

Uso de ODR en las comunidades de desarrollo de software libre

Por David Fernández Mena

Resumen

La revolución del software libre ha sido y es uno de los fenómenos mas imprevisibles y de mayor alcance que han surgido desde el desarrollo de las Tecnologías de la Información. Sin embargo es un fenómeno que debido a propia construcción social y distribuida esta intrínsecamente unido a la noción de conflicto, cuya manifestación mas notoria es la bifurcación o *fork*. En el presente trabajo analizamos los distintos tipos de conflictos y los mecanismos habitualmente empleados para la resolución de los mismos, y si forman parte de los denominados métodos alternativos de resolución de controversias en Linea u ODR o si son parte de una categoría distinta y separada.

Palabras Clave

Software libre, comunidades, conflicto, ADR, ODR, fork, organización, mecanismos alternativos.

Para su posible presentación en el XI Congreso Mundial de Derecho Informático a celebrarse en Octubre de 2011 en a ciudad de Lima (Peru) o su publicación en la revista iDP de la Universitat Oberta de Catalunya.

Introducción

La revolución del software libre ha sido y es uno de los fenómenos mas imprevisibles y de mayor alcance que han surgido desde el desarrollo de las Tecnologías de la Información. Probablemente, el auge y penetración de las mismas no habría tenido el mismo alcance sin el desarrollo de múltiples aplicaciones, la mayoría a nivel de servicios, que un grupo de personas sin animo de lucro y por muy diversas motivaciones (ideológicas, de aprendizaje, superación e incluso económicas) se coordine pseudoespontaneamente para generar un producto de calidad igual o superior a la comercial para luego permitir su modificación y su mejora por el resto de usuarios no tiene un precedente histórico de las mismas dimensiones y alcance.

El fenómeno ha sido estudiado desde diversos enfoques, tales como la Ingeniería de Software y de Procesos¹, los paradigmas económicos ² y la etnografía propia de las comunidades^{3 4}. A nivel jurídico, se han estudiado las consecuencias y aplicación de las licencias consideradas libres ⁵ y los problemas contractuales y de cumplimiento de las licencias son materia existente en tribunales y cortes de todo el orbe. Existe sin embargo una laguna en la investigación del fenómeno al respecto de los conflictos existentes en las comunidades, su impacto manifestado en bifurcaciones (conocidas como *forks*) así como en los mecanismos habituales para la resolución de controversias empleados, teniendo en cuenta la naturaleza en línea y la dispersión geográfica y por tanto normativa, de estas comunidades.

La importancia de los conflictos en este tipo de comunidades es tal, que es hecho fundacional para muchas de ellas y el impacto negativo o positivo que tengan esos conflictos esta en discusión por la academia y por los miembros de la propia comunidad.

-
- 1 Ahmed, F., Campbell, P., Jaffar, A., & Capretz, L. Defects in Open Source Software: The Role of Online Forums. *Proceedings of World Academy of Science: Engineering & Technology*, 52174-178. Retrieved from Academic Search Complete database.
 - 2 Garzarelli, G., Limam, Y., & Thomassen, B. (2008). Open source software and economic growth: A classical division of labor perspective. *Information Technology for Development*, 14(2), 116. Retrieved from MasterFILE Premier database.
 - 3 Concas, G., Lisci, M., Pinna, S., Porruvecchio, G., & Uras, S. (2008). Analysing the Social Networks constituted by Open Source communities. *AIP Conference Proceedings*, 1060(1), 147-150. doi:10.1063/1.3037039.
 - 4 Forero, E. (2007). Cibersocioantropología de comunidades virtuales. (Spanish). *Revista Argentina de Sociología*, 5(9), 64-89. Retrieved from Academic Search Complete database.
 - 5 De Alaiza Cardona, J. (2007). Open Source, Free Software, and Contractual Issues. *Texas Intellectual Property Law Journal*, 15(2), 157-211. Retrieved from Academic Search Complete database.

Comunidades en Línea

Las comunidades en línea son un fenómeno relativamente nuevo y estrechamente relacionado con la penetración de las tecnologías de información en los últimos lustros. La existencia de esos nuevos medios de comunicación, ha generado lugares virtuales de encuentro e intercambio de información, conocimiento e incluso esparcimiento. Sin embargo estas comunidades tienen diferencias muy notorias con las comunidades presenciales, tanto a nivel de composición, heterogeneidad y aportación de sus miembros. Castells, en su obra “La era de la información: economía, sociedad y cultura”⁶, consideraba que “en esas comunidades virtuales ‘viven’ dos tipos muy distintos de poblaciones: una diminuta minoría de aldeanos electrónicos –que se han asentado en la frontera electrónica– y una multitud transeúnte”. Estos diferentes tipos de usuario, cuyas aportaciones se suele considerar que cumplen con la Ley de Pareto (un 20% de los usuarios efectúa un 80% de las aportaciones) tal y como Raymond ya había hincapié⁷ en su trabajo seminal implican necesariamente distintos tipos de implicación, acceso y relaciones entre los distintos miembros, así como diferentes mecanismos de presión y vigilancia social entre los miembros de las mismas.

Existen diversos enfoques sobre la comunidad en línea, o comunidad virtual. Para Cunha Recuero⁸, una comunidad virtual es una agrupación de humanos que “trabajan y mantienen relaciones sociales en el ciberespacio” definición amplia que deja fuera los diversos mecanismos de comunicación con diferentes consecuencias así como el factor interpersonal inherente a toda comunidad y a su composición heterogénea, no solamente por personas físicas, sino también por diferentes agrupaciones e intereses, también comerciales y mercantiles. Al respecto de su composición, Michelle Wilson⁹ considera que “comunidades virtuales se forman y funcionan al interior del ciberespacio, siendo éste el espacio existente dentro de las conexiones, redes y sistemas de las

6 CASTELLS, Manuel, *La era de la información. Economía, sociedad y cultura*. Volumen 1, *La sociedad red*. Alianza Editorial, Madrid, 1997.

7 Raymond, E. (1998) 'The cathedral and the bazaar'. *First Monday*, vol. 3, no. 3; online at <www.firstmonday.org>.

8 DA CUNHA RECUERO, R. (2004), “La Comunidad Virtual: Uma Abordagem Teórica”, Trabajo presentado en el V Seminario Internacional de Comunicação, en el GT de Comunicação e Tecnologia das Mídias, promovido por la PUC/RS.

9 WILSON, M. (2000), “Community in the abstract: a political and ethical dilemma?”, en BELL, D. y KENNEDY, B. (eds.), *Cybercultures reader*, London y New York, Routledge.

Tecnologías de Comunicación. ”. Parece claro que dos elementos definen estas comunidades, siendo el común denominador el empleo de los nuevos sistemas de comunicación que proporcionan las redes para la comunicación de grupos humanos con intereses convergentes.

Mecanismos de comunicación

De acuerdo con DuVal Smith, A¹⁰ y con Donath, J¹¹ los mecanismos de comunicación en línea estructuran el funcionamiento de las comunidades en línea, generando tipos de comunicación diferente y favoreciendo la generación de ciertos comportamientos, tales como los *flame wars* o interacciones agresivas e insultantes entre miembros de una comunidad .

Entre los mecanismos de comunicación, existen dos grandes categorías:

- Los mecanismos síncronos, que permiten la retroalimentación de respuesta en un tiempo breve o en tiempo real, permitiendo y favoreciendo la espontaneidad y reacciones más naturales.
- La comunicación asíncrona es aquella en la que el mensaje se queda en espera y la retroalimentación puede recibirse en cualquier otro momento. Esta falta de sincronía evita la respuesta rápida y apoya la reflexión en la respuesta si bien no inhibe la espontaneidad totalmente.

Así mismo, los mecanismos asíncronos suelen implicar una formalidad y protocolo de comunicación más estructurado e idóneo para trabajos con diferentes zonas horarias, mientras que en los síncronos, la informalidad y la falta de organización en los contenidos es mucho mayor, dando un mayor pie a la generación de conflictos.

Existen multitud de mecanismos de comunicación que se emplean en el funcionamiento habitual de las comunidades en línea. Los principales medios que son de importancia para el presente trabajo son los siguientes:

- *Mailing lists* (o listas de correos): Son sistemas de tipo asíncrono en los cuales, el emisor envía un mensaje por vía de correo electrónico a otra única dirección, la cual a su vez reenvía el citado mensaje a los miembros registrados en la lista, que reciben una copia

10 DuVal Smith, A. (2003) Los <http://dgroups.org/ViewDiscussion.aspx?c=caf8f8bf-0ff2-410e-b0d5-7adacbe19c5a&i=5ffe676b-8c18-4a2e-a2b3-ad2f39ea6e25> problemas de la resolución de conflictos en las comunidades virtuales. En M. Smith & P. Kollock (Eds.) Comunidades en el ciberespacio (pp. 49 – 88). Barcelona: Editorial UOC

11 Donath, J. (2003) Identidad y engaño en la comunidad virtual. En M. Smith & P. Kollock (Eds.) Comunidades en el ciberespacio (pp. 49 – 88). Barcelona: Editorial UOC

idéntica a la enviada. Los *mailing list* suelen tener la posibilidad de ser moderados y que el mensaje tenga que ser aprobado para su posterior reenvío. Sin embargo, esta última opción no es obligatoria y en listas de correo con mucho tráfico esa moderación resulta compleja y de difícil aplicación. Son los sistemas de comunicación más empleados en las comunidades de desarrollo, quedando registro de todos los mensajes enviados, por lo que son los más estudiados a la hora de efectuar análisis sobre los comportamientos de las mismas¹²

- IRC: Es un sistema de comunicación de tipo síncrono, en “tiempo real”, en los cuales se escribe el mensaje el cual es recibido instantáneamente por todas las personas conectadas en ese momento al sistema, los cuales pueden a su vez responder. Implica necesariamente que para poder comunicarse, los usuarios han de estar conectados en ese mismo instante. La comunicación desarrollada es más informal, con el uso de expresiones coloquiales, abreviaturas y, en general, una menor estructuración de los temas tratados¹³. Si bien los mensajes pueden registrarse, esta opción no está centralizada ni quedan almacenados los mismos en el sistema al que se conectan. A diferencia de los modernos *Instant Messengers* (MSN Messenger, ICQ o similares) no existe una autoridad centralizada, por lo que cualquier usuario se puede conectar con cualquier identidad y a su vez, tiene que conectarse a cada servidor diferente de IRC.
- CVS: Un CVS o *Concurrent Version System* es un sistema de control de versiones, concretamente para el código fuente de los programas de ordenador, en el cual los usuarios y desarrolladores, con distintos tipos de permisos jerárquicamente distribuidos, transmiten sus aportaciones al código generándose un número de versión y almacenando y haciendo posible la gestión de los cambios. Si bien CVS era originalmente el nombre de un sistema en concreto, hoy es muy habitual su empleo como genérico de este tipo de sistemas, aunque el sistema de control de versiones más empleado sea Subversion, una evolución del CVS original. No es un medio de comunicación usual, pero implica el envío de mensajes que pueden tener importancia comunicativa, particularmente el hecho de tener permiso de lectura y aún más importante, de escritura en el trabajo del grupo de desarrollo. La mera

12 DUCHENEAUT, N. (2005). Socialization in an Open Source Software Community: A Socio-Technical Analysis. *Computer Supported Cooperative Work (CSCW)*, 14(4), 323-368. Retrieved from E-Journals database.

13 DuVal Smith, A. (2003) Los problemas de la resolución de conflictos en las comunidades virtuales. En M. Smith & P. Kollok (Eds.) Comunidades en el ciberespacio (pp. 49 – 88). Barcelona: Editorial UOC

existencia de un sistema de control de versiones estructura una jerarquía en la comunidad¹⁴ sobre la cual se pueden efectuar estudios de tipo etnográfico ¹⁵.

Conflictos en las comunidades en línea

Los conflictos forman parte intrínseca de las relaciones humanas, y por lo tanto de las comunidades. Sin embargo, las comunidades en línea por sus especiales características, son proclives a la existencia de conflictos de unos tipos determinados y diferentes.

Basándonos en Bely Mora ¹⁶ podemos establecer una clasificación de las diferentes causas y facilitadores de conflictos detectadas por diversos autores:

1. El diseño del propio software de comunicación: El software de comunicación genera una forma diferente de interactuar entre los distintos miembros de la comunidad ¹⁷. El uso de mecanismos síncronos o asíncronos, así como la existencia de moderación jerárquica o de sistemas de apreciación de las aportaciones¹⁸ median las relaciones entre los miembros de la comunidad, modelando su comportamiento.
2. El anonimato. Aunque existan sistemas de comunicación que traten de evitar la existencia de anonimato, es conocido de la gran mayoría de usuarios que estos sistemas son mayormente burlables, existiendo en un gran número de ocasiones la posibilidad de mostrarse con un avatar anónimo diferente al de la propia persona. Wallace, Reid, y Donath (citados por DuVal Smith, A) consideran esto una de los mayores generadores de conflictos; dejan de existir factores de inhibición social, (Wallace), no existen consecuencias probables hacia la seguridad física (Duvall Smith) y se genera una posible fragmentación de identidad implicando un comportamiento diferenciado al de

¹⁴ *ibidem*

¹⁵ (2006). A Methodological Framework for Socio-Cognitive Analyses of Collaborative Design of Open Source Software. *Computer Supported Cooperative Work (CSCW)*, 15(2-3), 229-250. Retrieved from E-Journals database

¹⁶ Bely Mora, El estudio del conflicto en comunidades virtuales: Una revisión de aportes recientes. <http://www.cibersociedad.net/congres2006/gts/comunicacio.php?llengua=es&id=1>

¹⁷ M. Smith & P. Kollock (2003) Las Comunidades en el ciberespacio. En M. Smith & P. Kollock (Eds.) *Comunidades en el ciberespacio* (pp. 19 – 48). Barcelona: Editorial UOC

¹⁸ ESTALELLA, A.: «Pobrecito hablador. Conflictos por la libre participación en una comunidad colaborativa», *Athenea Digital*, núm. 7, 2005a, <http://www.bib.uab.es/pub/athenea/15788646n7a12.pdf>

la identidad habitual (Kolko & Reid).

3. El conocimiento del entorno: Las comunidades virtuales implican necesariamente un grado de conocimiento de las tecnologías de información. Ese grado puede ir de una exigencia mayor o menor en función de las características propias de cada comunidad en línea. Sin embargo, las características de tecno-meritocracia descritos por Contreras¹⁹ implican una segregación y un efecto discriminatorio entre aquellos considerados con niveles de conocimientos más bajos, para los cuales existe toda una terminología específica: *Lamers*, *noobs* etc. Este factor, también descrito por Castells en términos de brecha digital²⁰, dificulta la integración de muchos sujetos en comunidades virtuales.
4. Diversidad cultural. La mayor heterogeneidad de las comunidades en línea viene dada por diversos factores; geográficos, sociales, socioeconómicos, políticos etc. que hacen que diversos sujetos de diversos trasfondos coincidan en este tipo de comunidades de maneras que no serían posibles mediante comunidades físicas o presenciales. Esta diversidad es una fuente natural de conflictos, tal y como hace mención Duvall Smith.
5. Diversas causas. M. Smith & P. Kollock en “Comunidades en el Ciberespacio” describen otro conjunto de causas, que son comunes en ocasiones con las comunidades en el mundo no virtual, tales como ideología o incluso raza. Si bien es cierto que estas causas de conflicto pueden ocurrir, particularmente las de ideología política, no consideramos que sean de una especificidad suficiente a las comunidades en línea siendo replicas de comportamientos habituales en los conflictos no en línea.

Las comunidades de desarrollo de software libre como sujeto de estudio diferenciado.

Las comunidades de desarrollo de software libre son un tipo particular de comunidad en línea, orientadas a la finalización de un producto (un programa de ordenador) y con herramientas bastante estandarizadas, tanto de comunicación como para el desarrollo del producto. Una característica clara que las define es la aportación voluntaria de trabajo por parte de sus miembros, para posteriormente liberarlo para terceros bajo un esquema de licenciamiento de los considerados libres o de código abierto, lo cual en la práctica supone la adopción de una licencia aprobada por la *Free Software*

19 Contreras, P. (2004) *Me llamo Kohfam. Identidad hacker: una aproximación antropológica*. Barcelona: Editorial Gedisa.

20 CASTELLS, Manuel, *La era de la información. Economía, sociedad y cultura*. Volumen 1, *La sociedad red*. Alianza Editorial, Madrid, 1997.

Foundation o por la *Open Source Initiative*, aunque esta última implique diferencias leves ideológicas que hagan que no se le pueda denominar software libre, al no tener los mismos requisitos para su aprobación²¹.

Esta estandarización de sus necesidades y herramientas ha hecho posible que existan alojamientos en masa que permiten la rápida creación de una comunidad de este tipo, como por ejemplo Sourceforge²², la cual es una fuente habitual de estudio de la composición, mecanismos y casos de éxito por parte de la investigación sobre el software libre.

Las comunidades de Software libre son extremadamente heterogéneas en su composición, existiendo comunidades locales que desarrollan adaptaciones, traducciones y localizaciones de algún otro producto (como Joomla hispano²³) comunidades que desarrollan un producto sectorial pero útil para multitud de países, como la comunidad Moodle²⁴ o proyectos de tamaño extremadamente considerable y que abarcan no solo el desarrollo del software, sino el marketing, diseño gráfico e incluso protección legal, como GNOME o la propia *Free Software Foundation*.

Sin embargo, siguiendo a Demazière, Horn y Zune²⁵, su funcionamiento interno sigue tres mecanismos de regulación:

- Un control basado en la propia existencia de producto como finalidad, el cual por su propia naturaleza impone mecanismos de control para evitar la pérdida de su naturaleza y funcionalidad perseguidas. Según Fogel²⁶, las comunidades de éxito deberían tener esta finalidad y misión documentadas para resolver cualquier futura controversia y posibilidad de *fork*.
- Una regulación autónoma basada en los distintos compromisos y sensibilidades: Cada comunidad persigue sus únicos fines, siendo estos fines los que evitan un exceso de

21 Estos requisitos exigidos por la FSF son las llamadas cuatro libertades:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2)
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.

22 <http://sourceforge.net/>

23 <http://www.joomlahispano.org/>

24 <http://www.moodle.org/>

25 Demazière, D., Horn, F., & Zune, M. (2007). The Functioning of a Free Software Community: Entanglement of Three Regulation Modes -- Control, Autonomous and Distributed. *Science Studies*, 20(2), 34-54. Retrieved from Academic Search Complete database.

26 Fogel, K. (2005). Producing Open Source Software: How to Run a Successful Free Software Project, O'Reilly Press, Sebastopol, CA.

heterogeneidad interna que evitaría el cumplimiento de los fines de la comunidad. Los compromisos y aportaciones de cada comunidad implican diferentes niveles de regulación hacia los miembros internos.

- Una regulación distribuida y basada en el juicio de los miembros. Según los autores es el mecanismo de equilibrio entre los dos mecanismos de control previos. Esencialmente, se manifiesta como un reconocimiento de las aportaciones de los miembros de la comunidad, sea en forma de reconocimiento de los logros, como de la construcción de una reputación. Este mecanismo de regulación también es mencionado por varios estudiosos de la etnografía de comunidades en línea, considerándolo parte de los ritos de paso para la pertenencia a una comunidad (Duchenaud y J. Ajith Kumar²⁷). En otros términos, se ha llegado a manejar que el mecanismo y estructura de una comunidad en línea es la llamada meritocracia (Estaella y Raymond) llegando a considerarse algo innato a las comunidades de desarrollo. Sin embargo, esto está siendo discutido por una parte de los estudiosos de las comunidades, incluyendo por parte del autor original del concepto²⁸.

Tamaño de las comunidades

No existe actualmente una medición confiable al respecto de la magnitud del movimiento del software libre. La tremenda dispersión, la falta de estructura común y, finalmente, lo espontáneo y autónomo de la mayor parte de las comunidades hace que resulte extremadamente complejo efectuar estudios cuantitativos al respecto. Debido a esto, la mayor parte de los estudios cuantitativos sobre el Software Libre intentan basarse en las aportaciones al código²⁹ en el número de proyectos y aportaciones³⁰ y casi todos, en el análisis de estudio de caso de una comunidad concreta: Michelaidis con KDE³¹, German con GNOME³², Duchenaud con Python).

27 Kumar, J. Ajith (2010) Forking, Leadership Control and Social Capital in Open Innovation. En Management of Innovation and Technology -ieeexplore.ieee.org

28 Young, Michael (1958) The Rise of the Meritocracy (1870-2033): An Essay on Education and Equality.

29 Koch, S. (Ed.), (2005). Free/Open Source Software Development, Idea Group Publishing, Hershey, PA

30 Mockus, A., Fielding, R., & Herbsleb, J.D., (2002). Two Case Studies of Open Source Software Development: Apache and Mozilla, ACM Transactions on Software Engineering and Methodology, 11(3), 309-346.

31 Michaelides, G. (2006) Far From Equilibrium: Emergence, Self-Organization, and Diversity in the KDE Community, Ph.D thesis. Business School, University of Nottingham.

32 German, D. M. (2003), The GNOME project: a case study of open source, global software development. Software Process: Improvement and Practice, 8: 201–215. doi: 10.1002/spip.189

Existen herramientas y voluntad para continuar con el estudio de las comunidades y se puede observar un incremento en el número de publicaciones sobre la materia en los últimos años. Sin embargo, la tremenda magnitud hace que los estudios que pretendan medir el tamaño del conjunto se enfrenten a un gran reto. En los últimos años, se han ido generando herramientas para automatizar ese análisis por lo que podrá efectuarse un mayor análisis al respecto en los próximos años.³³

La importancia cualitativa de los desarrollos y el incremento sostenido de su uso por parte de los usuarios y administraciones públicas se dan por sentadas, tanto que diferentes organismos públicos e internacionales están fomentando su uso y desarrollo de manera sostenida por lo que solo podemos esperar un incremento del tamaño de las comunidades para los próximos años..

El *fork*

Un *fork* o bifurcación consiste en la creación de una nueva versión y comunidad de un proyecto de desarrollo de software libre ya existente empleando el código fuente del proyecto original. Debido a las características de libertad de modificación del software libre, la posibilidad de generar un nuevo software no están limitadas por constricciones legales de ningún tipo y es una posibilidad y riesgo real a lo largo de todo el proyecto³⁴.

Un *fork* esta considerado generalmente como algo negativo, puesto que los recursos humanos que se dedicaban al proyecto quedan divididos y el producto final a elaborar, amenazado. El famoso glosario “*Jargon File*” comenta que esto es debido tanto a la pérdida de recursos como a la acritud y al conflicto generado entre los miembros de la comunidad.

Sin embargo, en los últimos años se ha venido redefiniendo la negatividad del *fork*, debido tanto a los resultados de muchos *forks* como a la existencia de algunos *forks* bendecidos por los propios miembros de la comunidad.

Forks notorios y de resultado han sido el proyecto X.org, sustituyendo a Xfree, Mozilla Firefox de Mozilla, Xvid de Open Divx así como un gran número de distribuciones de BSD. Es novedad, por ejemplo, el *fork* de OpenOffice en un nuevo proyecto llamado Libreoffice por disconformidad de miembros destacados de la comunidad con el manejo de Oracle de la comunidad desde la adquisición de Sun por parte de la misma.

33 German, D. and A. Mockus (2003): Automating the Measurement of Open Source Projects. In Proceedings of the 3rd Workshop on Open Source Software Engineering. Portland OR, pp. 63–68.

34 Karatzogianni, A., & Michaelides, G. (2009). Cyberconflict at the edge of chaos: Cryptohierarchies and self-organisation in the open-source movement. *Capital & Class*, (97), 143-157. Retrieved from Academic Search Complete database.

Aun así, es sencillo iniciar un *fork*, por las razones legales comentadas, pero es extremadamente difícil llevarlo a buen término. Existen gran cantidad de *forks* que no lograron sostenerse y desarrollar un producto final. Entre otros muchos, tenemos a goneME de Gnome, y diversos forks de MySQL. Ciertamente, la existencia de un *fork* es probablemente la principal manifestación cuantificable de un conflicto en una comunidad de desarrollo de software libre, aunque el desarrollo de las comunidades en los últimos tiempos obliga a matizar esa idea, puesto que existen forks que no han implicado conflicto alguno, aunque son esencialmente anecdóticos, por lo que podemos afirmar que el *fork* sigue siendo el conflicto más grave que pueda existir en una comunidad. Esto no implica que las comunidades que tengan conflictos necesariamente tengan *forks*, siendo el conflicto algo inevitable y coexistente con los grupos de esta naturaleza ³⁵.

Basándonos en Moen ³⁶ existen dos tipos de *forks*:

1. *Forks* internos, en los cuales miembros de la comunidad deciden tomar el código y generar un proyecto paralelo al proyecto principal, generando su propia comunidad con sus diferentes mecanismos de control y funcionamiento. Para definir más específicamente este *fork* debemos distinguirlo de la creación de un *branch* dentro de una comunidad, la cual sigue bajo el paraguas de la misma comunidad. Estos *forks* suelen estar asociados, generalmente, a conflictos internos. Ejemplos clásicos son el proyecto Xorg del proyecto Xfree86 o más modernamente, el videojuego Diaspora basado en la serie de televisión “Galactica”, sobre el proyecto original “Beyond the red line”
2. *Forks* externos en los cuales un grupo de desarrolladores externos o mayormente externos a la comunidad de desarrollo, decide tomar el proyecto y generar otro paralelo para darle otro rumbo de desarrollo e incluso otra finalidad al mismo código fuente. Ejemplos de esto serían el compilador “pgcc” optimizado para procesadores Pentium sobre el original “gcc” del proyecto GNU.

Causas del fork

Las causas de un *fork* son variadas, pero se puede llegar a establecer una clasificación tentativa basada en los casos de mayor renombre:

³⁵ ibidem

³⁶ Moen, Rick (1999). “Fear of Forking”. http://linuxmafia.com/faq/Licensing_and_Law/forking.html

- Arquitectura: desacuerdos en la arquitectura de desarrollo, en las funcionalidades o en la falta de documentación o misión son generadores de un alto número de forks, los cuales atraen a desarrolladores con iniciativa que no ven satisfechas sus expectativas a crear un nuevo proyecto para poder ejercer su labor de desarrollo. Ejemplos ya citados tendríamos el compilador “pgcc”, para implementar la optimización del código fuente de C para procesadores Pentium, lo cual fue rechazado por el proyecto “gcc” original. También entra en esta categoría el proyecto NeoOffice, para llevar a la plataforma MacOSX la suite OpenOffice, la cual en su versión para esta plataforma tenía un rendimiento escaso, por lo que parte de la comunidad decidió generar un proyecto separado.
- Políticas: El problema del desarrollo organizativo de las comunidades, por el estilo de liderazgo, la acumulación de poder (o la percepción de acumulación) o los mecanismos de toma de decisiones suelen ser causa común de separación de la comunidad. Teniendo en cuenta los mecanismos de ingreso y gestión en las comunidades basados en el prestigio, la existencia de diversos grados hacen estos conflictos una amenaza constante a la persistencia de la comunidad. Existen muchos casos paradigmáticos de este tipo de forks, entre los cuales caben mencionar el CMS Joomla como *fork* de Mambo, debido a lo que miembros de la comunidad percibían como un exceso de poder por parte de uno de los líderes del proyecto, o actualmente el *fork* de OpenOffice llamado LibreOffice debido a la desconfianza en el control que pueda ejercer Oracle sobre el proyecto principal desde que adquirió Sun microsystems antiguo y principal *sponsor* del original.
- El licenciamiento es una causa común de separación de la comunidad. Los miembros que colaboran en el desarrollo de un producto de software libre suelen tener razones diversas³⁷ para que su proyecto tenga la mayor difusión posible mediante una licencia aprobada por los organismos habituales. Sin embargo, no todas las licencias garantizan las mismas libertades sobre el producto del trabajo de la comunidad, que consideran destinado al público y sobre el cual puede y suele haber conflictos al respecto de su grado de explotación y control. Xorg fue uno de estos casos, debido a la nueva licencia que se implementó para la versión XFree86 4.4 y que generó la creación de una fundación que llevaría el nombre Xorg y que desarrollaría un *fork* bajo la licencia del proyecto original (MIT). También entre otros muchos, cabe mencionar los múltiples forks de Unix BSD (FreeBSD, OpenBSD y como productos comerciales MacOSX

37 Bonaccorsi, A. and Rossi, C., (2006). Comparing motivations of individual programmers and firms to take part in the open source movement: From community to business. Knowledge Technology & Policy, 18(4), Winter, 40-64.

y SunOS) que incluyen conflictos con el titular original AT&T.

Principales forks

Es extremadamente ardua la tarea de efectuar un catalogo de los forks existentes en el movimiento del software libre y que excede por mucho el alcance de este articulo. La labor es compleja debida a las razones expuestas sobre la investigación de las comunidades de software libre y por razones conceptuales adicionales, tales como la existencia de forks no declarados (el licenciamiento libre significa que cualquiera puede crear una nueva aplicación basada en el código fuente de cualquier programa así licenciado) o la linea gris que pueda llegar a distinguir un *fork* y el reuso de código fuente de otros productos. Aun así, por la importancia e impacto que han supuesto para el movimiento del Software libre y para el desarrollo de las tecnologías de información en general, cabe mencionar los siguientes :

- NSCA-Apache:El servidor http llamado Apache, popularmente conocido como servidor de paginas web, originalmente nace de un proyecto en estado de abandono, el http daemon de la NSCA (National Center for Supercomputing Applications). Un grupo de *webmasters* individualmente comenzaron a desarrollar parches para darles nuevas funcionalidades y corregir errores. Finalmente se transformaron en comunidad mediante diversos apoyos y licenciaron su producto cumpliendo las 4 libertades exigidas en el software libre generando su propia licencia. El resultado es el servidor de paginas web de referencia , con una cuota de mercado del 59,35% en Enero de 2011 acuerdo a Netcraft.³⁸
- Mozilla-Firefox:Firefox nace como una rama experimental del proyecto Mozilla, el cual había nacido tras la liberación del código fuente del antiguo Netscape Navigator. Sin embargo, muchos desarrolladores lo consideraban particularmente pesado y lento debido a las exigencias comerciales impuestas por lo que comenzaron como un *branch* que posteriormente se transformo por derecho propio en una comunidad completa y en un producto de alta penetración en el mercado. Actualmente, de acuerdo a W3Counter ³⁹tiene una cuota de mercado de un 30,3 % y tendencia a un incremento todavía mayor.
- egcs-gcc. Egcs es un compilador de programas basado en el compilador “gcc”, el cual originalmente estaba desarrollado por la FSF. Un grupo de desarrolladores y compañías estaba

38 <http://news.netcraft.com/archives/category/web-server-survey/>

39[<http://www.w3counter.com/globalstats.php?year=2010&month=12>

inconforme por la falta de desarrollo hacia otras arquitecturas y lenguajes (C++) que tenía el proyecto, así como por la falta de aceptación de sus sugerencias y aportaciones, por lo que decidieron generar su propio proyecto llamado “egcs” (*Experimental/Enhanced GNU Compiler System*). El resultado fue incluso superior al original y la FSF lo aceptó deteniendo el desarrollo de “gcc”, considerando a “egcs” como la nueva base de desarrollo y reintegrándolo en “gcc” en su versión 2.95, que realmente era la versión 1.2 de “egcs”.

El breve análisis de estos tres proyectos da cuenta de la importancia del desarrollo de proyectos basados en forks o bifurcaciones y el impacto tenido en el desarrollo de las TIC .

Análisis sobre la negatividad del Fork

Si bien el *fork* es en muchas ocasiones la expresión de la negatividad de un conflicto, su negatividad es ampliamente debatida, tanto en y por la comunidad como por las publicaciones científicas. Los argumentos que le otorgan una consideración negativa comienzan por el desperdicio de recursos dedicados a la misma finalidad (Michaelides, Garzarelli), por las posibles incompatibilidades que puedan ocurrir entre ambas versiones debido a los diferentes rumbos que puedan tomar las comunidades (Michaelides) y por la acritud entre los diferentes miembros que pueda implicar el proceso de división de las comunidades. Sin embargo, la mayor parte de los autores reconoce que el fenómeno del *forking* está implícito en la generación del software libre y que no es posible considerar de manera taxativa como negativo la generación de otras comunidades como reconocen Moen y Ajith Kumar. En muchos casos, la libertad de efectuar un *fork* está considerada implícitamente en el licenciamiento, que es la esencia del propio software libre por lo que este se considera una consecuencia necesaria e incluso deseada.

El éxito de muchos proyectos nacidos de un *fork* hacen que actualmente se esté reconsiderando la negatividad con la que habitualmente se mira el fenómeno, no necesariamente considerándolo pernicioso, sino efectuando un análisis casuístico para descubrir el tipo de *fork* y ver si este está asociado a conflicto negativo. Generalmente, son de esta consideración los forks internos de tipo político y de licenciamiento, y algunos de tipo técnico. Es rara la ocasión en la que un *fork* externo tiene connotaciones conflictivas, existiendo incluso relaciones cordiales entre ambas comunidades, como es el caso de las comunidades de Ubuntu y Debian.

Mecanismos de resolución de controversias en Comunidades de desarrollo

Como ya se ha efectuado mención, las controversias y conflictos forman parte esencial de cualquier relación humana incluyendo las comunidades en línea y más específicamente, las comunidades de desarrollo de software libre, con una tendencia inherente a la bifurcación y a la ordenación caótica tal y como establecen los trabajos de Michaelides.

Sin embargo, las comunidades no cumplirían su función y su misión si no hubieran establecido algún tipo de mecanismo para la resolución de estos conflictos. Estos mecanismos en ocasiones pueden estar considerados como mecanismos de resolución alternativa de controversias o ADR (u ODR en su versión online). Consideraremos ADR, a efectos de esta clasificación a todo mecanismo de resolución de controversias de naturaleza no jurisdiccional o administrativa, excluyendo explícitamente cualquier litigio civil o mercantil así como los procedimientos administrativos u arbitrales impuestos por una autoridad o por un miembro externo a la propia comunidad.

Los ODR

Siguiendo la taxonomía propuesta por Lucia Ruggeri⁴⁰ los ODR o mecanismos de resolución de controversias en línea, se pueden clasificar en:

- Subastas a ciegas: Un sistema seguro recibe las pujas otorgándolas al mejor postor, de acuerdo a algún criterio establecido en el sistema, generalmente la puja más alta.
- Arbitraje a distancia: en el cual las partes establecen sus posturas ante un tercero independiente el cual toma una decisión vinculante para ambas partes
- Mediación a distancia: similar a la mediación tradicional en la cual un tercero independiente asiste a las partes a que resuelvan sus problemas sin efectuar propuestas
- Negociación asistida en la cual un sistema o persona les va guiando paso a paso para la resolución de su controversia y facilitando el acuerdo mediante un proceso de negociación ya establecido y estricto, documentando el proceso de manera proactiva y resolutiva.

40 RUGGERI, Lucia (2010). «ADR y ODR y su taxonomía. La identificación de caracteres». En: «Justicia relacional y métodos electrónicos de resolución (ODR): hacia una armonización técnica y legal» [monográfico en línea]. IDP. Revista de Internet, Derecho y Política. N.º 10. UOC.

Mecanismos habituales: El dictador benevolente o el proceso democrático.

Las comunidades de éxito analizadas (GNOME, KDE, Ubuntu, Moodle, Subversion, Apache y OpenOffice) debido a su gran tamaño y dilatada trayectoria, cuentan con una estructura organizativa documentada y publicada en sus diversos sitios. Collins-Sussman y Fitzpatrick, en su conferencia en el GooglePlex en 2007⁴¹ establecen (basados en Fogel) una clasificación de las diversas comunidades de desarrollo y su éxito basados precisamente en su estructura organizativa⁴²

La gran mayoría de las comunidades (GNOME, Moodle, Ubuntu, KDE incluso la comunidad de Linux) establecen como mecanismo de toma de decisiones algún proceso que incumba a un miembro de la comunidad (Linus Torvalds es el caso mas representativo) o a varios en todo lo que sean decisiones de desarrollo. Estos mecanismos de toma de decisiones originalmente pertenecen mas al ámbito de la cultura organizacional que a la resolución de controversias, pero en muchas ocasiones, el papel del tomador de decisiones implica el apoyo a la resolución de conflictos. Sin embargo, es observable que con muchas comunidades (como Moodle software de gestión educativa y docente de referencia) establecen la figura de un “dictador benevolente” el cual tiene la ultima palabra, ejerciendo la función de “arbitro” ante las diferencias en conflictos de tipo técnico. Algo similar ocurre en Debian y Ubuntu, aunque en estos casos se hablaría de organismos colegiados, no de una única figura. En otras comunidades, se opta por procesos de votación voluntaria, si bien estos sufragios solo toman en cuenta a ciertas categorías de miembros de la comunidad (sufragio censitario). Estos dictadores benevolentes ejercen una función de moderador en los debates pero en muchas ocasiones, proceden a efectuar procesos de mediación⁴³ (cuando las partes en conflicto no tienen un desequilibrio de poder notable, como es propio en los procesos de mediación) y en caso de falta de acuerdo, toma una decisión. Es difícil encajar a priori estos mecanismos dentro de los ADR u ODR, debido al probable interés por parte del tercero que toma la decisión y que puede llegar a formar parte del conflicto en si, cuando no generarlo, cuyo efecto es ejecutivo y que generalmente ya esta prevenido de la existencia del conflicto desde un punto temprano de su desarrollo. Sin embargo, en los casos del dictador benevolente , el mismo suele ejercer funciones de conciliador y de mediación entre las partes en controversia. La novedad del proceso consiste en que se efectúa en la practica totalidad de los casos, mediante mecanismos de comunicación en linea, por lo que podríamos mencionar que existe una forma

41 <http://www.youtube.com/watch?v=ZtYJoatnHb8>

42 Elliott, M. and Scacchi, W., (2005). Free Software Development: Cooperation and Conflict in A Virtual Organizational Culture, in S. Koch (ed.), Free/Open Source Software Development, 152-172, Idea Group Publishing, Hershey, PA.

43 VOYLES, Rick (2004) "Managing an imbalance of power". <http://www.mediate.com/articles/voylesR3.cfm>

espontánea de mediación informal que podría encajar conceptualmente dentro de los ODR, si bien el rol del dictador benevolente puede y suele imponerse sobre el del mediador. En ninguna comunidad analizada hemos podido encontrar RUGGERI, Lucia (2010). «ADR y ODR y su taxonomía. La identificación de caracteres». En: «Justicia relacional y métodos electrónicos de resolución (ODR): hacia una armonización técnica y legal» [monográfico en línea]. IDP. Revista de Internet, Derecho y Política. N.o 10. UOC.mención alguna a clausula arbitral o remisión a un proceso de mediación, conciliación o negociación asistida.

Podemos observar que en base a lo anteriormente explicado, se podría considerar la existencia de mecanismos de ODR dentro de las comunidades de desarrollo de software libre, aunque su desarrollo organizativo hace difícil encajar los diferentes mecanismos de resolución de controversias dentro de una sola categoría. La informalidad y espontaneidad inherente a las comunidades de desarrollo de software libre suele implicar que el sujeto que se dedique a la resolución de la controversia, si bien lo efectúa de manera virtual o en línea, no adopta un único rol de mediador o de arbitro, oscilando entre el conciliador y el mediador (o incluso el arbitro), cuando finalmente no acaba ejerciendo la imposición como mecanismo de resolución. Cada comunidad amerita un estudio de caso distinto y como tales se han observado diferentes pautas de comportamiento y resolución. Como ejemplo, tenemos los trabajos antes mencionados de Mockus (GNOME), Michaelides(KDE) y Duchenaout(Python).

Inexistencia de jurisprudencia sobre conflictos en comunidades.

El desarrollo y propia naturaleza gratuita de las comunidades de software libre hacen que existan dificultades para el acceso de las partes y los conflictos que arrastran hacia un proceso jurisdiccional. Si bien es cierto que existen procesos que han envuelto a comunidades de software libre (la marca Unix) o incluso fundaciones que se dedican a su defensa (la propia FSF) estos conflictos tienen una naturaleza de defensa de la propiedad intelectual o de la propiedad industrial y sus patentes, no siendo los conflictos resueltos referentes a desavenencias entre miembros de la comunidad debido a alguna de las causas antes expresadas. La dificultad de acceso, el alto costo y los largos tiempos de espera para la resolución hacen que la jurisdicción ordinaria carezca de sentido para la resolución de estos conflictos cuyo trasfondo es la forma de un producto de naturaleza tecnológico, que seguramente estaría obsoleto (el producto o la controversia) para el momento de la emisión de la sentencia.

Validez y valor legal

La falta de jurisprudencia no necesariamente implica que haya una falta de prueba de concepto al respecto de la validez jurídica de estos mecanismos. Generalmente estarán contemplados dentro del principio jurídico de la libertad contractual de las partes las cuales se someten a un nuevo acuerdo o contrato (quizás una novación de las obligaciones) con la resolución de cada controversia en el caso de la negociación, o mediante la aceptación de las normas de la comunidad que establezcan la resolución de controversias por parte de un tercero legitimado. No forma parte del alcance de este trabajo efectuar un análisis detallado al respecto del marco legal y uso de los ADR y ODR en España pero es necesario efectuar una mención sobre la regulación de los mismos. Actualmente, en España, el panorama es de un notable dispersión legislativa, existiendo leyes de mediación y justicia alternativa a nivel de Comunidades Autónomas, con diferentes alcances y ambiciones en cada una de las comunidades. Sin embargo, existen procesos de armonización legislativa que apoyarían a aclarar y regular la situación, tal y como hace notar Vilalta⁴⁴, dando lugar a una mayor seguridad jurídica.

Conclusiones:

La existencia de mecanismos ADR y ODR es una realidad de manera informal aunque de manera documentada en los diferentes “estatutos” que regulan la estructura organizativa de la mayoría de las comunidades de Software Libre. Sin embargo, las comunidades mas pequeñas podrían verse beneficiadas por la existencia de un servicio especializado que con gran legitimidad por el conocimiento del proceso de desarrollo de software libre (tecno-meritocracia) les apoyaran en la resolución de sus conflictos. El impulso a los ODR resultante, que son los únicos medios funcionalmente validos en este tipo de comunidades, resultaría de gran ventaja e impulso para la socialización y popularización que estos medios necesitan⁴⁵.

44 VILALTA, Aura Esther (2010). «Resolución electrónica de disputas. Primeras reflexiones en torno a la necesidad de un marco normativo internacional». En: «Justicia relacional y métodos electrónicos de resolución (ODR): hacia una armonización técnica y legal» [monográfico en línea]. IDP. Revista de Internet, Derecho y Política. N.o 10. UOC.

45 RABINOVICH-EINY, Orna (2010). «Mejorar la responsabilidad y el aprendizaje en la resolución de disputas a través de la tecnología». En: «Justicia relacional y métodos electrónicos de resolución (ODR): hacia una armonización técnica y legal» [monográfico en línea]. IDP. Revista de Internet, Derecho y Política. N.o 10. UOC

Referencias básicas:

- (2004). OSS watch: Open source software advisory service. *New Review of Information Networking*, 10(1), 87-89. Retrieved from E-Journals database.
- (2006). A Methodological Framework for Socio-Cognitive Analyses of Collaborative Design of Open Source Software. *Computer Supported Cooperative Work (CSCW)*, 15(2-3), 229-250. Retrieved from E-Journals database
- Ahmed, F., Campbell, P., Jaffar, A., & Capretz, L. Defects in Open Source Software: The Role of Online Forums. *Proceedings of World Academy of Science: Engineering & Technology*, 52174-178. Retrieved from Academic Search Complete database.
- Concas, G., Lisci, M., Pinna, S., Porruvecchio, G., & Uras, S. (2008). Analysing the Social Networks constituted by Open Source communities. *AIP Conference Proceedings*, 1060(1), 147-150. doi:10.1063/1.3037039.
- Concas, G., Lisci, M., Pinna, S., Porruvecchio, G., & Uras, S. (2008). Analysing the Social Networks constituted by Open Source communities. *AIP Conference Proceedings*, 1060(1), 147-150. doi:10.1063/1.3037039.
- De Alaiza Cardona, J. (2007). Open Source, Free Software, and Contractual Issues. *Texas Intellectual Property Law Journal*, 15(2), 157-211. Retrieved from Academic Search Complete database.
- de Joode, R., Lin, Y., & David, S. (2006). Rethinking free, libre and open source software. *Knowledge, Technology, and Policy*, 18(4), 5-16. Retrieved from E-Journals database.
- Demazière, D., Horn, F., & Zune, M. (2007). The Functioning of a Free Software Community: Entanglement of Three Regulation Modes -- Control, Autonomous and Distributed. *Science Studies*, 20(2), 34-54. Retrieved from Academic Search Complete database.
- Donath, J. (2003) Identidad y engaño en la comunidad virtual. En M. Smith & P. Kollock (Eds.) Comunidades en el ciberespacio (pp. 49 – 88). Barcelona: Editorial UOC
- DUCHENEAUT, N. (2005). Socialization in an Open Source Software Community: A Socio-Technical Analysis. *Computer Supported Cooperative Work (CSCW)*, 14(4), 323-368. Retrieved from E-Journals database.

- DuVal Smith, A. (2003) Los problemas de la resolución de conflictos en las comunidades virtuales. En M. Smith & P. Kollock (Eds.) Comunidades en el ciberespacio (pp. 49 – 88). Barcelona: Editorial UOC
- Elliott, M. and Scacchi, W., (2005). Free Software Development: Cooperation and Conflict in A Virtual Organizational Culture, in S. Koch (ed.), *Free/Open Source Software Development*, 152-172, Idea Group Publishing, Hershey, PA.
- ESTALELLA, A.: «Pobrecito hablador. Conflictos por la libre participación en una comunidad colaborativa», Athenea Digital, núm. 7, 2005a, <http://www.bib.uab.es/pub/athenea/15788646n7a12.pdf>
- Forero, E. (2007). Cibersocioantropología de comunidades virtuales. (Spanish). *Revista Argentina de Sociología*, 5(9), 64-89. Retrieved from Academic Search Complete database.
- Forero, E. (2007). Cibersocioantropología de comunidades virtuales. (Spanish). *Revista Argentina de Sociología*, 5(9), 64-89. Retrieved from Academic Search Complete database.
- Gálvez, A. & Tirado, F. (2004) III Conflicto y Sociedad Red. En Persona y Sociedad, Vol. XVIII (1), 225 – 240.
- Garzarelli, G., Limam, Y., & Thomassen, B. (2008). Open source software and economic growth: A classical division of labor perspective. *Information Technology for Development*, 14(2), 116. Retrieved from MasterFILE Premier database.
- Gruen, N. (2005). GEEKS BEARING GIFTS OPEN SOURCE SOFTWARE AND ITS ENEMIES. *Policy*, 21(2), 39-44. Retrieved from Academic Search Complete database.
- Jones, B. (2007). COMMENT: VIRTUAL NEIGHBORHOOD WATCH: OPEN SOURCE SOFTWARE AND COMMUNITY POLICING AGAINST CYBERCRIME. *Journal of Criminal Law & Criminology*, 97(2), 601. Retrieved from MasterFILE Premier database.
- Karatzogianni, A., & Michaelides, G. (2009). Cyberconflict at the edge of chaos: Cryptohierarchies and self-organisation in the open-source movement. *Capital & Class*, (97), 143-157. Retrieved from Academic Search Complete database.
- Kemp, R. (2010). Open source software (OSS) governance in the organisation. *Computer Law and Security Report*, 26(3), 309-316. Retrieved from E-Journals database.
- Lee, J. (2010). ORGANIZING THE UNORGANIZED: THE ROLE OF NONPROFIT

ORGANIZATIONS IN THE COMMONS COMMUNITIES. *Jurimetrics: The Journal of Law, Science & Technology*, 50(3), 275-327. Retrieved from Academic Search Complete database.

- Long, J. (2009). Open Source Software Development Experiences on the Students' Resumes: Do They Count?--Insights from the Employers' Perspectives. *Journal of Information Technology Education*, 8229-242. Retrieved from ERIC database.
- Michaelides, G. (2006) Far From Equilibrium: Emergence, Self-Organization, and Diversity in the KDE Community, Ph.D thesis. Business School, University of Nottingham.
- Raymond, E. (1998) 'The cathedral and the bazaar'. *First Monday*, vol. 3, no. 3; online at www.firstmonday.org.
- RUGGERI, Lucia (2010). «ADR y ODR y su taxonomía. La identificación de caracteres». En: «Justicia relacional y métodos electrónicos de resolución (ODR): hacia una armonización técnica y legal» [monográfico en línea]. IDP. Revista de Internet, Derecho y Política. N.o 10. UOC.
- Solorzano, S. (2010). GNU/LINUX DESDE LA CONSTRUCCIÓN SOCIAL DE LA TECNOLOGÍA. (Spanish). *Revista Reflexiones*, 89(1), 87-89. Retrieved from Fuente Académica Premier database.
- VILALTA, Aura Esther (2010). «Resolución electrónica de disputas. Primeras reflexiones en torno a la necesidad de un marco normativo internacional». En: «Justicia relacional y métodos electrónicos de resolución (ODR): hacia una armonización técnica y legal» [monográfico en línea]. IDP. Revista de Internet, Derecho y Política. N.o 10. UOC. [Fecha de consulta: dd/mm/aa]. <http://idp.uoc.edu/ojs/index.php/idp/article/view/n10-vilalta/n10-vilalta>