos intrínsecos como la creatividad, el desafío intelectual y el contribuir al desarrollo de un producto que les ha sido útil.

Otro autor que nos permite delimitar lo que entendemos como incentivos para comprender los móviles que llevan a actuar a los miembros de las organizaciones FOSS es James Q. Wilson que hace parte de la teoría de las organizaciones (1974, citado por Molins 1998). Él considera que una organización se mantiene en el tiempo si los miembros potenciales encuentran incentivos tangibles e intangibles y los clasifica en cuatro clases ideales que se presentan en mayor o menor preponderancia en los distintos grupos, retomando su clasificación planteo lo siguiente:

- 1) Materiales individuales: Recompensas y castigos tangibles. Recursos como el dinero, obtener trabajo o la amenaza de perder cierto recurso o el acceso al mismo. En relación a este incentivo ciertas compañías, las más destacadas IBM y Nokia, destinan además de presupuesto, programadores de su nómina para trabajar en proyectos FOSS. Mozilla, Apache y ciertas distribuciones de Linux no sólo dependen del trabajo voluntario, al actuar como empresas que ofrecen un servicio para desarrollar el producto tienen empleados a quienes les pagan un salario y que están dedicados a crear nuevas aplicaciones para mejorar el producto de acuerdo a las necesidades de los usuarios. Otros de los incentivos materiales son la valoración en el mercado laboral en cuanto se pueda acreditar experiencia de ser desarrollador de un proyecto exitoso y así obtener posibles trabajos de prestación de servicios vinculados con el software.

Llegar a ser reconocido en las organizaciones es una carta de presentación que enriquece el currículum de un programador.

- 2) Solidarios colectivos: Recompensas colectivas de las que ninguno de los miembros de la organización o red puede excluirse. Lo que es una característica esencial de los bienes colectivos. En el caso de estudio el acceso al software y puntualmente al código fuente, la descarga de actualizaciones y aplicaciones en múltiples casos sin ningún coste económico. Las personas que desarrollan el producto (conocidos como desarrolladores) son al mismo tiempo usuarios lo que genera que las contribuciones que hagan al proyecto las puedan aprovechar y queden a disposición del grupo.
- 3) Solidarios específicos: Recompensas intangibles que se otorgan individualmente y su valor depende de su disfrute exclusivo y al margen del mercado. Honores, tratamiento especial o diferenciado, reconocimiento y estatus. La meritocracia como una regla de ascenso dentro de la comunidad juega un papel clave en la decisión de los programadores de cooperar, es el reconocimiento por tus aportes, la respuesta técnica ante cierto desafío te hace ganar el respeto de los miembros de la organización y poder ser elegido como en el caso de Apache, Mozilla y Linux para ocupar cargos de mayor responsabilidad.
- 4) Intencionados: También es un tipo de recompensa intangible pero en vez de venir de parte de la comunidad implica el sentimiento individual de satisfacción, de haber hecho lo que se consideraba correcto y que mayor placer generaba por ejemplo el divertirse con la programación o con el reto de intentar superar lo ya existente. El estudio de Lakhani, Karim R. and Robert G Wolf (2005) resalta la importancia de sentirse creativo como una de las razones para cooperar. Además tener un enemigo común como sería Microsoft refuerza motivos como la idea de que se está luchando para contribuir en la defensa de

una causa como puede ser la libertad de información, la reciprocidad y el poder ayudar a otros a solucionar un problema técnico lo que permitirá el avance de la tecnología. Este tipo de manifestaciones parecen más evidentes en la corriente de Free Software liderada por Stallman pero en general son comunes de los hackers tal como lo precisa Pekka Himanen (2002) y también como lo dice Torvals

“el código que simplemente funciona es un producto, código que representa una solución elegante a un problema complejo es un objeto de belleza que en el contexto del FOSS puede ser compartido con otros” (Citado por Weber 2004:136)

Tanto Moe como Wilson enfatizan que de acuerdo a las motivaciones para adherirse o no a un grupo se puede ver afectado el modo de actuar y organizarse al interior de los mismos.

3.2. Atributos del grupo.

3.2.1. Homogeneidad.

Estudios como los de Cárdenas (2003) en RUC tradicionales han demostrado que grupos pequeños y homogéneos son más aptos para preservar los RUC. Homogéneo en el sentido que el colectivo presenta una unidad de propósitos y objetivos en relación al recurso común. Efectivamente al interior de las organizaciones existen diferencias y los individuos o miembros pueden tener motivos diferentes para decidir cooperar pero solo basta con revisar las páginas institucionales:

- <http://www.mozilla.org/about/>,
- <http://www.linux.org/about/mission.html>
- http://httpd.apache.org/ABOUT_APACHE.html

De los respectivos proyectos para darnos cuenta que los grupos comparten unas metas y propósitos comunes lo cuál también se ratifica con los diferentes estudios que incluyen perfiles sociodemográficos de los desarrolladores. (Sales, 2006, David, Waterman y Arora, 2003).

Entre las metas y propósitos comunes prima el perfeccionamiento de la tecnología, considerar que ésta enriquece la vida de las personas y en la necesidad de garantizar el libre acceso a ésta mediante la libertad de información, primando valores meritocráticos para su desarrollo. Sin negar que pueda haber heterogeneidad entre los miembros el que como grupo sean homogéneos entorno a sus propósitos y metas favorece que la cooperación prospere.

3.2.2. Heterogeneidad

Desde la visión de Olson a mayor crecimiento hay un aumento de la heterogeneidad que podría ser una traba adicional para favorecer la cooperación. Sin embargo, ¿qué entendemos por heterogeneidad? Ostrom (1999) divide el concepto en tres categorías:

1. Carácter sociocultural.
2. Activos y recursos.
3. Intereses.

Carácter sociocultural

La heterogeneidad sociocultural hace referencia a atributos como la etnia, el género, la religión y el lenguaje. La presunción es que los grupos que presenten diferencias en este aspecto tendrán mayores dificultades para trabajar conjuntamente ante la dificultad de entenderse.

Las organizaciones FOSS en este aspecto son altamente heterogéneas, son organizaciones transnacionales (Tarrow, 2005) y no globales en cuanto no todas

las nacionalidades ni regiones están igualmente representadas en gran medida consecuencia de lo que se conoce como la brecha digital, concepto que esgrime que el desarrollo de las nuevas tecnologías genera nuevas formas de exclusión¹². De acuerdo al estudio de David, Waterman y Arora (2003) y los resultados del estudio de FOSS Merit (2006:37) un 53% de los desarrolladores encuestados son de Europa Occidental, 27% viven en Norte América, el 8% en Rusia y Europa del Este, el 5% en el Este Asiático, 3% en Australia y Nueva Zelanda, 3% en Latino América y menos del 1% en Medio Oriente y África.

Ante esta heterogeneidad la cooperación podría verse afectada pero debemos tener en cuenta los valores entorno a los cuales se congregan los individuos que hacen parte de las redes FOSS. Cuando hago referencia a los incentivos entre las motivaciones que guían a los individuos miembros de estas comunidades está el deseo de desarrollar la tecnología y el defender la libertad de información como una garantía para la innovación. El lenguaje común es el lenguaje de programación y el inglés. Es cierto que existen comunidades locales de los casos seleccionados que tienen el propósito de satisfacer una demanda pero los debates sustanciales acerca de modificaciones al código se dan en inglés y los aportes se desarrollan en el lenguaje de programación que utilice el software, lenguaje que es el mismo para todos los programadores. El inglés es el idioma en el que están escritos la mayoría de manuales y documentos de programación y por excelencia muchos de los términos técnicos son derivaciones o abreviaciones de este idioma lo que no es coincidencia en cuanto fue en Estados Unidos donde el desarrollo de los ordenadores, la informática e Internet alcanzaron su esplendor. Esto favorece que la mayoría de desarrolladores algo entiendan de inglés y sea más usual y sencillo utilizar esta lengua entre las personas de distintas nacionalidades y culturas que constituyen las redes FOSS.

¹² A pesar del crecimiento exponencial de las conexiones a Internet, razones principalmente económicas, políticas y generacionales hacen que aproximadamente solo el 22% de la población mundial tenga conocimientos informáticos básicos, la posibilidad de aprenderlos y de acceder a los equipos. <http://www.internetworldstats.com/stats.htm> (consultado 20 noviembre de 2008)

Activos y recursos.

El nivel de competencias de todos los miembros del grupo no es el mismo. Que organizaciones económica y tecnológicamente representativas como el caso de IBM, Sun, Nokia o Hewlett Packard o de gobiernos como China, Brasil o ciertas dependencias del gobierno de Estados Unidos (Nasa, Pentágono...) utilizasen y contribuyesen con el desarrollo del productos FOSS generaría grandes diferencias en términos de riqueza y posibilidades de influencia entre los miembros. Este tipo de heterogeneidad tal como lo precisan los estudios de (Blomquist, 1992 e Issac and Walker, 1988) en RUC tradicionales podría impactar negativamente en el interés de organizarse y en querer contribuir con el grupo.

Sin embargo, la vinculación de estos actores más que ser un problema ayuda a un mayor desarrollo de las comunidades. IBM por ejemplo ha cedido parte de sus patentes a la comunidad FOSS y ha contratado desarrolladores para mejorar el programa lo que ha repercutido en una mejora del producto. Lo importante parece ser tener las reglas de propiedad definidas para que ningún actor se pueda apropiarse de las posibles mejoras. Las empresas que contribuyen con el desarrollo de este tipo de software parecen comprender que la mejor forma de innovar y mantenerse a la vanguardia es compartiendo esfuerzos y conocimiento.

En referencia a los gobiernos la razón para contribuir con la comunidad es la seguridad y la independencia. Seguridad en cuanto no existe la intención de que un tercero controle el código, elemento fundamental que podría permitir manipular el software que procesa información considerada de seguridad nacional e independencia al no requerir de un tercero (en muchos casos extranjero) para desarrollar o dar mantenimiento a un software que gestiona información y datos en muchos casos confidenciales. Un último criterio sería la transparencia que por lo menos en los regímenes democráticos permitiría que personas con conocimientos técnicos ajenas a la Administración pudieran saber qué tipo de algoritmos se

estuviesen implementando en un software que se estuviera utilizando con fines públicos como por ejemplo el que se usa en cómputos electorales.

Intereses.

Tal como expuse existen diversos incentivos por parte de los miembros para decidir cooperar. Las razones del programador de una empresa como IBM para contribuir a desarrollar proyectos FOSS no son iguales a las de otra persona que dedica parte de su tiempo libre a desarrollar un proyecto sin que se le pague u ordene. La tensión que se pudiese dar entre los intereses de los contribuyentes voluntarios o de una gran corporación también podría ser un motivo que afectase a la cooperación.

Sin embargo, también puede darse que haya miembros de la comunidad que estén dispuestos a asumir costes mayores con el propósito de obtener el bien en cuanto consideran que el beneficio colectivo e individual que obtienen del mismo es mayor que los costes en los que incurren y deciden asumir la mayor e incluso la totalidad de la carga de su desarrollo y mantenimiento. Para Olson (1992:44) esta situación es posible en grupos pequeños marcados por la desigualdad de intereses entorno al bien. El que programas como Linux, Apache o el Firefox de Mozilla hayan logrado en algunos casos superar y competir en igualdad con el software propietario parece ser motivo suficiente para que empresas como DEC, IBM, Sun o HP entre otras tengan interés en contribuir en su desarrollo y decidan aceptar las reglas que rigen a las organizaciones FOSS. Esas reglas junto con la función de producción que el bien presenta parecen ser condiciones que hacen que la heterogeneidad más que ser un inconveniente para la cooperación sea una ventaja, tema que se desarrollará en otro capítulo.

3.3. Tamaño del grupo

Para comprender lo que se entiende como un grupo pequeño, intermedio o grande -lo que puede resultar bastante ambiguo- cito a Olson retomando el texto de Ostrom. (1990:32):

“su definición de un grupo de tamaño intermedio no depende del número de actores involucrados sino de lo conspicuas que sean las acciones de cada persona.”

La explicación de Olson considera que la diferencia en la categoría del tamaño es el resultado del valor que se le atribuye a las acciones individuales, lo significativas o sobresalientes (conspicuas) que puedan ser la acciones de uno o de algunos de los jugadores, la influencia que pueda tener la acción del agente en el mercado. La noción del tamaño del grupo dentro de la economía política está íntimamente asociada con la idea de competencia perfecta y posibles colusiones o formaciones oligopólicas. De acuerdo al tamaño él clasifica los grupos en privilegiados, intermedios y latentes. En el primer caso por lo menos uno de los miembros tiene el incentivo para que se proporcione el bien colectivo e incluso está dispuesto a asumir los costes de suministrarlo. Se presume que se puede garantizar el suministro sin coordinación y organización que lo soporte. En el grupo intermedio no hay un beneficio suficiente que incentive a uno de los miembros a asumir los costes. No es predecible si el bien será o no obtenido, para ello se requiere coordinación y organización. Por último está el grupo latente que sería una analogía a la competencia atomística, se distingue por el hecho que ante la participación o no de un miembro los otros no sentirán un mayor efecto. En éste una persona sola a pesar de una posible contribución notable no logra con su esfuerzo garantizar el suministro del bien, como ejemplo Linus Torval que necesitó del grupo para desarrollar el proyecto embrionario que él había logrado; sin la

comunidad su vida entera no hubiese alcanzado para desarrollar un software de las dimensiones actuales de Linux.

En los grupos numerosos o latentes los costos de organización serán más altos y los individuos estarán más lejos de obtener una provisión óptima del bien. De esta manera el individuo tendrá pocos incentivos para analizar si vale la pena o no realizar su aportación. La fracción que le corresponde de beneficios se verá disminuida a medida que entren nuevos miembros a la asociación y sumado tanto a los costes de organización crecientes como a los de producción del bien desencadenará en que no resulte conveniente o incluso relevante cooperar. La asociación solo proveerá el bien colectivo a menos que existan incentivos separados y selectivos que obliguen a actuar en favor del interés común (Olson, 1992:60). El análisis de cómo el tamaño del grupo afecta a la cooperación también se puede explicar con el siguiente ejemplo: a medida que aumenta el número de miembros, la aportación del individuo sólo representa una pequeña y casi imperceptible diferencia en la suma del bien colectivo que el grupo obtiene. Al mismo tiempo, cada aportación reduce euro a euro el volumen de consumo personal y de beneficio privado y las tasas marginales de sustitución que van disminuyendo convierten estos sacrificios en algo cada vez más oneroso. Como mecanismo de compensación, en los grupos numerosos, el colaborador racional tiende a contribuir menos a la obtención del bien colectivo (Olson, 1985:205)

Ante una reducción en los posibles beneficios y las dificultades de focalizar el castigo en los grandes grupos, racionalmente hay incentivos para gorronear (free riding) consecuencia de la flexibilidad del tamaño. Esto último conlleva a cierta jerarquización y división de roles que favorece una "disidencia" práctica en cuanto su "contribución puede no ser perceptible" (Olson, 1992: 64). La situación la podemos aplicar a un sinnúmero de situaciones cómo el decidir participar en procesos electorales, el pertenecer a un sindicato o el pagar impuestos. Para

Olson el tamaño determinará si es o no posible que el interés individual dé lugar al comportamiento orientado hacia el grupo.

En resumen desde la perspectiva olsoniana si no hay incentivos selectivos ni una federalización del grupo cuando éste crece tienden a aumentar los problemas de coordinación que dificultan la sostenibilidad en el tiempo de la acción colectiva. A mayor tamaño será más difícil identificar a los gorriones y en esa medida focalizar un tipo de represalia contra aquellos que caigan en defección. En este escenario el beneficio por traicionar aumenta y parece que la única forma de superar esta situación será la propuesta de Olson (1965:2) de imponer incentivos selectivos que en su dimensión negativa se traducen en mecanismos de coerción o algún otro dispositivo tipo Gran Hermano o Leviatán que obligue a los individuos a actuar en favor del interés común. Como muestra, en una huelga poner matones a la entrada de la fábrica y no permitir que entre ningún empleado que no esté de acuerdo en parar.

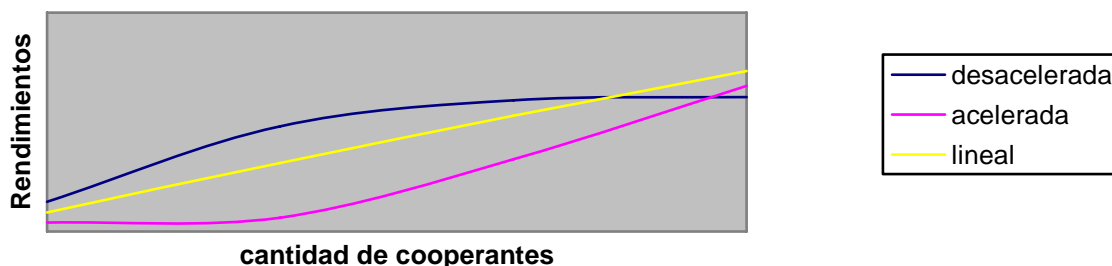
De todas maneras, el desarrollo y la expansión exponencial de algunos proyectos FOSS con un porcentaje considerable de trabajo voluntario parece poner en duda la visión olsoniana en cuanto el mayor crecimiento experimentado aparentemente no ha redundado en menor cooperación. La creencia que el tamaño importa estaría sobredimensionada. Ostrom (1990, 2005) con sus estudios en sistemas de recursos comunes y con un amplio soporte empírico identifica cooperación exitosa en grandes grupos como también observa que puede fallar en grupos pequeños (Ostrom, 1990:188).

4. Características Estructurales.

4.1. Funciones de producción.

La función de producción que relaciona las acciones individuales con los resultados del grupo puede tomar una amplia diversidad de formas. Las predicciones teóricas dependen de manera sensible de la forma particular de la función de producción en la cual todos los participantes son homogéneos o tienen diferentes niveles de intereses.

Figura 4 Tipos de funciones de producción.



Adaptación personal (Goodin, 2007 artículo Ostrom: 192.)

Las funciones lineales se utilizan en juegos de bienes públicos donde la predicción es que grupos homogéneos contribuirán con cero recursos. Este tipo de función es característica en los casos que aborda Olson. La provisión del bien aumenta de manera lineal en proporción a las contribuciones de los participantes. Para lograr una mayor provisión del bien es necesario un grupo mayor, el problema es que ante el acceso abierto al usufructo del sistema, se desincentiva a los posibles cooperantes, a menos que el grupo sea heterogéneo en sus intereses y alguien esté dispuesto a asumir unos costes desiguales ante el beneficio que el bien le

representa o que exista una autoridad como el Estado que obligue a los miembros a cooperar como en el caso del pago de impuestos.

Entre otros tipos de funciones de producción estudiadas por Marwell and Oliver (1993) están las funciones desaceleradas en las que cada contribución incremental aumenta el total de beneficios que el grupo recibe pero los retornos marginales van decreciendo a medida que más individuos contribuyen. Cuando las contribuciones son hechas secuencialmente, las contribuciones iniciales presentan un rendimiento mayor que las últimas contribuciones, es un ejemplo de rendimientos decrecientes dado el caso que acabáramos de hacer una fiesta en nuestra casa que tiene un espacio limitado de 30 metros cuadrados, el que uno nos ayude a arreglar el piso su aporte será significativo, que dos lo hagan también, pero a medida que más personas se involucren en los trabajos de limpieza su aporte marginal tenderá a disminuir, las posibles tareas ya estarán asignadas y al estar en un espacio limitado más que poder contribuir, estorbarán y se desencadenará tal confusión, congestión y dificultades para limpiar que en un caso extremo se generará más desorden lo que al final hará más difícil el proceso de limpieza.

En relación a la función acelerada ésta refleja que las contribuciones iniciales requieren un gran esfuerzo, los rendimientos en una primera fase son pequeños ante la magnitud de la tarea pero al presentarse una interdependencia positiva cada contribución adicional hace que otra tenga más sentido o valga más la pena y así sucesivamente. La predicción para grupos homogéneos y funciones aceleradas es pesimista. La clave es que haya heterogeneidad de intereses reflejada en niveles diferentes de cooperación en los que algunos miembros están dispuestos a contribuir más y asumir costes iniciales es lo que Olson (1968) denomina grupos privilegiados e idea que Shelling (1978) desarrolla con la necesidad de una masa crítica necesaria para favorecer la cooperación. Este tipo de función es la que encontramos en las organizaciones FOSS exitosas. Todo

proyecto requiere de una comunidad mínima que lo haga sostenible y de alguien o algunos como Linus Torval, Blake Ross o Bryan Behlenmdorf junto con los *core developers* que se dediquen a tiempo completo al proyecto y estén dispuestos a iniciarlo. Sus motivos no son los mismos que los de un simple usuario (no programador), aunque algunos podrían coincidir. Los *core developers* están dispuestos a hacer mayores contribuciones en término de tiempo y aportes para el desarrollo del programa porque este se ha convertido en parte esencial de su desarrollo profesional mientras que un simple usuario así pueda identificarse con ciertos valores que representa el programa lo que desea es un producto de calidad y en algunos casos más económico o gratuito.

Otro elemento a tener en cuenta es el de los costes iniciales; un grupo tiene que organizarse como tal si quiere desarrollar un bien colectivo y esto implica un coste. La primera unidad del bien es siempre la más costosa según Olson (1992:32), claro que ésto depende de la función de producción del mismo. En esa medida es fundamental tener en cuenta quién o quiénes estarán dispuestos a asumir los costes iniciales para producir el bien. En los casos estudiados encontramos la necesidad de una masa crítica en términos de Schelling (1978). Sólo cuando un grupo alcanza un mínimo número de miembros y se cubren los costes iniciales es probable que se presente un “efecto bola de nieve” que incentive a otros a contribuir. Fue vital que los costes iniciales fueran asumidos por personajes como Torvals pionero de Linux o el grupo de desarrolladores fundadores tanto de Apache como de Mozilla, sin ese impulso inicial lo más seguro es que el proyecto se hubiese quedado tan solo en una buena idea. El que los costes iniciales estén cubiertos nos puede ayudar a explicar porqué las organizaciones más grandes usualmente son las que más cooperantes atraen. (Ver estudio de Schweik, English, Kitsing y Haire, 2008)

El tamaño en este caso también es un indicador del horizonte de futuro que presenta el proyecto, el crecimiento sostenido es un indicador de lo atractivo que

resulta para los potenciales cooperantes el proyecto. Tal como lo plantea Axelrod (1986) la cooperación dependerá en una alta probabilidad de un juego al largo plazo sin una fecha final previamente establecida (amplia perspectiva de futuro y seguirse encontrando) aspectos que aparentemente están presentes entre los miembros de las diferentes organizaciones que desarrollan el software que no sólo están vinculados con el mismo a través del desarrollo sino que al ser usuarios les interesa el correcto funcionamiento del programa y su constante avance.

Su perspectiva de futuro radica en el usufructo, desarrollo conjunto del producto y crecimiento del mismo más que en el encuentro cara a cara con los otros programadores. A pesar que en algunas situaciones el encuentro cara a cara entre los programadores más activos y reconocidos pueda darse en eventos (congresos¹³, seminarios o ante una diferencia de una gran magnitud entre los líderes de la organización que amenace con un cisma en la organización¹⁴) organizados por la misma comunidad o por fundaciones dedicadas a promover los productos FOSS, claro está que no es una situación necesariamente periódica como las que identifica Axelrod.

4.2. Características de los bienes

Los bienes que estudia Olson (1968) a medida que más usuarios los utilizan reducen el beneficio disponible y pueden presentarse problemas de congestión y sobreexplotación en gran medida consecuencia de suministrarse en un contexto de acceso abierto o no excluibilidad. Él considera que el beneficio total será dividido haciendo que la fracción que les corresponde a los participantes cada vez

¹³ Por ejemplo Debian organiza la conferencia anual Debian , GCC convoca a GCC Conferencia, y GNOME organiza GUADEC, que significa GNOME Users' And Developers' European Conference.

¹⁴ Una situación excepcional en la que líderes de Linux se han reunido cara a cara para zanjar diferencias técnicas fue a finales de septiembre de 1998, cuando Linus Torvals y Dave Miller, con la intermediación de Eric Raymond se encontraron para llegar a posibles acuerdos que permitieran liberar de carga de trabajo a Torvals y no ir a entorpecer el desarrollo de Linux ante el crecimiento de la organización. Para profundizar en la historia ver Weber (2004:119).

sea menor, lo cual es situación de los bienes rivales. La provisión del bien depende de manera lineal de las contribuciones de los participantes. En igual proporción a mayor número de participantes los costes aumentaran. No obstante él hace la salvedad que ciertos tipos de costes por ejemplo a nivel personal en algunos casos pueden tender a disminuir como consecuencia de economías de escala (Olson, 1968: 56-57) pero los costes de organización a medida que el grupo crezca siempre serán crecientes lo que sumado a la reducción de beneficios aleja al grupo de la posibilidad de lograr una acción colectiva exitosa entonces sólo con coacción u incentivos se logrará la provisión del bien. (Olson, 1968:58)

En el estudio de Ostrom (2000) los RUC tradicionales presentan las características de excluibilidad y de sustractibilidad. La excluibilidad se refiere a la posibilidad o no de poder limitar el libre acceso al sistema pues esto resultaría muy costoso o prácticamente imposible en un espacio de acceso abierto como puede ser el mar, un bosque o un río. Es más los sistemas de recursos comunes suelen ser lo suficientemente grandes como para permitir su utilización de manera simultánea por muchos usuarios. La sustractibilidad se explica en la medida en que cada usuario al extraer unidades del recurso reduce (pescado, árboles, especies de cierto animal...) el beneficio disponible de los otros usuarios ya que cada unidad extraída de un recurso reduce la posibilidad de otros usuarios de consumir esa misma unidad, son pues bienes rivales. De la misma manera tampoco se puede excluir a otros de las mejoras o contribuciones que se le haga al sistema de recursos: cuando los pescadores deciden hacer un periodo de veda o los leñadores deciden reforestar, en estos casos todos en la comunidad se verán beneficiados por estas mejoras¹⁵. Por lo tanto muchas personas tendrán incentivos privados para disfrutar de estos beneficios sin sacrificar nada a cambio. Esta persona es la que Ostrom define como “el gorrón” o “free rider”.

¹⁵ Los bienes comunes presentan problemas de congestionamiento y sobre utilización vinculados con las mismas categorías que definen la sustractibilidad y excluibilidad. A pesar de que los RUC (recurso uso común) son abundantes y en alguno casos presentan capacidad de auto-regeneración, éstos son finitos.

En lo que se refiere a las características que Olson y Ostrom identifican, Benkler, Yochai (2006) nos habla que a diferencia de lo que sucede en los RUC tradicionales, la información se considera un bien no rival (“nonrival”), el que otro decida utilizarla no me impide a mi hacerlo. No es cómo un bien privado o RUC tradicional que si alguien utiliza (por ejemplo mi martillo) me impide utilizarlo o si acaban con el atún blanco del Cantábrico no pueda consumirlo más, el que otra persona vea la noticia que yo veo o descargue e instale el programa que yo instalo no me impide a mi utilizar el programa o ver la noticia. En esa medida los productos de software del caso FOSS son bienes no rivales que presentan las características de no excluibilidad y de no sustractibilidad. Los beneficios individuales no se reducen al compartir el beneficio total.

La no excluibilidad o acceso abierto no debe entenderse en el mismo sentido en los RUC tradicionales que en los bienes informacionales debido a que estos últimos no presentan de manera “natural” problemas de congestión o sobreutilización que conduzcan a situaciones como las que anuncia Hardin (1968). Enfatizo “natural” en cuanto ciertos esquemas institucionales pueden desencadenar en dilemas sociales que trataré en el capítulo de instituciones.

Dentro de la clasificación de bienes no rivales estaría la categoría que Steven Weber (2004) denomina un antirival good que son bienes creados o resultado de un proceso de intercambio recíproco y de mútuo beneficio. Los productos FOSS entrarían en esta categoría. Presentan características de los bienes públicos puros donde no hay problemas de congestión y sobreutilización. Se generan efectos positivos que tienden a autoreforzarse en las que los individuos se benefician al colaborar con el proyecto y el que más personas utilicen el producto y se vinculen con el mismo lo hace más viable, un ejemplo de este tipo de bienes que cita Weber (2004) son los idiomas, que tú por ejemplo hables Catalán para los catalanes es una ventaja, que tú lo hables no excluye a otros del uso del lenguaje

sino que permite utilizarlo con más personas, reducir los costes en tiempo y pérdida de información que acarrear las posibles traducciones, lo que hace conveniente un uso masivo contrario a lo que sucede con otros bienes. En cuanto más gente hable un idioma más usado será, se ampliará su vocabulario y los catalanoparlantes se verán beneficiados al poder ampliar su léxico y sus zonas culturales de influencia. Es lo que sucede cuando se asumen estándares, los costes de transacción tienden a reducirse. Estas características hacen que los bienes antirivales sean difíciles de clasificar dentro del cuadro de tipo de bienes desarrollado por la pareja de los Ostrom en 1977.

Figura 5 Cuadro del tipo de bienes.

<i>B i e n e s A n t i r i v a l e s ?</i>		<i>R i v a l i d a d</i>	
		B a j a	A l t a
Exclusión	Difícil	<i>P ú b l i c o s</i>	<i>R e c u r s o s d e u s o c o m ú n</i>
	Fácil	<i>T i p o c l u b .</i>	<i>P r i v a d o s</i>

Adaptación personal (Hess-Ostrom, 2007:9)

El que el FOSS produzca bienes antirivales y presente una función de producción acelerada son algunas (las otras son institucionales y de incentivos socioculturales como identificación con la comunidad) de las condiciones que nos permitirán comprender porque las grandes divisiones de las organizaciones no son algo recurrente. Los proyectos requieren de una masa crítica en cuanto los costes iniciales son altos, iniciar un proyecto nuevo me implica intentar conseguir una comunidad que lo apoye. Este tipo de bienes se benefician de la estandarización, que más personas lo estén utilizando y contribuyendo en su desarrollo. No tiene sentido crear un nuevo proyecto y dividir esfuerzos si esto implica perder los beneficios del estándar que aparecen con un mayor crecimiento. Sólo tendría

sentido en cuanto este tipo forking respondiera a los intereses de un grupo de usuarios y desarrolladores que permita garantizar esa masa crítica que hace sostenible la provisión del recurso, es el caso de ciertas distribuciones de Linux tipo Ubuntu enfocada en equipos de escritorio.

4.3. Internet y la Web.

En el clásico libro de Ostrom (2000:11) “El Gobierno de los Bienes Comunes”, la autora enfatiza en como cuando los usuarios de un RUC no tienen la oportunidad de comunicarse tienden a sobreexplotar el recurso dándose una situación similar a la que predice Hardin (1968). En esa línea de análisis también resalta que cuando los usuarios se pueden comunicar y tienen la opción de deliberar para alcanzar acuerdos públicos los niveles de cooperación son mayores y se favorece la acción colectiva.

La tecnología con la que se cuenta determina las posibilidades y limitaciones en la gestión que se le pueda dar a un sistema de recursos comunes. El que Internet y la Web permitan una comunicación descentralizada geográficamente sin que los agentes estén obligados a coincidir temporalmente es un instrumento que inmediatamente amplía las barreras naturales de la comunicación humana al facilitar la gestión de un recurso sin necesidad de encuentros cara a cara.

El que se pueda hablar de redes FOSS transnacionales con una amplia y creciente cantidad de miembros es gracias al desarrollo del Internet y la Web. Efectivamente antes del Internet existieron redes transnacionales (ver Tarrow 2005) y creaciones como la imprenta en su momento facilitaron el intercambio de información entre personas y asociaciones en diferentes partes del mundo. Castells (2001) también considera que Internet ha favorecido el crecimiento y explosión de las redes pero éstas antes existían. La situación es que este nuevo medio no sólo facilita una comunicación más expedita y deslocalizada sino que en

las redes existentes permite la retroalimentación de los procesos organizacionales prácticamente en tiempo presente -o en el momento que están sucediendo las cosas- lo que facilita el surgimiento de nuevas manifestaciones culturales e institucionales que transforman algunas formas de organización tradicional.

El Internet y la Web potencializan las posibilidades de comunicación al ofrecer una amplia diversidad de medios tales como foros, listas de mails, boletines, chats en los que se discuten diferentes aspectos concernientes al desarrollo del producto y la organización. Esta diversidad de medios crea un entorno propicio para la cooperación. La Web suministra los medios para que un gran número de voces de diversas tendencias sean escuchadas y puedan ponerse en contacto y organizarse. Es lo que observamos en los foros, wikis y listas de mails de algunos proyectos FOSS. Las redes que surgen con el Internet y la Web son formas de comunicación e intercambio de información en un espacio de cooperación que está disponible las 24 horas del día en cualquier lugar del mundo dónde exista la posibilidad de conectarse.

A pesar de que las organizaciones FOSS se autodefinen como comunidad en cuanto comparten una serie de valores comunes, característica que hace parte de la noción clásica de comunidad, hay otra serie de elementos que no están presentes como la existencia de relaciones más de carácter personal, clánico, familiar, tribal, hasta nacional o nacionalista y que sus miembros compartan un territorio y espacio físico tal como lo encontramos en los casos que estudia Ostrom (1990). No obstante esto es un tipo ideal y en cuanto los grupos se autodefinen como comunidades no tendría mayor sentido llamarlos de otra manera, además los podemos entender como una comunidad de práctica¹⁶ entendida como ese grupo de trabajo que comparte e intercambia ideas con un propósito común en este caso el perfeccionamiento de una tecnología. Sin embargo, considero

¹⁶ Agradezco los comentarios y críticas al grupo POLNET y en esta referencia a Camilo Crisanchó que me remitió a la definición de Etienne Wenger.

importante resaltar la característica de red que presentan estas organizaciones lo que no es excluyente con la noción de comunidad. Las redes sociales que encontramos en la Web al igual que las comunidades se construyen con base a afinidades y reflejan los incentivos que llevan a los cooperantes a hacer parte de ellas pero al no contar con un espacio físico por lo menos en el sentido clásico sus límites son más porosos o difusos. La red se configura a partir de las relaciones entre los diferentes nodos, la cantidad y tipos de las mismas en los que ciertos nodos están altamente interconectados y otros aislados. Torval o Cox serían nodos altamente interconectados. La delimitación geográfica deja de ser determinante lo que importa es el compartir incentivos comunes.

En los recursos de uso común RUC tradicional el no poder establecer límites físicos claramente definidos sería un inconveniente para la cooperación¹⁷ (Ostrom, 1990:148.). No obstante, establecer esos límites para el FOSS no parece tan sencillo en cuanto que el entorno donde se produce el recurso es la Web. También la noción de comunidad en la que el territorio juega un papel determinante de reconocimiento de sus miembros no se aplica en el caso FOSS. Las redes representan nuevas formas de sociabilidad tal como lo cita Castells (2001 p.146). En este caso la red permite el desarrollo de productos pero también permite la creación de lazos interpersonales, proporcionan sociabilidad, información, un sentimiento de pertenencia e identidad social. Redes como el FOSS se construyen de acuerdo a las elecciones o a las estrategias de los actores sociales y responden al fenómeno del individualismo (Beck, 1986). Pero tal como lo enuncia Castells (2001) se trata de un individualismo en red donde aquellos que tienen la opción de conectarse buscan y se incorporan a redes o incluso las crean de acuerdo a sus preferencias e intereses como pudimos ver en los incentivos que

¹⁷Tal como lo señala Ostrom (1990, p. 66) no debemos confundir dentro de los RUC, lo que sería el sistema de recursos comunes y el flujo de las unidades de recursos producidas por el sistema.

motivan a los cooperantes. Es algo que se elige estar o no conectado no en el que se nace como en el tipo del ideal comunitario.

Tal como lo mencioné la red se configura a partir de sus nodos (miembros) y las relaciones entre éstos. Hay nodos mucho más interconectados que otros en las redes FOSS como es el caso de desarrolladores como Torvals en el caso Linux, Brian Behlendorf en Apache y Blake Ross para el caso de Mozilla quiénes además de haber asumido costes iniciales que permitieron comenzar el proyecto muchas de las decisiones más trascendentales del desarrollo del proyecto les son consultadas. A medida que la red creció, surgió la necesidad de que ciertas competencias para tomar decisiones se descentralizaran. No obstante, ellos como nodos siguen siendo un punto de atracción en los que la red confluye. En esa medida se puede afirmar que hay nodos más importantes que otros. No es lo mismo ser un nodo poco interconectado ubicado hacia la periferia de la red que uno de los ejemplos mencionados (Ver estudio de Valverde y Solé, 2006)

Personajes como los mencionados ejercen una jefatura en las organizaciones FOSS, pero es un tipo de jefatura que no es resultado de un cargo prediseñado, son reconocidos líderes gracias a las relaciones que presentan en las redes. Su influencia está en proporción de los múltiples vínculos que tienen en la red, es el grupo quién los reconoce como líderes¹⁸ consecuencia del papel que han jugado dentro de los proyectos. Ellos pueden sugerir pero como tal no mandar ya que por lo menos el 50% de voluntarios elige en que desea trabajar o que aporte quiere hacer.

El trabajo individual voluntario presenta otra consecuencia importante dentro de las comunidades. El que cada quien decida lo que quiere hacer y no sea algo

¹⁸ A Linus Torval por ejemplo se le da el calificativo de un dictador benevolente, de pronto no es el calificativo más afortunado pero se le reconocen sus atributos de líder no autoritario que con su forma de ser ha atraído a más cooperantes al proyecto.

impuesto permite que la persona adecuada tome la tarea o labor que considera puede hacer mejor. Es un proceso de autoselección de tareas en el cual es más probable que el individuo sepa que es lo mejor que sabe hacer. A medida que el proyecto crece y desarrolla mecanismos que identifican las contribuciones de menor calidad, las comunidades tendrán una mayor probabilidad de conectar a la gente más competente para la tarea correspondiente en comparación a lo que podrá hacer una corporación tradicional que depende del departamento de recursos humanos para buscar los talentos. En el FOSS y en especial en las comunidades exitosas el talento no se busca, llega a la comunidad.

A pesar de que el aporte de un individuo analizado aisladamente sea tan sólo una parte o un módulo o aplicación puntual en conjunto el sistema como un todo resulta más que la suma de las partes. Es el poder del individuo en red que en interacción puede generar grandes efectos. Es el planteamiento que se le atribuye a Robert Metcalfe conocido como el efecto red, término que se utiliza más en el ámbito de las telecomunicaciones y los negocios e implica que el valor de la red está en relación a la cantidad y al potencial de usuarios o nodos que presenta.¹⁹ El efecto consiste que cuando un nuevo usuario o desarrollador decide vincularse con el proyecto estará beneficiando a los que ya se encuentran en el mismo en cuanto: 1. hacen que el proyecto tienda a estandarizarse y a desarrollar una economía de escala 2. al presentar más colaboradores las mejoras del mismo serán más probables, 3. las posibilidades de comunicación entre los miembros serán más robustas en cuanto se presentarán múltiples interconexiones para conectarse. 4. el producto en sí será más útil consecuencia de haber más individuos utilizando un sistema 100% compatible.

Que Olson no hubiese conocido redes como las FOSS hicieron que considerara excepcional que en ciertos tipos de bienes la función de beneficios aumentara y la

¹⁹ El valor de una red será igual al cuadrado de los nodos que presente, consecuencia de la posibilidad de relaciones que se pueden establecer entre los mismos.

de costos disminuyera a medida que más usuarios entraban al sistema después de haber alcanzado una masa crítica. Es un tipo de efecto positivo dónde a más usuarios mayor es el beneficio y para el caso estudiado no necesariamente un mayor coste en cuanto una vez producido el bien al ser éste no rival (como en el caso de los bienes públicos puros) no he de incurrir en el coste de volverlo a producir. Para los tipos de situaciones con bienes rivales que él estudiaba obviamente una situación de estas características no tenía sentido

5. Características organizativas

5.1. *¿Cómo lograr la coordinación y coherencia del producto?*

Los medios de comunicación son vitales para la gestión exitosa de un RUC pero para el caso de estudio no sólo basta con tener los medios sino que hay que tener en cuenta el tipo de organización que establecemos para lograr la coordinación y mantener la coherencia y calidad del resultado que permita en este caso que el software funcione correctamente. Contrario a lo que sucede con otros desarrollos colectivos en el espacio de Internet como puede ser una Wiki, el software es diferente.

Un producto como Linux, Apache o el FireFox de Mozilla son desarrollos de una altísima complejidad entendida ésta en cuanto los programas están compuestos por miles de líneas de código que configuran instrucciones de lo que debe hacer o no el programa y como debe actuar de manera coherente sin entrar en bloqueos. Piensen sólo un momento en su sistema operativo, no necesariamente tiene que ser libre, cada vez que ustedes dan un clic al mouse o a una tecla hay una instrucción que permite realizar la función que ustedes desean. Piensen en cada una de las teclas, en la ventanas que aparecen en la pantalla de su ordenador, para que todas funcionen tienen asignadas instrucciones, si quiere incluya también las instrucciones resultado de las combinaciones de teclas lo que implica aún más líneas de código. Además de eso, el programa debe brindar la opción de instalarse en distintos ordenadores no sólo en el modelo que usted tiene en su escritorio o empresa. El software constantemente debe estar adaptándose acorde a las nuevas tareas que el usuario requiere y a los nuevos equipos que se desarrollan, esto hace que el software se considere algo tan complejo y un desafío para la acción colectiva ante la coordinación y coherencia que requiere su creación.

¿Pero entonces cómo una cantidad de personas logran coordinarse para lograr un producto de éste nivel de complejidad? Una visión clásica nos diría que es necesario un jefe que como en el caso de un ejército prusiano, el general asigne funciones a sus subalternos y determine que debe hacer cada uno fijando unos lineamientos que se deben seguir. De acuerdo a esta postura una organización de estas características solo sería posible si existiera una dirección totalmente centralizada donde una mente o tan sólo unas pocas determinasen que es lo que debe hacerse, la metáfora de la Catedral desarrollada por Raymond (1999) dónde un ingeniero o arquitecto tiene en la mente el proyecto y él dibuja los planos e indica como deben desarrollarse. Sin embargo a medida que la organización crece los problemas aumentan y los costes de organización también. Es lo que se conoce en el ámbito de la informática como la ley de Brooks que consiste en que tú puedes aumentar el número de programadores en un proyecto y el desarrollo del producto aumentará en una escala lineal pero los errores y vulnerabilidad del sistema aumentarán en una escala geométrica consecuencia de los problemas de comunicación y de interoperabilidad entre las diferentes partes desarrolladas por los programadores. El símil es el juego del teléfono roto en el que se decide enviar un mensaje en el inicio de la cadena humana y a medida que se transmite de una persona a otra el contenido del mensaje se va tergiversando. Son los costes que tienes que estar dispuesto a asumir ante el crecimiento del sistema.

Las organizaciones FOSS obviamente no han sido ajenas a este tipo de situaciones pero a diferencia de los ejemplos mencionados en los casos FOSS no se presenta una planeación centralizada de asignación de tareas, cada miembro de la red es libre de dedicarse a lo que le despierte mayor interés (obviamente exceptuando los que son contratados para desarrollar una tarea específica). La visión clásica que sólo el sabio o el estudiado era la voz a escuchar es totalmente revaluada. Los aportes pueden venir de cualquier colaborador lo que en un primer momento podría hacernos pensar que el desorden sería mayor y por lo tanto la probabilidad de un aumento exponencial de los errores.

Eric Raymond (1999) acuñó lo que se conoce como Linux Law. Ésta consiste en que “al tener múltiples ojos observando todos los errores parecen superficiales”. El que las contribuciones individuales tengan lugar en un contexto que permita ponerlas en consideración de otros, criticarlas, mejorarlas, aprender de ellas y replicar el proceso produce un ciclo de retroalimentación que hace posible aumentar, corregir y volver otra vez sobre el proceso corrigiendo los posibles errores que presente la versión inicial. En la medida que los miembros de la red aumentan, los pequeños o grandes aportes al sistema permiten alterar el estado macro de todo el sistema. Sin este tipo de aportes en diferentes niveles, desde el usuario no programador que reporta un problema hasta el desarrollador fundador del proyecto, la red no sería capaz de encontrar nuevas fuentes de innovación de mejora y la posibilidad de adaptar el producto según los requerimientos de los usuarios y en general del entorno. Si bien una pequeña contribución a nivel individual puede ser insignificante y con poco efecto, en una red gracias a la retroalimentación puede llegar a amplificarse. El efecto red pareciera que permitiera ayudar a superar el planteamiento de Brooks. Pero así como se amplifican las contribuciones convenientes pueden también amplificarse las perjudiciales para el sistema.

5.2. ¿Si todos opinan diferente en relación a lo que observan como ponerlos de acuerdo?

En las redes FOSS todos tiene derecho a opinar y hacer sugerencias lo que no implica que todos tengan derecho a decidir. Así como observamos que los RUC tradicionales exitosos están organizados en entidades anidadas (Ostrom, 2000:148) los proyectos FOSS se organizan en múltiples niveles (Jensen y Scacchi, 2008) lo que nos podría parecer similar al planteamiento de Olson de estructuras federales. No obstante una entidad federal per se no es multinivel que es lo que encontramos en los proyectos FOSS. El software está organizado en lo

que se conoce como módulos que son una colección de líneas de código que en sí mismos crean un paquete coherente de aplicaciones. Cada módulo realiza tareas o funciones específicas y están bajo responsabilidad de un coordinador elegido por los miembros de la red bajo criterios meritocráticos. Los desarrolladores más competentes y antiguos que se identifican con los valores de la organización tienen el honor así como el compromiso de coordinar el desarrollo de los módulos que implican ciertos tipos de funciones. El coordinador aprueba el trabajo de los denominados pares (peer) quienes son los que asumen la decisión de contribuir con el desarrollo del módulo y el coordinador los acepta para el mantenimiento del mismo. Los cambios que sugieran los pares han de pasar por la aprobación de los coordinadores. Cuando el proyecto crece el mismo criterio se aplica para asignar nuevos coordinadores y pares.

Los problemas que se presentan se reportan en las listas de mails, en los foros y wikis. Se guarda un registro de todas las comunicaciones y de las soluciones sugeridas. Las modificaciones realizadas al código junto con las anteriores versiones también son archivadas y se aplica un seguimiento de los cambios que se le realizan al programa. En cualquier momento quién lo desee puede consultar una comunicación, una antigua modificación o propuesta de solución o decidir ponerse a desarrollar una de las versiones del programa. Todas estas bibliotecas facilitan el trabajo de los nuevos desarrolladores que quieren conocer el proyecto, así como a los antiguos que cuando se les presente una dificultad técnica similar a la de un problema del pasado podrán consultarlas.

La forma como está organizada la red social u organización FOSS se ve reflejada en la estructura misma del software que se desarrolla. Es lo que se conoce como la *Ley de Conway* (el programador Melvin Conway en 1968 la formuló). En otras palabras, el diseño del producto final es reflejo de la configuración y mecanismos de comunicación de la organización que lo desarrolló. Tanto la organización como el software es resultado de la suma de pequeñas unidades de trabajo, lo que se

pretende es una alta modularización de los procesos lo que implica en qué medida el software puede ser dividido en pequeños componentes o módulos que puedan producirse de manera independiente antes de fusionarse como un todo (Benkler 2006). El software más que ser una única herramienta es el resultado de la unión de múltiples herramientas (los módulos), una gran caja de herramientas que se conectan unas con otras y permiten crear nuevas funciones. En el ámbito informático es lo que se conoce como la filosofía Unix en cuanto fue en este software en el que se innovó con la creación por módulos ante los límites de computabilidad que con los primeros ordenadores se presentaban. Los principios de la organización y desarrollo de Unix aplicados a los casos FOSS son tres (Weber 2004:28):

- Partir de una interfaz universal o de lenguajes comunes
- No importa que el programa módulo haga sólo una cosa, lo importante es que la haga bien.
- Desarrollar programas que sean compatibles para no presentar problemas al fusionarse y trabajar juntos.

La razón para organizar los proyectos en módulos es organizar pequeños sistemas autónomos. El propósito es quitar las posibles interdependencias entre los módulos y que ante el posible fallo de alguno, la estabilidad del sistema como un todo no se ponga en riesgo. Esta característica maximiza la autonomía y la flexibilidad y el tiempo a dedicar por parte del cooperador. Es más fácil desarrollar y corregir un pequeño módulo que unido forma un programa más grande que corregir un macro programa que presente múltiples interdependencias y posibles ciclos internos. Los costes de administrar y organizar estos pequeños módulos son menores que tratar de coordinar y dar coherencia a un programa que fuera un solo bloque de miles de instrucciones que interactúan.

Olson (1992: 75) parte de la premisa que los agentes envueltos son racionales en cuanto para alcanzar los objetivos que se proponen utilizan los medios disponibles más eficientes y eficaces. Es racional buscar ahorrar recursos escasos como el

tiempo y el talento y no tener que dedicarlos en inventar algo que ya existe. En esta línea mientras con el menor tiempo posible pueda obtener lo que considere el mejor resultado por qué no hacerlo. Lo que consideremos mejor es bastante subjetivo. No existe una única respuesta pero tal como lo plantean Weber (2004) y Himanen (1999) podemos hacer el símil de los Hackers con los zorros, astutos y que intentan trabajar lo menos necesario. Si no toca reinventar la rueda, mejor no hacerlo y dedicarse a otros menesteres. El FOSS permite esto, compartir el conocimiento previo y gracias a la estructura modular que presenta permitir diferentes niveles de cooperación de acuerdo a la disponibilidad de tiempo y compromiso que se tenga. Es el criterio de granularidad que desarrolla Benkler (2006) que hace referencia al tamaño de los módulos en términos de tiempo y esfuerzo que los individuos deben invertir en producirlo. El esfuerzo a realizar por los voluntarios puede darse en distintos niveles dónde cada quién se compromete de acuerdo a su disponibilidad de recursos, en términos de tiempo, dinero y conocimientos. Lo que importa es colaborar y la suma de los granos de arena de cada uno permite construir una gran montaña. Se establece una ley de producción en la cual el número de personas que pueden participar en el proyecto es inversamente proporcional al tamaño de la menor contribución necesaria para desarrollar un módulo que sea funcional.

Si la gente decide organizarse es porque considera que obtiene ventajas de la especialización al ahorrar recursos. Los desarrolladores son mas productivos si no se cambian continuamente de trabajo y se concentran en actividades en las cuales tienen ventajas relativas lo cual corresponde con el hecho que el mayor porcentaje de desarrolladores sólo colaboran con uno o máximo dos proyectos afines (David, Waterman y Arora, 2003)

5.3. Cómo filtrar el ruido.

Pero no he dado respuesta al problema de como clasificar la información y conocimiento que se considera relevante y qué es descartado o considerado basura o ruido. Esta es la razón que hace necesario monitorear el proceso y aplicar mecanismos de resolución de posibles conflictos o diferencias que permitan mantener la coherencia y forma del proyecto. Por ejemplo en el caso de los productos de Mozilla, Niton (2008) clasifica en diferentes fases el proceso de desarrollo del producto. En un primer momento se da la fase que él denomina como “caos”. En espacios como el boletín de reportes o el espacio de noticias del grupo o Wiki cualquiera que tenga Internet puede acceder y reportar un problema o posible error que presente el software, también están las listas de mails de desarrolladores para el caso Linux y Apache pero éstas son exclusivas tal como su nombre lo dice para que desarrolladores reporten los problemas o sugerencias de un cambio en el código. Tanto en las listas como en los boletines se identifica la identidad de quién envía el reporte, el área a la que pertenece el problema, además de categorizar la gravedad del inconveniente. Este registro es archivado y permite hacer un seguimiento de los cambios sugeridos, de los problemas presentados y de quién hizo cierto aporte.

En el caso Mozilla el que decide qué sugerencia aprobar es el coordinador y en el caso de Apache los desarrolladores centrales (*core developers*) (Mockus, Fielding, Herbsleb , 2002) o de Linux el equivalente a los coordinadores (committers) de Mozilla. En la eventualidad que el par no esté de acuerdo con el coordinador puede recurrir la decisión a un segundo nivel o instancia conformada por un programador más experimentado que evalúa la relevancia del requerimiento. Cualquier cambio en el código usualmente requiere la evaluación en distintos niveles hasta llegar al grupo central de desarrolladores que son los que deciden la aprobación final. Ésta es la fase denominada orden que permite mantener un proceso coherente y administrar el “caos”. En algunas situaciones extraordinarias y según la organización como es el caso de Apache ciertos cambios que se consideran trascendentales son aprobados por votación de todos los

desarrolladores donde es necesaria una mayoría simple que expresa su voto vía mail (Mockus, Fielding and Herbsleb, 2002). Para este caso no existen tantas revisiones de segunda instancia en comparación a Mozilla. El núcleo de desarrolladores discute los cambios, los prueba en sus servidores y después de contrastar argumentos y comparar el desempeño de las diferentes soluciones propuestas votan y deciden incluir la que consideran más conveniente para el programa.

Para evitar inestabilidad en el software y que un posible cambio aprobado pueda desencadenar problemas no previstos, el software maneja diferentes versiones. Una estándar que ya ha sido probada en distintos escenarios y se caracteriza por su estabilidad y una versión beta o de prueba en la que los usuarios, tal vez menos comprometidos en el sentido que no programan ni pueden modificar el código, juegan un papel crucial. Ellos son los encargados de reportar todos los posibles inconvenientes que el sistema presente, son ellos sumados a los otros tipos de cooperantes los que hacen el papel de vigías y permiten identificar los errores de una manera más diligente en comparación de si se le encargara esta misión a tan sólo unos cuantos desarrolladores. La red como un todo desde los cooperantes más comprometidos hasta los menos en términos de contribución al código juega un papel crucial en el desarrollo del software, lo que hace que no sea correcto calificar como free rider a los que simplemente descargan el producto como simples usuarios y no saben o no les interesa programar. El que utilicen el software además de promover una mayor influencia del mismo por el efecto red que mencionamos y el estándar que se favorece, hace que sean los principales probadores (testers) del producto.

6. Características Institucionales y Reglas

6.1. Tipos de reglas.

El neoinstitucionalismo al estudiar la acción colectiva pretende encontrar las características institucionales que evitan o reducen la proliferación de subgrupos de beneficiarios que puedan destruir la organización al no contribuir con la misma (Gorronear-free-riding). Su objetivo es conocer las causas que permitan evitar la proliferación de grupos o individuos que deciden maximizar en el corto plazo a costa de pérdidas mayores en el largo. Se busca encontrar instituciones que favorezcan la cooperación lo que no siempre puede ser algo positivo para el conjunto social. En el caso del estudio propuesto espero que así sea pero vale la pena aclarar que es posible identificar organizaciones e instituciones mafiosas que son altamente cooperativas pero no por ello socialmente convenientes²⁰. Aunque en ese caso podríamos argumentar desde la perspectiva de Olson que son los incentivos selectivos²¹ de carácter negativo que recaen sobre los individuos los que disuaden a los miembros del colectivo de cualquier intento de negarse a cooperar.

Para contrastar la visión pesimista de Olson y Hardin retomo lo que sería la segunda generación de la acción colectiva dentro del neoinstitucionalismo. Las instituciones las entiendo como los usos, hábitos, costumbres y normas por los que se rigen las relaciones sociales y económicas entre los miembros del grupo.

²⁰ La valoración de lo que se considera socialmente conveniente o benéfico también depende de la respuesta que se dé a la siguiente pregunta. ¿A quiénes benefician o perjudican las instituciones y las organizaciones que estudiamos? Para el grupo mismo el tipo de instituciones pueden ser altamente benéficas en cuanto promueven la cooperación. Para el conjunto social asumo que no serían convenientes organizaciones que restringen la posibilidad de elección, utilizan la coerción y coacción para obtener sus objetivos. En este sentido evitar la existencia de instituciones mafiosas sería altamente benéfico para el conjunto social.

²¹En este caso se imponen unos costos de deserción tan altos que los miembros prácticamente se ven obligados a ser leales a la organización.

Son el conjunto de reglas formales e informales que estructuran la interacción social; las reglas establecen las restricciones de lo que se puede y no se puede hacer, de lo que se debe y no debe hacerse en determinadas situaciones de la interacción humana (Ostrom, Hess 2007:42). Las reglas son los códigos de conducta que definen las prácticas sociales, asignan los roles de los participantes en estas prácticas y guían las interacciones de los agentes en esos roles (Young, 1995, en Ostrom y Keohane, 1995:33).

Cuando estas instituciones normativas son meramente escritas como parte de un código o compendio administrativo o legal pero no necesariamente son conocidas por los participantes o no los obligan son las denominadas reglas de forma. Las reglas en uso por el contrario son conocidas y comprometen al grupo son de conocimiento común lo que implica que cada agente conoce las reglas y sabe que los otros las conocen. Estas últimas generan oportunidades como limitaciones a aquellos que interactúan.

Las reglas pueden ser de tipo regulativo o constitutivo (Searle, 1997). Las primeras me configuran qué tengo permitido hacer en el juego y qué papel tengo en él. Las constitutivas me definen el juego mismo, cambiarlas implicaría estar jugando otra cosa. Por ejemplo mientras que para el football (soccer) yo podría considerar que no haya once jugadores sino trece, lo que sería una regla regulativa y no me cambiaría el juego, si se decidiera que ahora se puede jugar cogiendo el balón con la mano estaría afectando una regla constitutiva que me cambiaría el juego mismo.

Las reglas regulativas deben ser flexibles y más en un campo en el que se presentan tantos cambios es necesario que lo sean. Si se aspira a desarrollar instituciones duraderas el poder de adaptación acorde a los nuevos contextos y avances de la tecnología es vital. Entre las reglas regulativas están por ejemplo quiénes tienen autorizado modificar el código, quiénes pueden hacer sugerencias

o reportar problemas, qué medios se privilegian para reportar los cambios, quién y cómo se eligen a los desarrolladores centrales y cómo se castiga a los que infringen las reglas.

A diferencia de lo que sucede en una empresa, no es posible despedir al que no cumple las reglas. La excepción sería para aquellos que trabajan en la plantilla de empleados de la organización. Pero entonces ¿qué hacer frente al comportamiento no apropiado de algún usuario activo o desarrollador voluntario? El grupo juega un rol de control fundamental, primero porque es el que determina hasta donde están permitidas las disputas y críticas de carácter ideológico entre los miembros tal como cita Weber (2004:176). Cuando las críticas son acerca de los valores y de hacia dónde debe dirigirse el grupo algunas discusiones pueden llegar a ser muy fuertes pero como todo queda registrado, sirve de ilustración para los nuevos desarrolladores y para que la comunidad tenga argumentos de juicio en relación al proceder de cada uno de los involucrados en la posible disputa y para poder hacer un pronunciamiento de condena pública.

En lo que corresponde a las diferencias técnicas parece darse un acuerdo tácito que las posibles críticas no serán sobre éste campo, en este ámbito la resolución será dada por la calidad del código que se presente, los filtros técnicos que tiene la organización decidirán pero no se permite descalificar a otro por sus contribuciones técnicas. Esto garantiza que las personas sigan estando incentivadas a proponer soluciones que es parte de la fortaleza organizativa del FOSS. Usualmente se privilegia la primera solución que llegue y funcione adecuadamente pero lo que más importa es la superioridad técnica que la comunidad ratifica o descarta cuando prueba las versiones beta del programa.

Los castigos más severos son excepcionales y además difíciles de aplicar pero se presenta lo que se conoce como “Shunning” (Weber: 177) en que el grupo sanciona con una especie de castigo de silencio a aquél que rompe la norma.

Como son sistemas de acceso abierto es imposible impedir que siga utilizando el bien pero no contará con el apoyo del grupo ante cualquier problema que presente y tendrá que aprender solo lo que puede resultar muy costoso. Otro de los posibles castigos es vetarlo de su condición de jefe de módulo o de división si es el caso, esto implica que la comunidad ya no reconoce su autoridad ni los aportes que haga. Siempre existe la opción de hacer forking pero si no se cuenta con apoyo no sirve de mucho, sería un usuario más.

En cuanto las reglas constitutivas está la libertad del código. Esto es lo que configura el juego. Si el código no fuera libre el FOSS no existiría, podría ser otra cosa pero no FOSS. Libre en el sentido que precisa Stallman de poder conocer, modificar, copiar, adaptar y mejorar la información del programa. Se garantiza el derecho de la libertad de información y de la libertad de hacer forking. Esto último puede parecer absurdo porque lo he tipificado en ciertas circunstancias como free riding pero que exista la alternativa de crear un proyecto propio e intentar convencer a parte de la comunidad original que apoye una perspectiva diferente a la del proyecto madre es una garantía para que los diferentes cooperantes que hacen parte de un proyecto decidan pertenecer al mismo no porque no hay más alternativas sino porque lo desean. La opción de independizarse existe y no es vetada, nadie te obliga a estar donde no quieres, convence a los demás que tu proyecto es atractivo y crea tu propia comunidad, que sea sencillo ya es otra cuestión.

Recordemos que para el caso FOSS las tareas que realiza cada desarrollador no son preasignadas, cada uno de acuerdo a su interés decide a qué se dedica. Estas características hacen que el grupo tenga mecanismos de resolución de conflictos, instancias de discusión y mecanismos de decisión que se consideran legítimos en el sentido de reconocimiento y aceptación por parte del grupo. Toda la organización tal como lo expuse en el apartado de cómo filtrar el ruido es un gran sistema de resolución de conflictos y diferencias. Si se revisan las listas de

mails, los foros y las wikis, el conflicto como en cualquier sistema de decisiones está presente, no siempre las opiniones entre los miembros coinciden pero para eso el grupo asigna competencias y reconoce personas meritorias que se consideran aptas para decidir.

Para garantizar la regla constitutiva de la libertad del código ha sido necesaria la creación de las licencias públicas, en su desarrollo Stallman ha sido un pionero con el copyleft (denominación creada por él como parodia al copyright) y la General Public Licence GPL con lo que se pretende evitar que alguien se apropie del esfuerzo colectivo y desincentive la cooperación. Con este tipo de instrumentos jurídicos la licencia es universal y está al acceso de cualquier persona que desee acceder a la misma respetando los derechos establecidos por el creador original. Derechos que garantizan la libertad de información en cuanto nadie puede restringir el acceso a la misma y cualquiera tiene derecho de modificarla, copiarla y distribuirla.

Cualquiera que desee hacer parte del proyecto tiene que aceptar la licencia. Es un tipo de compromiso público con efectos legales en el marco de los derechos de propiedad intelectual. La propiedad no se configura sobre la exclusividad del manejo y conocimiento de la información sino que se configura en su distribución. (Weber, 2004). Los derechos de alienación y sus contribuciones recaen en todo el grupo, ningún miembro tiene el derecho de vender, arrendar o restringir el acceso al código. El acceso es abierto y al ser un bien no rival contrario a los bienes comunes tradicionales no desencadena en problemas de sustractabilidad y sobreexplotación. Aproximadamente el 85% de todos los productos FOSS utilizan licencias GPL sin embargo, no todas las licencias GPL son iguales, los autores del software pueden elegir entre una amplia variedad aproximadamente cincuenta

tipos que implican la opción que tiene el autor del software de retener o ceder ciertos derechos²².

Mozilla y Apache tienen su propia licencia pero también es de carácter público aunque no incluyen la restricción viral lo que facilita la interacción con software propietario. La restricción viral consiste en que cualquier programa que se desarrolle derivado de un programa con licencia GPL o interactúe con librerías del software en este esquema de licencia debe adoptar el mismo tipo. Esta última condición al ser confusa ha generado la reticencia de empresas de software privado para desarrollar proyectos conjuntos con programas GPL. De todas maneras la licencia pública o el esquema de licencias públicas estilo Mozilla o Apache es una condición fundamental que hace que los desarrolladores sientan confianza para decidir cooperar con el proyecto en el sentido que a nadie le gusta perder su trabajo y que otros usufructúen tu esfuerzo excluyéndote. El tipo de licencia es una garantía que evita que tus aportes no sean reconocidos o que una empresa u otro decidan apropiarse de ellos y cobrar por su uso o consulta.

Por ejemplo en los inicios del proyecto Mozilla, Netscape creó la Netscape Public License (NPL). Sin embargo la compañía estaba interesada en mantener algunos de sus desarrollos bajo esquema privativo y mantenía la opción de relicenciar en esquema privativo parte del código desarrollado bajo este esquema. Netscape consideraba que era un precio justo por el código con el que contribuía para iniciar el proyecto. Para los que no se acuerdan o no habían nacido antes de que el Explorer se vendiera junto con el Windows, Netscape era el software explorador por excelencia. El caso es que la iniciativa inicial de hacer del proyecto de Netscape un proyecto FOSS bajo la licencia NPL fue un fracaso, debido a que los posibles cooperantes no se vincularon al mismo a pesar de la calidad del producto que la empresa ofrecía (Hamerly ,Paquin and Walton, 1999). El que la licencia no

²² Para ver los tipos de licencias consultar <http://www.gnu.org/licenses/license-list.es.html>

fuera transparente y presentara ambigüedades en relación a quién mantenía los derechos de acceso, alienación y exclusión desincentivó la cooperación. Un marco legal apropiado que brinde garantías hacia el futuro de no ir a ser privatizado es un elemento clave a considerar por parte de los posibles cooperantes.

6.2. ¿Qué sucede con la propiedad intelectual?

Contrario a ciertas explicaciones simplistas, el movimiento FOSS no está en contra del reconocimiento de los derechos de autor ni en contra de quienes contribuyen con su trabajo y creatividad en el desarrollo de la ciencia y la tecnología. Tal como lo presento uno de los incentivos principales de los desarrolladores para contribuir con los proyectos es el reconocimiento y los diferentes estudios sociodemográficos del FOSS ratifican esta idea (University of Maastricht, 2007 Sales, 2006 David, Waterman y Arora, 2003)

La oposición es en contra de los derechos de propiedad intelectual que sustentan su razón de ser en la exclusión y el monopolio. Es cierto que la propiedad privada se sustenta en el derecho de exclusión pero para el caso del avance de la ciencia y la tecnología este derecho se convierte en un obstáculo. El conocimiento tal como lo planteaba Newton se da gracias a que “logramos ver más lejos porque estamos sobre hombros de gigantes”, si cobramos por acceder a la información incrementamos los costes de producción del mismo lo cual en un contexto donde los productos de información requieren de múltiples fuentes y desarrollos previos hace muy costoso crear cualquier producto nuevo.

Es lo que Michael A. Heller and Rebecca S. Eisenberg (1998) denominan la tragedia de los anticomunes que consiste en que la gente subutiliza recursos escasos como puede ser el conocimiento en cuanto el que es poseedor de los derechos de propiedad bloquea a alguien que requiere este conocimiento. El bloqueo se puede dar mediante los costes que le cobra por transferir el

conocimiento o simplemente por mecanismos legales y técnicos que hacen que sólo uno tenga el monopolio.

Productos de software como los estudiados son de un alto nivel de complejidad en el sentido que son resultado de la interacción de múltiples procedimientos, aplicaciones y algoritmos que en interacción dan como resultado el producto que los usuarios utilizan. Ninguno de estos productos es fruto del trabajo de una sola persona. A pesar de que los niveles de contribución varían no se podría tener lo que se tiene hoy sino fuera gracias a la contribución colectiva y a la posibilidad de acceder a la información incorporando un amplio espectro de conocimiento reflejado en aplicaciones previas. Tener que pagar por cada uno de las aplicaciones desarrolladas en cuanto se encuentran protegidas por marcos regulatorios privativos o por mecanismos técnicos no habría permitido el desarrollo de la investigación o lo hubiese limitado tan sólo a aquellos inversionistas o empresas que tienen el capital y el oligopolio de las patentes. En lugar de buscar premiar a los innovadores, en este contexto los derechos de propiedad intelectual tradicionales ante el tipo de producto que se está desarrollando serían una barrera al desarrollo de la investigación al proteger monopolios y concentrar las posibilidades de desarrollo de software en unas abadías al estilo medieval, las transnacionales y los programas de investigación de los Estados más ricos serían los únicos con la capacidad de asumir los costes para producir nuevo conocimiento por lo menos bajo el esquema de derechos de propiedad intelectual tradicionales. Las sociedades sin dinero pero que quisieran desarrollar tecnologías informáticas se verían obligadas a comenzar de cero y reinventar la rueda si quisieran intentar mejorar lo ya existente, procedimiento que tomaría años y los dejaría rezagados tecnológicamente. La alternativa para evitar esta situación parecería ser alejarse de los esquemas tradicionales de propiedad intelectual.

Para Benkler (2006) retomando la tesis de Arrow (1962) cobrar por la información es crear ineficiencias en el mercado. El coste marginal después de producir la

información es cero. Es decir una vez haya escrito esta tesina cada vez que alguien la lea no deberé escribir un nuevo texto. Contrario a lo que sucede con un coche o con un plato de comida o bienes de la economía industrial que cada vez que requiero un producto para un nuevo usuario incurro en un coste de producción. Si recordamos que el precio va de la mano del coste marginal del producto cobrar más que ello generaría mercados ineficientes. Ante este argumento resulta paradójico que ciertas corporaciones y gobiernos que tanto reivindican la liberalización de mercados y de competencia no se pronuncien de igual manera ante la ineficiencia de cobrar por la información.

No obstante, en ciertos casos podríamos aceptar cierto grado de ineficiencia en el mercado bajo el argumento de incentivar el ejercicio creativo. Pero al estudiar el caso FOSS surge la duda de si realmente es necesario. Nos tendríamos que poner de acuerdo en el grado de ineficiencia que debemos aceptar, qué monto y en qué casos se debe pagar sin ir a limitar la producción del conocimiento y no caer en la tragedia de los anticomunes e ir a limitar el acceso a la información y tan sólo favorecer a ciertas corporaciones que sustentan sus ingresos a través de una restrictiva política de derechos de propiedad intelectual.

Conclusiones

No podemos desconocer que el proceso de producción del FOSS definitivamente tiene consecuencias sociales importantes, Weber (2004) considera el proceso del desarrollo del FOSS el equivalente a la revolución Fordista en el capitalismo, Raymond (1999) nos habla de la *economía gift-to gift* como una nueva economía. El tiempo y los eventos que en el futuro sucedan les darán o no la razón. Lo que es claro es que se cuestionan condiciones clásicas de los RUC tradicionales. Primero porque estamos gestionando un tipo de bien diferente que no comparte los problemas de rivalidad y segundo porque no presenta los problemas de sobreexplotación y congestión, que más usuarios dispongan del bien potencializa su utilidad consecuencia del efecto red. El dilema social para este caso está en la provisión y mantenimiento del bien, no sólo interesa que haya usuarios lo que sería un incentivo importante por el estándar que se genera y el posible mercado para que empresas decidan contribuir en su desarrollo, también es esencial que haya *core developers* y en general desarrolladores que contribuyan en la continua actualización y mejora del recurso.

Para favorecer la provisión y mantenimiento de un proyecto FOSS y responder a la pregunta central de mi proyecto acerca de cómo mantener la cooperación y coordinación a medida que el grupo crece, formulo una serie de principios en que los niveles de cooperación y coordinación estarían en función de:

CC: $f(LI, Nm, , Ic, Mca, Isc, Im, Tg)$

- **CC: Cooperación-Coordinación en las organizaciones FOSS estarían en función (f) de:**
- LI: libertad de licencias. Qué margen de maniobra permite la licencia a los miembros del proyecto, si es clara o no en las garantías que ofrece a los que contribuyen al proyecto y qué les permite hacer.

- Nivel de modularización: Qué está directamente relacionado con la granularidad en relación al esfuerzo o dedicación requerida por el individuo para obtener el beneficio. Entre más tareas y funciones involucre un módulo el nivel de esfuerzo requerido para su correcto funcionamiento y seguimiento será mayor. El que los módulos presenten cierto tamaño y estén especializados en ciertas tareas es una virtud para poder hacer manejable el proyecto como un todo. Es mejor un sistema densamente interconectado con módulos simples en el que unos pocos o muchos desarrolladores dedican el tiempo que puedan al mismo y puedan ponerlo a punto que sistemas complicados que tengan problemas cómo la dificultad de coherencia ante el sinnúmero de tareas que se propone y relaciones de interdependencia que presentan.
- Ic: Infraestructura de comunicación como el Internet y la Web al ofrecer la opción de comunicaciones deslocalizadas geográficamente y de manera asincrónica. La conexión y desconexión a la red promueve las interacciones causales y trabajo en paralelo esencial para poder considerar múltiples soluciones a un mismo problema.
- Mca: Medios de comunicación y archivo que me permiten elegir el momento de la respuesta que puede ser prácticamente instantánea como en el caso de los chats o cuando el receptor decida o cuando pueda revisar el mensaje en el caso de los mails, forum o espacios de noticias. También están todos los mecanismos de archivo, los documentos con los que cuenta el proyecto, los sistemas de control de cambios que permiten a los nuevos programadores aprender el proceso y a los antiguos tener un registro de los cambios realizados y de las soluciones propuestas en situaciones similares.
- Isc: Incentivos socioculturales como el desafío intelectual que implica el proyecto y los valores que reivindica la organización con los que el desarrollador se identifica.
- Im: Incentivos materiales como las recompensas económicas, el tiempo y esfuerzo ahorrado por contribuir con el proyecto, poder contar con software de

calidad y que se pueda modificar de acuerdo a las propias necesidades o requerimientos y adquirir experiencia.

- Tg: Tamaño del grupo, tal como lo plantea Schelling (1978) es necesario una masa crítica que favorezca la cooperación, alguien o algunos que estén dispuestos a asumir los costes iniciales. La naturaleza estadística de la interacción entre miembros de la red requiere de una masa crítica para que el proyecto sea sostenible y sea capaz de autodesarrollarse. Un número reducido de miembros de una red podrán hacer su mejor esfuerzo, asumir unos costes iniciales altos al contribuir en el desarrollo de una primera fase del proyecto, pero si su esfuerzo no tiene eco, no despierta el interés de los usuarios y potenciales desarrolladores será muy difícil su sostenimiento. A medida que se incorporen nuevos voluntarios, el proyecto adquirirá una dinámica propia, las iniciativas provienen de diferentes partes, varias cabezas piensan más que una o que un reducido número, al tener más usuarios es más fácil identificar los errores del proyecto, recibir sugerencias y hacer un proceso de retroalimentación (Ley de Linux). Los grupos con un conjunto de desarrolladores centrales más grande tienden a atraer más voluntarios tal como lo plantea el estudio de Schweik , Charles M., Robert C. English, Meelis Kitsing , Sandra Haire (2008).

Que estas características sean aplicables a otros contextos diferentes al FOSS hace parte de la investigación en proceso. Ya podemos encontrar referencias al Open Politics, Open Science y manifestaciones que parecieran aplicar el proceso de desarrollo del FOSS en otros escenarios. Igual considero que debemos ser prudentes, es tal la diversidad en las características de los considerados nuevos bienes comunes que como Elinor Ostrom (2003b: 242) señala intentar una única teoría de acuerdo a la inspiración de Olson no tiene sentido. Puede ser más adecuado plantearse la necesidad de disponer de una familia de teorías relacionadas en la que el resultado en los estudios del FOSS podría ser una de éstas.

En relación a condiciones como la necesidad de límites geográficos claramente definidos, en el ciberespacio y por el bien con el que estamos tratando no aplica. En referencia a los otros principios formulados por Ostrom para instituciones exitosas en sentido general podemos decir que se aplican para el caso FOSS. El grupo como un todo juega un papel activo en la administración del recurso aprovechando las nuevas tecnologías que le han permitido desarrollar sus propios procesos de comunicación y control. Ésto les facilita hacer un seguimiento de los cambios y problemas que presenta el producto. En relación a quiénes tienen derecho a ejercer las labores de supervisión tal como lo menciono es un criterio meritocrático. También se presenta una organización de carácter similar a la propuesta de la estructura anidada que formula Ostrom permitiendo el control de los cambios en el código en diferentes niveles.

En lo que se refiere a mecanismos de resolución de conflictos, la organización como un todo es un proceso constante de resolución de diferencias para lograr la coherencia del producto y sancionar a los que no respetan las reglas. No existe un ente aparte a la comunidad para resolver los conflictos y las diferencias. Es la comunidad en la forma como está organizada y cómo se comunica la que va dirimiendo los conflictos que surgen. Las sanciones son fundamentalmente de tipo moral. Cuando las diferencias son de carácter técnico se adopta la solución que demuestre el mejor desempeño, sea la más sencilla (en términos de líneas de código) y llegue primero. Las discusiones sobre estos temas también se dan en espacios como las listas de mails, los foros y los archivos de control de cambios.

La existencia de las licencias públicas y el libre acceso y salida de los proyectos incluso con la opción de hacer forking es el mayor reconocimiento de derechos a los miembros de la organización, es la libertad de no estar coaccionado por autoridades externas en relación a qué puedo o qué no puedo hacer. La opción de irse siempre está presente, yo decido en qué quiero trabajar, la obligación no es

impuesta, yo como colaborador soy el que asumo el compromiso. Es un incentivo sociocultural que contrario a lo que decía Olson también se da en grupos grandes.

Es evidente que las nuevas tecnologías facilitan el trabajo en red pero la tecnología por sí sola no basta. Lo que ha permitido el FOSS es la organización política entendida como el espacio que los humanos creamos para negociar diferencias y conflictos y donde sus miembros como creadores de la organización fundan nuevos arreglos institucionales que permiten alcanzar un propósito común, en este caso el desarrollo del software. Como mencioné para los casos estudiados un producto de tal complejidad hace que la labor de un solo hombre no sea suficiente y sea necesaria la acción colectiva.

Para finalizar, el debate sobre propiedad intelectual me implicaría una nueva tesis doctoral pero lo que considero importante es preguntarnos si un esquema de propiedad que parte del supuesto de exclusión y desarrollado para bienes que presentaban características como la sustractibilidad es el más apropiado para los tipos de bienes antirivales (Weber,2004) que produce el FOSS. Este es un tema que deseo explorar más a fondo y que presento en el esquema por desarrollar.

Considero pertinente preguntarnos si ¿Ciertas corporaciones y gobiernos no deberían pensar mejor en reconfigurar su negocio y darse cuenta que el contexto tecnológico y social es diferente? Existen ejemplos como IBM que su principal fuente de ingresos ha dejado de ser las patentes, siendo el mayor productor de las mismas en Estados Unidos pero también es una de las empresas que más apoya el desarrollo de software de código abierto. El negocio está en el servicio, en el soporte que ofrezco, esto lo ha entendido IBM, Nokia, Sun,.. Además de darse cuenta que si desean mantenerse a la vanguardia es necesario compartir, muchas cabezas en red y bajo el adecuado esquema organizativo serán más innovadoras que un grupo de creativos que como empresas ellas puedan contratar.

Las organizaciones que ofrezcan el servicio seguirán existiendo en cuanto aprender implica un coste en términos de tiempo, esfuerzo y actividades que dejamos de realizar. Puede interesarme aprender a programar y creo tener las capacidades pero no tengo el tiempo ni la paciencia para hacerlo, prefiero contratar a alguien que lo haga por mi, que me arregle el ordenador cuando deja de funcionar y me instale el software acorde a mis necesidades. Obviamente el servicio vale, la información esta disponible en la Web pero aprender toma tiempo y otros recursos, por esto como muchos otros estoy dispuesto a pagar por el conocimiento.

Las organizaciones de la era industrial quieren mantener los esquemas antiguos e imponer las reglas que les convengan en la nueva época que se configura. Espero que no caigamos en la tragedia de los anticomunes y la tecnología con el componente humano que la crea (organizaciones) sea capaz de superar las barreras que se le quieren imponer al conocimiento y a la información. Las instituciones más duraderas en el tiempo tal como lo plantean los estudios de Ostrom (1990, 2005, 2007) son aquellas que se adaptan a las nuevas circunstancias espacio temporales.

BIBLIOGRAFÍA:

Aiyer, Rishab Ghosh y UNU MERIT (2006) Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU.

Anduiza, Eva, Ismael Crespo y Mónica Méndez (1999). Metodología de la Ciencia Política, Madrid: Centro de Investigaciones Sociológicas

Annenberg, Janet Fulk (2004) "Stocking Shared Knowledge Repositories: Motivations Toward Collective Action" School for Communication University of Southern California April 30, MIT Sloan Colloquium

Axelrod, Robert (1986). La evolución de la cooperación Alianza Universidad

Axelrod, Robert (1997) the complexity of cooperation Princeton University press.

Ball, Philip. (2005) Critical Mass How One thing leads to another. Arrow.

Benkler, Yochai. (2006) The Wealth of Networks. How Social Production Transforms Markets and Freedom. Yale university press.

Bloomquist, A. (1992) "Dividing the Waters: Governing Groundwater in Southern California". San Francisco: ICS Press.

Burt, Ronald y Towards, S. (1982) A structural theory of action: Network models Of social structure, perception, and action. New York: Academic Press.

Cahir J. (2004). *The withering away of property: The rise of the internet information commons*. Oxford Journal of Legal Studies. 24: 619-41

Cardenas Juan Camilo. (2001-2003). *Cooperación en Comunidades Rurales* financiado por la Fundación MacArthur (Research and Writing Grants Competition). El estudio incluye la aplicación de experimentos económicos en 10 comunidades rurales de Colombia y cerca de 800 habitantes de estas, además de una validación con cerca de 80 estudiantes de la Universidad de los Andes.

Castells, Manuel, (2001). *La Era de la información: economía, sociedad y cultura* : 3ª ED. Madrid: Alianza, 3 vol. ; 23 cm.

Castells, Manuel, (2001). *La Galaxia Internet*. Arête- Plaza y Janes-

Crowston, Kevin and Howison, James (2004) "The social structure of Free and Open Source software" Syracuse FLOSS research working paper, available at <http://floss.syr.edu>

David Paul A., Andrew H. Waterman y Seema Arora (2003) *Floss-Us The Free/Libre & Open Source Software Survey For*, Stanford University, California, Usa September,

DiBona Chris, Sam Ockman, Mark Stone *Open Sources: Voices from the Open Source Revolution*, Editorial O'Reilly 1999, ISBN: 1-56592-582-3

Echevarria, Javier (1999.) *Los señores del aire: Telépolis y el tercer entorno*. Editorial Destino

Farrell, Joseph John Hayes, Carl Shapiro, and Theresa Sullivan (2007), Standard Setting, Patents, and Hold-Up, <http://faculty.haas.berkeley.edu/shapiro/standards2007>.

Gintis H, Samuel Bowles, Robert Boyd, Ernst Fehr, eds. (2005). Moral Sentiments and Material Interests: The Foundations of Cooperation in Economic Life, Cambridge, Mass.: MIT Press, cop. 404 p

Goodin, Robert. (General Editor) (2007) The Oxford Handbook of comparative politics. "Collective Action Theory. Elinor Ostrom". Oxford University Press. P.186-210

Gosh, R.A., R. Glott y G. Robles. (2002). Free /Libre and Open Source Software: Survey and Study. Technical report. University of Maastricht, the Netherlands: International Institute of Infoeconomics, June. <http://www.infonomics.nl/Floss/report/index.htm>.

Granovetter, Mark (1973). "The strength of weak ties". American Journal of Sociology. N°78 pp. 1360-1380

Hardin, Garrett (1968) "The Tragedy of the Commons," Science, 162 (1968):1243-1248.

Hess, Charlotte. (2008) Mapping the New Commons Presented at "Governing Shared Resources: Connecting Local Experience to Global Challenges;" the 12th Biennial Conference of the International Association for the Study of the Commons, University of Gloucestershire, Cheltenham, England, July 14-18.

Hess, Charlotte and Elinor Ostrom (editors) (2006) Understanding Knowledge as a Commons . "Introduction: An Overview of the Knowledge Commons." MIT Press 381p.

Heller Michael A. and Rebecca S. Eisenberg (1998) "The Tragedy of Anticommons" SCIENCE VOL. 280 1 MAY www.sciencemag.org)

Himanen Pekka, Linux Torvalds , Manuel Castells , MelerOrt F, (2002). La Ética del hacker y el espíritu de la era de la información, Barcelona: Ediciones Destino

Issaac, R. Mark, Walker, James. (1988). "Group Size Effects in Public Goods Provision: The Voluntary Contribution Mechanism," Quarterly Journal of Economics, 53, 1988, pp.179-200,

Janssen, Marco A. Elinor Ostrom. (2007). TURFs in the lab: institutional innovation in real-time dynamic spatial commons.

Jensen, Chris and Walt Scacchi (2008) Governance in Open Source Software Development Projects: A Multi-Level Case Study Analysis. Institute for Software Research University of California, {cjensen, wscacchi}@ics.uci.edu

Kaldor, Mary. (2003). Global Civil Society. An Answer to War. Polity. Blackwell. 189 p.

Kalmanovitz, Salomón (2003) "El Neoinstitucionalismo como Escuela" Revista Economía Institucional, v. 5. n. 9 Bogotá diciembre.

King, Gary, Robert O. Keohane, Sidney Verba (2000) El Diseño de la Investigación Social. La Inferencia Científica en los estudios cualitativos. Ciencias Sociales. Alianza.

Kogut B, Metiu A. (2001). *Open-source software development and distributed innovation*. Oxford review of economic policy. 17:

Kriesi, Hanspeter [et al.] (1995) *New social movements in Western Europe: a comparative analysis*: Minneapolis: University of Minnesota, cop. 310 p

Lannacci F. (2005). "The Social Epistemology of Open Source Networks". Department of Information Systems, London School of Economics and Political Science.

Lakhani, Karim R. and Robert G Wolf. (2005) *Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects*. in *Perspectives on Free and Open Source Software*. MIT press.

Marwell, G. y P. Oliver (1993), *The Critical Mass in Collective Action: A Micro-Social Theory*, Nueva York, Cambridge University Press.

Masuda, Yoneji (1981) *La Sociedad informatizada como sociedad post-industrial* Madrid: Tecnos, Fundesco,

Michlmayr Martin (2007). *Quality Improvement in Volunteer Free and Open Source Software Projects Exploring the Impact of Release Management*. King's College. Centre for Technology Management Institute for Manufacturing University of Cambridge.

Mockus Audris, Fielding Roy, Herbsleb James D (2002) *Two Cases Studies of Open Source Software Development: Apache and Mozilla*. In *Perspectives on Free and Open Source Software*. MIT press. 2005.

Molins, Joaquim (1998), *Teoría de Grupos*, working paper, Universidad Autónoma de Barcelona

Moraes, Denis (Coord). (2007). Sociedad Mediatizada, Gedisa. Septiembre. 197p.

Niton, Tristan(2008), President and Founder Mozilla Europe, How Mozilla Ecosystem Works. Conference Bdigital Global Congress. May 20 th-22 nd.

Neveu, Eric Neveu. (2002). Sociología de los movimientos sociales. Editorial hacer. 190p.

North, Douglass C. (1990). Institutions, Institutional Change, and Economic Performance. New York: Cambridge University Press.

Olson, M. (1992) La lógica de la acción colectiva, bienes públicos y teoría de grupos. Limusa, México,

Olson, M. (1985) Auge y decadencia de las naciones. (32-55) Ariel en Battle Albert (1992) Diez Textos básicos de Ciencia Política. Ariel.

Ophuls, W. (1973) Leviathan or oblivion. In H.E. Daly (ed.), toward a steady state economy. San Francisco: Freeman,

Ostrom, Elionor (2000).El gobierno de los bienes comunes. La evolución de las instituciones de acción colectiva. Universidad Nacional Autónoma de México, Centro Regional de Investigaciones Multidisciplinarias, Fondo de Cultura Económica. (1990 en Ingles)

Ostrom Elionor y Robert O. Keohane (1995). Local Commons and Global Interdependence. Sage publications

Ostrom, E., Burger, J., Field, C.B., Norgaard, R.B., and Policansky, D. (1999). "Revisiting the Commons: Local Lessons, Global Challenges." *Science*. 284: 278-282. (categorías de heterogeneidad)

Ostrom, Elinor T. K. Ahn (2003) Una perspectiva del capital social desde las ciencias sociales: capital social y acción colectiva. Instituto de Investigaciones Sociales. Revista Mexicana de Sociología, año 65, núm. 1, enero-marzo, México, D. F., pp. 155-233. ISSN: 0188-2503/03/06501-03/

Ostrom, Elinor (2003b), "How types of goods and property rights jointly affect collective action", *Journal of Theoretical Politics*, vol. 15, nº 3, pp. 239-270.

Peters Guy (2003) El nuevo institucionalismo: Teoría Institucional en Ciencia Política. Gedisa.

Raymond Eric S. (1999) A Brief History of Hackersdom. Freeing the Source The Story of Mozilla in Open Sources: Voices from the Open Source Revolution. O'Reilly.

Sales, Meritxell Roca (2006) Internet Interdisciplinary Institute, El software libre en Cataluña y en España, UOC, Barcelona julio de

Schelling, (1978) T.C. Schelling, *Micromotives and macrobehavior*, Harvard University Press, Cambridge.

Schweik, Charles M. (2004) Open Source Programming as a Framework for Scientific Collaboration: An Example in the Context of Land-use Change Modeling Prepared for the Workshop on Scholarly Communication as a Commons, Indiana University, Bloomington, March 30-April 2nd.

Schweik , Charles M., Robert C. English, Meelis Kitsing , Sandra Haire (2008) University of Massachusetts Amherst, Amherst, MA "Brooks' versus Linus' law: an empirical test of open source projects" Source ACM International Conference Proceeding Series; Vol. 289 archive Proceedings of the 2008 international conference on Digital government research

Searle, John (1997). La construcción de la realidad Social. Paídos. 236p.

Simon, Herbert. (1986) De la racionalidad sustantiva a la procesal, en Frank Hahn, Martin Hollis, Filosofía y teoría económica, FCE, México.

Stehr, Nico Knowledge societies "One of the first authors to employ the term 'knowledgeable society' fue Robert E. Lane (1966:650)..."

Tarrow, (2005) Sidney, Cornell University. The new Transnational Activism, Cambridge University Press.

Valverde,Sergi Richard V Solé. (2006). Self-organization and hierarchy in open source social networks. Working papers Santafe institute,

Vallès J. (2007). Ciencia política una introducción, Barcelona: Ariel

Von Neumann John y Oskar Morgenstern, (1944) Theory of Games and Economic behavior, Princeton University press.

Wallach Hanna, Moray Allan, Dafydd Harries (2005) the Debian New Maintainer Process: History and Aims. *Hanna Wallach is the maintainer of several Debian packages, and is currently in the New maintainer process. Moray Allan went through the New Maintainer process in 2003, and has been an Application*

Manager since April 2004. Dafydd Harries completed the New Maintainer process in June 2005.

Watts, Duncan J. (2003). Six degrees: The science of a connected age, ed. London: Heinemann, cop, 368 p.

Weber Steven. (2004) The success of open source. Harvard University Press.

Wilson, James Q. (1974) Political Organizations. Edicion Princeton 1995.

Paginas Web de comunidades y bases de datos FOSS

-
- <http://people.apache.org/index.html>
- <http://www.polnetuab.net/resulten.php?pagina=Proyecto&Idioma=English&ipg=01> (Proyecto de investigación participación política e Internet)
- <http://www.gnome.org/press/>
- <http://www.gnu.org/licenses/license-list.es.html>
- <http://www.mozilla.org/community/>
- <http://ossmole.sourceforge.net/> (collaborative collection and analysis of free/libre/open source project data)
- <http://libresoft.es/> (web site at the Grupo de Sistemas y Comunicaciones GSyC at the Universidad Rey Juan Carlos located in Móstoles, near Madrid Spain).
- <http://sourceforge.net/> (portal encargado de registros y gestión de productos FOSS)
- University of Maastricht and Berlecon Research GmbH 2002 Free/Libre and Open Source Software: Survey and Study FLOSS, en <http://www.infonomics.nl/FLOSS/report/> consultado (June 1 st of 2007)
- <http://www.catb.org/~esr/> (pagina de Eric Raymond)

