US
UNIVERSITY OF SUSSEX

Sussex Research

Viral contracts or unenforceable documents? Contractual validity of copyleft licenses

Andres Guadamuz

Publication date
01-01-2004

Licence
This work is made available under the Copyright not evaluated licence and should only be used in accordance with that licence. For more information on the specific terms, consult the repository record for this item.

Citation for this work (American Psychological Association 7th edition)
Guadamuz, A. (2004). *Viral contracts or unenforceable documents? Contractual validity of copyleft licenses* (Version 1). University of Sussex. https://hdl.handle.net/10779/uos.23397563.v1

Published in
European Intellectual Property Review

Copyright and reuse:
This work was downloaded from Sussex Research Open (SRO). This document is made available in line with publisher policy and may differ from the published version. Please cite the published version where possible. Copyright and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners unless otherwise stated. For more information on this work, SRO or to report an issue, you can contact the repository administrators at sro@sussex.ac.uk. Discover more of the University's research at https://sussex.figshare.com/

Sussex Research Online

Viral contracts or unenforceable documents? Contractual validity of copyleft licenses

Article (Unspecified)

Citation:

Guadamuz González, Andrés (2004) Viral contracts or unenforceable documents? Contractual validity of copyleft licenses. European Intellectual Property Review, 26 (8). pp. 331-339. ISSN 0142-0461

This version is available from Sussex Research Online: http://sro.sussex.ac.uk/46011/

This document is made available in accordance with publisher policies and may differ from the published version or from the version of record. If you wish to cite this item you are advised to consult the publisher's version. Please see the URL above for details on accessing the published version.

Copyright and reuse:
Sussex Research Online is a digital repository of the research output of the University.

Copyright and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable, the material made available in SRO has been checked for eligibility before being made available.

Copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

http://sro.sussex.ac.uk

Viral Contracts or Unenforceable Documents? Contractual Validity of Copyleft Licences

ANDRÉS GUADAMUZ GONZÁLEZ
Andrés Guadamuz González, Lecturer, University of Edinburgh

This article attempts to ask the question of whether copyleft free software licences constitute valid legal contracts, in particular with regard to the fact that they may create obligations through a distribution chain. There is increasing interest in the non-proprietary licence model expressed in popular documents such as the general public licence ("GPL"), but not enough work has been done in asking perhaps the most important question of all: are these contracts enforceable? Is there really a viral transmission of obligations? To do this the GPL licence will be analysed to try to determine whether or not the terms included are contractually valid.

Introduction

The issue of non-proprietary software licences—such as the free software ("FS") and open source software ("OSS") licence models—is gaining interest in legal circles, a development that must be welcomed, taking into consideration that the phenomenon of open source/free software licensing was initiated with almost no intervention from legal scholars, leaving the legal profession once again to play catch-up in the fast-paced computer world.

Non-proprietary software licences pose some interesting questions from a traditional contractual law perspective because they create what some authors have defined as a viral contract, a contract that is to be transmitted through a distribution chain. The question must be asked whether the obligations arising from the initial licence are to be considered enforceable, or if any of these contractual terms should be suspect, particularly in jurisdictions where unfair contractual terms are strongly regulated. There have been a surprisingly small number of court cases generated by these licences—something that will undoubtedly change with the legal battle started by SCO—so a full study of the eventual validity or invalidity of the contractual copyleft clauses must be subject to an analysis by the academic community, something which has not been forthcoming in this side of the Atlantic.[1] The present work will attempt to redress the trend in Europe by looking at the contractual validity of the FS licensing (in particular copyleft licences) as opposed to the OSS model, which is less restrictive and whose contractual clauses are much less likely to generate judicial revision. The author is aware that this may prove difficult because of the lack of judicial review of the licences, but the main objective of the article is to start a much needed debate in this area.

Non-proprietary Software

Free software or open source?

It has become increasingly common to read the term "open source" applied to all types of software developed under a free distribution of the programme's source code.[2] It is important to stress that it is technically incorrect to refer to all of these models of software development as either open source ("OS") or free software ("FS"), which are the two main types of non-proprietary software, but not the only ones by far.

In general, there are some philosophical differences between both terms. In the strictest sense, the FS concept is centred on the concepts and philosophies of developing programs and distributing them freely.[3] This is not the place to provide a detailed description of the birth of the FS model,[4] but suffice it to say that FS is not new. It has been noted that software sharing is "as old as computers, just as sharing of recipes is as old as cooking".[5] It is vital to note that the meaning of the word "free" in FS does not mean free as in having no price, but rather free as in "freedom".[6] Stallman defines free software as having the following four characteristics:

- The freedom to run the program.
- The freedom to study how the program works by giving access to the source code.
- The freedom to redistribute copies.
- The freedom to improve the program and release those improvements to the public.[7]

As understood by the proponents of free software, programmers and other developers can charge for the software if it is their desire to do so, but the same underlying freedom behind the software must exist either if it is acquired for a fee or if it is not. The user

1 This is not the case in the United States, where there has been some interesting work in this area. See for example R. Gomulkiewicz, "De-Bugging Open Source Software Licensing" (2002) 64 U. Pitt. L. Rev. 75; D. Ravicher, "Facilitating Collaborative Software Development: The Enforceability of Mass Market Public Software Licences" (2000) 5 Va. J.L. & Tech. 11, 2000; and C. Nadan, "Open Source Licensing: Virus or Virtue" (2002) 10 Tex. Intell. Prop. L.J. 349.

2 Source code refers to the programming statements in a programming language that exists before the program is compiled into an executable application.

3 T. Stanco, "We are the New Guardians of the World", May 16, 2001, *http://lwn.net/daily/guardians.php3*.

4 There are several works that achieve this, see: J. Naughton, *A Brief History of the Future* (1999), pp.172–174; H. E. Pearson, "Open Source: The Death of Proprietary Systems?" (2000) 16/3 *Computer Law & Security Report* 151–156; and R. Stallman, "The GNU Project", 1998, last updated October 24, 2001, *www.gnu.org/gnu/thegnuproject.html*.

5 Stallman, *ibid.*

6 Or as it is often stated in OS and FS circles, free must be understood as in freedom, not as in beer.

7 R. Stallman, "The Free Software Definition", 1996, last updated October 17, 2001, *www.fsf.org/philosophy/free-sw.html*.

must still be able to have all of the freedoms described, with access to the source code as the most basic requisite.[8] The Free Software Foundation ("FSF") goes as far as to state that

> "The freedom to use a program means the freedom for any kind of person or organization to use it on any kind of computer system, for any kind of overall job, and without being required to communicate subsequently with the developer or any other specific entity".[9]

This freedom is protected by the adoption of a restrictive licensing model that makes use of existing copyright legislation that guards the source code from proprietary software developers who want to copy it, adapt it and include it in their own programmes. This licensing model will be explained in more detail later.

Open source is closely related to the free software development, but it does contain a different emphasis with regard to the freedoms involved. The term "open source" was coined during a strategy meeting in February 1998 in Palo Alto, California, by a group of software developers with links to the Linux operating system.[10] The group met to plan a new strategy in response to the groundbreaking announcement by Netscape that they would be opening their operations and providing the source code of the popular Netscape internet browser to the public, prompted by fierce competition from Microsoft.[11] They believed that this gesture would give them a precious opportunity to sell the Open Source development approach to the corporate world.[12]

The need to create a new term to define this viewpoint had become evident because, until then, the prevalent way to describe all output produced by the non-proprietary approach was by using the expression "free software", based mostly on the FS philosophy described. It was apparent to many software developers that this movement had a tarnished reputation in the business world as a result of the more radical ideas held by people linked to the FSF.

In the widest sense, open source is the opposite of "closed source", the traditional proprietary approach to software development in the commercial world. Closed source is software "in which the customer gets a sealed block of bits which cannot be examined, modified, or evolved".[13] The main idea behind Open Source is to provide software for which the source is available for examination, modification and peer-review. The official definition of OSS came out of the original meeting, and was based on the Debian Free Software Guidelines, a licensing model that accompanies the Debian GNU/Linux system, a Linux distribution.[14] This has generated an Open Source Definition ("OSD"), which includes a recommended set of clauses that an OSS licence should contain.[15]

There are several similarities between OSS and FS licences. In fact, some OSS licences have been deemed to be compatible with FS principles, and vice versa.[16] Nevertheless, there are several differences between the FS and OSS philosophies. The main difference is the fact that OSS does not impose in its licences obligations for derivative software to be kept free—such as the case of the copyleft licences that will be explained later—a practice that has been deemed too restrictive and commercially unfriendly by its proponents. One of the many complaints that FS advocates make of the open source philosophy is that it is not strong enough in trying to keep software free, and that it simply allows anybody to name their software "open source" even if it is not.[17] This is a problem that has been partially acknowledged by OSS proponents, which is why they have created the Open Source Initiative ("OSI") certification. This certification is given to those licences that follow the Open Source definition and provides a certification to inform the public that the software is indeed open source.[18] There are many different OSI certified licences,[19] and it is important to point out that this list includes all sorts of FS licences that comply with their definitions and guidelines.

Regardless of whether one prefers the term open source or free software, it has become important to use a phrase that encompasses all sorts of definitions within this development model. The author prefers the use of the phrase "non-proprietary" as an umbrella term which refers to all the different sub-categories encompassed by this movement, and which would ultimately mean that the source code is made available for all sorts of derivative purposes. Another acceptable term is "Libre Software"—now in use by the Information Society Directorate General of the European Commission[20]—as the Spanish and French word "libre" has a more precise meaning than its equivalent in English, and encompasses better the philosophy behind non-proprietary development systems. Another valid way of describing this is to refer to free and open source software ("FOSS"), or even free, libre and open source software ("FLOSS"). The distinction may seem academic, but it is important because the use of each of

8 R. Stallman, "Selling Free Software", 1996, last updated August 8, 2001, *www.fsf.org/philosophy/selling.html*.

9 Stallman, n.7 above.

10 Open Source Initiative, "History of the OSI", 2001, *www.opensource.org/docs/history.html*.

11 It may even be said that Microsoft's competitive tactics against Netscape were excessive and even predatory, and they prompted the antitrust case brought by the US Department of Justice against Microsoft. A roadmap to the case can be found here: *www.stern.nyu.edu/networks/ms/top.html*.

12 Open Source Initiative, n.10 above.

13 E. Raymond, "Keeping an open mind", March 1999, *http://tuxedo.org/~esr/writings/openmind.html*.

14 The guidelines can be found here: *www.debian.org/social_contract.html#guidelines*.

15 The OSD can be found here: *www.opensource.org/docs/definition.php*.

16 For examples of these, see: Free Software Foundation, "Various Licences and Comments about Them", 1999, last updated June 15, 2003, *www.fsf.org/licences /license-list.html*.

17 R. Stallman, "Why 'Free Software' is better than 'Open Source' ", 1998, last updated August 20, 2001, *www.fsf.org/philosophy/free-software-for-freedom.html*.

18 Open Source Initiative. "OSI Certification Mark and Program", April 30, 2001, *www.opensource.org/docs/certification_mark.html*.

19 At the moment there are a total of 43 OSI certified licences.

20 Working group on Libre Software, "Free Software / Open Source: Information Society Opportunities for Europe?", April 2000, *http://eu.conecta.it/paper/paper.html*.

these terms presupposes a specific development philosophy behind the software. The author also believes that the use of either FOSS or non-proprietary software is better than to use FS or OSS separately. This is because they encompass all different types of philosophies and distributions, ranging from commercial variations of the non-proprietary model to those that are offered freely to the public.

Copyleft licensing

From the many different types of FS recognised by most non-proprietary proponents, the most popular type of FS distribution is by means of copyleft licensing —with surveys estimating more than 70 per cent of non-proprietary software uses copyleft licences as their main contractual mechanism.[21] Copyleft is free software with a twist; it maintains the general freedoms awarded to FS users, but by acquiring a copyleft program, the user has to agree to a licence agreement that states that that the software will not be used to develop proprietary applications derived from it.[22] The FSF has a specific definition of what a proprietary program is for the purposes of copyleft licences. According to them, a proprietary program is one that is "software that is not free or semi-free. Its use, redistribution or modification is prohibited, or requires you to ask for permission, or is restricted so much that you effectively can't do it freely".[23]

Copyleft was created from a perceived need to protect the fruits of non-proprietary development. After several years of producing computer programs with a sharing mentality and offering the code to the public, developers realised that some software developers had started using FS outputs in a parasitical fashion, obtaining the source code, tweaking it and selling it as commercial proprietary software with very low production costs.[24] Copyleft became the contractual solution to stop companies profiting from non-proprietary products by distributing software that must remain free.[25]

For GNU software, the recommended contract to use is the general public license ("GPL"), which is a standard contract that ensures that the software code is passed on, and anyone who redistributes the software —with or without changes—must also pass along the freedom to further copy and change it. This places a burden on the person transferring the software; the burden is that the software must remain "free", as defined by the FSF and the GPL. This is different from just placing software in the public domain because the work maintains its copyright protection.[26]

The GPL is the main exponent of the legal framework that sustains the copyleft system. It reads as a mixture of a legal contract and an ideological manifesto. The preamble to the work states clearly some of the most common beliefs of free software and the non-proprietary approach, with several admonitions about the meaning of the word "free". The main point is that the source code must be made available to the users. The preamble says:

> "For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights."[27]

The licence specifies that this is achieved by two means: by protecting the software by means of copyright; and by providing the users with a licence that gives them the freedom to use and modify the software in any way they see fit. The main body of the licence reiterates these ideas. Section 1, for example, states:

> "1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this Licence and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program."[28]

This section also mentions that the user can make monetary charges when passing the copy, which is also consistent with the general free software characteristic that does not discriminate against commercial software as long as it is not proprietary commercial software.

Many of the provisions of the GPL can be found in other non-proprietary licences, including several OSS ones. What makes the GPL unique is s.2(b), as this is where the restrictions against using the software to create commercial software are specified. The section reads:

> "2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions: . . . b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License."[29]

What this means is that any software developed by using the open source code of the copyleft programme must not charge for the derivative product, and most importantly, must ensure that the GPL is transferred to further users of the derivative software. This type of licence has been aptly named a "viral contract" by Professor Radin, defining them as "contracts whose obligations

21 M. O'Sullivan, "Making Copyright Ambidextrous: An Exposé of Copyleft" (2002) 3 *Journal of Information, Law and Technology (JILT)*, 2002, *http://elj.warwick.ac.uk/jilt/02-3/osullivan.html*.

22 R. Stallman, "*What is copyleft?*", 1996, last modified 05/11/2001 (???), *www.fsf.org/copyleft/copyleft.html*.

23 R. Stallman, "Categories of Free and Non-Free Software", last modified December 29, 2002, *www.gnu.org/philosophy/categories.html*.

24 R. Stallman. "Copyleft: Pragmatic Idealism", 1998, last updated August 26, 2002, *www.fsf.org/philosophy/pragmatic.html*.

25 H. Meeker, "Why you need to understand open source licences" (2001) 19 *International Technology Law Review* 24–27.

26 P. Lambert, "Copyleft, copyright and software IPRs: is contract still king?" [2001] E.I.P.R. 165–171.

27 Free Software Foundation, "GNU General Public License", last modified July 15, 2001, *www.fsf.org/licences/gpl.html*.

28 *ibid.*

29 *ibid.*

purport to 'run' to successor of immediate parties".[30] These contracts would then spread in a viral form, as the licensee must include the terms of the GPL in any subsequent licence they will include in their derivative work because that obligation is part of the contract, and then those subsequent licensees will have to impose the same contractual terms in further licences that they perform, *ad perpetuam*.

The restrictions imposed by copyleft would seem to go against some of the principles of free software because of the viral imposition of restrictions and obligations, which denies the very freedom of doing what one desires with the software—and the FS proponents should face the fact that this may very well include the freedom to commercialise and profit from the subsequent use of the code. The use of non-proprietary software to create a proprietary or "closed source" software may be morally suspect, but one cannot elevate freedom to the highest pedestal and begrudge those who will use that freedom for purposes that are philosophically and politically adverse to those of the creator of the program.

Another conundrum that must be understood is the distinction between contractual enforceability and copyright protection awarded to computer programs. It could be said that copyleft licences create a double-pronged protection of the software. On one hand they pose contractual restrictions in the shape of a licence, in particular by the contractual enforceability of the GPL licence and its clauses. On the other hand, works protected by copyleft use copyright protection to be able to make this licence enforceable. This certainly creates an interesting relationship between the predominant nature of copyright, which is directed towards the protection and regulation of ownership, and a system that seems to advocate the exact opposite. The irony that such a contrary system requires copyright to survive cannot possibly be lost, and it is something that Stallman and many copyleft advocates have trouble answering, even though the websites belonging to the free software advocates are filled with essays that criticise copyright and intellectual property.[31] Regardless of these problems, the restrictions imposed by copyleft have a good number of outspoken defenders set on furthering the copyleft model in spite of any opposition.[32]

Validity of the GPL Licence

The viral nature of copyleft licences has generated a considerable amount of interest in circles that transcend software development. The idea of sharing materials is not new, and has been made more evident by the chaotic and sometimes anarchic nature of the internet.[33] However, shared materials tend to suffer from the possibility of third parties that use the freely acquired information to turn them into proprietary works. That is why many different organisations are turning to the copyleft model to protect works that are being freely shared online. One such project is the OpenContent Licence ("OPL"), a collaborative effort that sets a similar licence to the GPL, ensuring that shared works will continue to remain free to subsequent users.[34] A more ambitious project is Creative Commons, which offers a wide range of licences applicable to all sorts of creative material.[35] In the area of biotechnology, there have been some suggestions that the copyleft model could be used to protect the public results of the human genome race that are being placed in the public domain by researchers, something that has been suggested by a leading member of the Human Genome Consortium, although the idea has never been implemented.[36]

Despite the growing popularity of such licences, the actual validity of the licences—and in particular of the copyleft clauses—has yet to produce a court ruling during its relatively short history. Until recently, there had been no court cases against non-compliance with a copyleft licence, and the few incidents that have arisen had been dealt swiftly with cease-and-desist letters to those parties suspected of producing proprietary software.[37]

This all changed when a developer of non-proprietary database software named MySQL sued NuSphere, a software company that it believed was using its source code to produce proprietary software—something that contravened the terms of the GPL.[38] This file was issued in response to a suit filed by NuSphere claiming "breach of contract, tortious interference with third party contracts and relationships and unfair competition".[39] Unfortunately this case was settled out of court; hence the GPL did not receive a judicial review in this occasion. However, this was only the opening shot in what is set to become one of the largest and most complex legal battles that the software industry has ever seen. The legal question about the validity of copyleft licensing models broke spectacularly in legal circles in March 2003 when the SCO Group—a well-known software developer of UNIX-related products—filed a lawsuit against IBM alleging that the company was

30 M. J. Radin, "Humans, Computers, and Binding Commitment" (2000) 75 Ind. L.J. 1125.

31 For example, see: Free Software Foundation, "Reevaluating Copyright: The Public Must Prevail", 1996, last updated January 8, 2001, *www.fsf.org/philosophy/reevaluating-copyright.html*.

32 For one such defender, see: E. Moglen, "Anarchism Triumphant", *First Monday*, Vol.4 No.8, August 2, 1999, *www.firstmonday.org/issues/issue4_8/moglen/index.html*.

33 For more on this subject, see: A. Guadamuz, "The New Sharing Ethic in Cyberspace" (2000) 5/1 *Journal of World Intellectual Property* 129–139.

34 The licence can be found here: *www.opencontent.org/opl.shtml*. Other interesting copyleft licences include the Design Science License, the Open Audio Licence and even Open Cola, the world's first copyleft fizzy drink. See G. Lawton, "The Great Giveaway", *New Scientist*, *www.newscientist.com/hottopics/copyleft/copyleftart.jsp*.

35 *http://creativecommons.org*.

36 J. Sulston, "Intellectual Property and the Human Genome", *Global Intellectual Property Rights* (Drahos and Mayne ed., 2002), pp.561–573.

37 G. Moody, *Rebel Code* (2001), p.313.

38 A FAQ about the case can be found here: *www.mys.com/news/article-75.html*.

39 *ibid.*

infringing its intellectual property over the UNIX kernel.[40] The full details of the suit are still sketchy because SCO is keeping some of the most detailed information of the code they allege to have been protecting as a close secret, not letting it be known which part of the code it claims ownership of.[41] However, it is known that SCO claims that back in 1985 AT&T and IBM signed a contract to produce a version of UNIX called AIX. In 1995, SCO purchased all of the intellectual property related to UNIX from AT&T, hence the claim they have filed against IBM. It would seem that SCO is somehow making claims that they own part of the code for AIX, or that they own some other part of the UNIX kernel code that is used in most machines running Linux distributions. Furthermore, SCO threatened to sue every corporate Linux user for copyright infringement,[42] claiming that any Linux user must purchase a licence from them. This threat finally came to fruition in March 2004 when they sued DaimlerChrysler and auto parts retailer AutoZone, two corporate Linux users.[43] It is too early to ascertain the strength of SCO's arguments, but it has become clear that this case has increased the stakes in the financial importance of copyleft licences, and hence the importance in making sure that the licence terms are valid.[44]

It is also important to evaluate the validity of copyleft licences from a European perspective, as most of the existing literature in this subject come from the United States.[45] There are reasons to evaluate the validity of copyleft licences from a European contract law perspective in at least two different fronts that vary from the American approach: unfair contractual terms and the rights of passing to a third party. Other legal aspects, such as the competition law and the copyright aspect of the protection of GPL works will be analysed as well, as they vary in some aspects from the American approaches.

Unfair contractual term

The first concern for the consideration of the validity of the copyleft clauses must be to ask if they are unfair according to European consumer protection legislation. Most jurisdictions have different public policy restrictions to contractual terms, the most common being restrictions against terms that will give away basic human rights,[46] but beyond these basically recognised principles the range of restricted or excluded terms varies from one jurisdiction to another.[47] It is because of the wide variation in this area of contract law that the European Union felt the need to harmonise the different approaches to unfair terms across Member States. Consumers in the EU are now subject to a wide-ranging regime designed to protect them from unfair terms in a variety of circumstances in which they are presented with pre-formulated standard contracts, a regime implemented in the Unfair Terms in Consumer Contracts Directive ("the Directive"),[48] which specifies what an unfair contractual term is, and sets a number of considerations by which clauses will be analysed to test for unfairness. The Directive also provides a non-exhaustive list of some terms that will be considered unfair, none of which applies directly to copyleft licences.

The GPL contains several different clauses that may be considered in light of the existing unfair terms legislation. The most likely candidate for this is the limitation of liability expressed in ss.11 and 12 of the licence. Even though these are the sections most likely to be found unfair, they will not be analysed in this article as they are similar to any other limitation of liability, and likely to receive the same analysis as those.[49]

The main question then is to analyse whether or not the copyleft clause included in the GPL is unfair or not. There are many issues to consider when asking this question. The first one is whether the licensee of GPL-protected software should be considered a consumer as understood by the definition provided by Art.2(b) of the Directive, which states that a consumer will be any natural person who "is acting for purposes which are outside his trade, business or profession". This is a very broad definition of consumer, and even though the wording of the Directive would seem to exclude legal persons, it must be underlined that courts have generally taken a very broad interpretation as to what a consumer is, even to include companies.[50] The common interpretation of this requirement will be that the person entering into a standard contract—such as a software licence—will be considered to be a consumer if they are not signing the contract in the regular course of dealing in that business. It would be fair to assume that if a software firm develops a software program and licenses it to another software firm using the GPL, the licensee firm will probably not be considered a consumer for the purposes of the Directive. On the other

40 P. Galli, "SCO Group Slaps IBM With $1B Suit Over Unix", *E-Week*, March 10, 2003, *www.eweek.com/article2/0,3959,922913,00.asp*.

41 J. Harvey and T. McClelland, "SCO v. IBM: The Open Source benefits and Risks are Real" (2003) 20/9 *Computer & Internet Lawyer* 1.

42 P. Galli, "SCO Warns Linux Users of Legal Liability", *E-Week*, May 14, 2003, *www.eweek.com/article2/0,3959,1149623,00.asp*.

43 T. Weiss, "SCO Sues Two Linux Users, Warns About Further Action", *Computerworld*, March 8, 2004, *www.computerworld.com/softwaretopics/os/story/0,10801,90868,00.html?f=x72*.

44 The most recent developments in this case can be followed here: *www.groklaw.net*.

45 See for example: D. Kennedy, "A Primer on Open Source Licensing Legal Issues: Copyright, Copyleft and Copyfuture" (2001) 20 St Louis U. Pub. L. Rev. 345.

46 Radin, n.30 above.

47 In the United Kingdom for example, the Unfair Contract Term Act 1977 ("UCTA") contains an exhaustive list of unfair terms, which include exclusion, limitation and indemnity clauses.

48 Council Directive of April 5, 1993 93/13 on unfair terms in consumer contracts [1993] O.J. L95/29. The Directive has already been implemented in the United Kingdom in the shape of the Unfair Terms in Consumer Contracts Regulations 1999 ("UTCCR").

49 For more on exclusion of liability terms, see R. G. Lawson, *Exclusion clauses and unfair contract terms* (6th ed., 2000).

50 Most recently in the United Kingdom one can find examples of this in *SAM Business Systems Ltd v Hedley & Co* [2002] EWHC 2733.There are several older examples of this, such as *R&B Customs Brokers Ltd v United Dominions Trust Ltd* [1988] 1 W.L.R. 321; and even Cass. Civ. 1re, April 28, 1987. Most notably for software purposes are *St Albans City & District Council v International Computers Ltd* [1996] 4 All E.R. 481.

hand, an individual consumer who has acquired some copyleft-licensed software could possibly make a strong case arguing that he is signing the licence as a consumer. This is of course a general interpretation, and the circumstances of each contract must be individually determined on a case-by-case basis.

Assuming that the licence is considered to be a consumer contract as described, there is still a need to determine whether the term itself is unfair. Article 3(1) of the Directive specifies that:

> "A contractual term which has not been individually negotiated shall be regarded as unfair if, contrary to the requirement of good faith, it causes a significant imbalance in the parties' rights and obligations arising under the contract, to the detriment of the consumer".

A term will be considered not to have been negotiated individually if it has been drafted in advance and the consumer did not have a say in the terms of the final contract.[51] This definition is at the heart of any contractual dispute that may arise by the application of the Directive, and its interpretation is the one that offers more problems as it can be considered as using a very open-ended requirement, such as the often nebulous expression "good faith". In the United Kingdom, the test for unfairness as expressed by the Directive has been established by *Director General of Fair Trading v First National Bank Plc*.[52] According to this ruling, the consumer must prove that there has been bad faith on the part of the undertaking in the drafting of the contract, that there is a significant imbalance in the obligations and powers of the parties, and that such imbalance must be detrimental to the consumer. The court in this ruling specified that good faith would be present if the contract was signed with fair and open dealing. Openness means that the term must be clear, legible and not contain hidden pitfalls; and fair dealing would have to be understood as that the supplier should not take advantage of the other party's relatively weak position. It is important to note as well that some commentators suggest that the concept of "good faith" should be understood in accordance to civil law principles,[53] and as such many different aspects must be taken into consideration, for example the gravity of the imbalance, the social position of the parties and the way in which the term in question came into existence.[54]

Analysing the copyleft clause with the requirements presented by this ruling, one could say that there appears to be an imbalance in the obligations of the parties as the licensee will have to use the GPL and cannot profit from derivative works. This imbalance could also be assumed to be detrimental to the consumer as it is imposing the responsibility of not being able to use the work in whatever way it is desired. However, one must say that this is precisely the same type of imbalance that exists in every other copyright-based software licences, and hence it would be difficult to find it unfair.

The main question will be in trying to determine if there has been good faith by the drafter of the licence. This is more difficult to ascertain given the test of good faith presented above. In the case of the GPL, the test does not appear to be met. The copyleft clause is clear enough, does not contain hidden pitfalls, and the software owner is not taking advantage of the relatively weak position either because the consumer is always free not to use the software if he so desires, and is even free to look for similar software that does not use copyleft licences.

Based on this brief analysis of the copyleft contract term and the existing European unfair contract legislation, it would seem that the GPL copyleft clause is valid, as there are too many uncertainties as to whether or not a court would interpret this clause in favour of a licensee on the basis of the existence of good faith. It must also be assumed that the copyleft clause will be valid as it does not fall into any of the specified unfair terms provided in the Annex to the Directive. However, the question must remain open until the first case testing the validity of this type of licence comes up. Given the amount of money involved in software development, it is likely that at some point copyleft will indeed receive some judicial review.

Passing obligations and rights to third parties

Another interesting legal issue that arises when considering the validity of GPL clauses is the problem of passing obligations to third parties. The legality of this practice is usually covered under the English contract law concept of the privity of contracts, of which there are two rules, one for passing burdens and one for passing benefits.

The first rule exists under traditional privity doctrine, where "a third party cannot be subjected to a burden by a contract to which he is not a party".[55] This general principle is still in effect in most jurisdictions and responds to the reasonable principle of legal security by not allowing parties to place contractual burdens that they are not aware of. Wherever this practice is permitted, it is usually well regulated.[56] The question must be asked of whether the GPL constitutes the imposition of a burden to third parties. The initial response would be negative, as the imposition of the clause is done on a one-to-one basis. If one does not agree with the copyleft clause, then it is only logical that one should not use the software; and certainly one should not use it to create a derivative product.

If the passing of obligations is generally not accepted in contract law, what happens to the passing of benefits? There is a second controversial privity rule in English law which does not allow a third party to benefit from the contract, although the rule has been largely modified in England as to render it practically inexistent.[57] It

51 Directive 93/13, Art.3(2).

52 *Director General of Fair Trading v First National Bank Plc* [2001] UKHL 52; [2002] 1 A.C. 481.

53 E. McKendrick, *Contract Law* (4th ed., 2000), p.369.

54 Some of these principles in civil law can be seen in several continental cases, such as *Saladin/HBU*, Hoge Raad, NJ 1967 261 (G. J. Scholten). For a more complete work on the subject of good faith in civillLaw, see *Good faith in European contract law*, (R. Zimmermann and S. Whittaker ed., 2000).

55 McKendrick, n.53 above, at p.133.

56 Radin notes for example where passing burdens are accepted in competition law and in public policy issues, see: Radin, n.30 above, at p.135.

57 This was done by the Contracts (Rights of Third Parties) Act 1999.

is important to point out that this second privity rule exists in civil law jurisdictions,[58] where third-party rights (known in Scotland as *jus quaesitum tertio*), have been an integral part of contract law.[59]

The relevance of third-party rights to copyleft results in the question of whether the originator of a program licensed under the GPL may sue a licensee who is located further down the software distribution chain for breach of contract. Assuming that A is the software creator and B is the copyleft licensee and B licenses the software to C using the GPL; could A sue C for contractual breach if C does not comply with the copyleft clause? Contractually speaking, one would have to assume that for A to successfully sue C; A must have a third-party right arising from the contract between B and C, which appears to be an invalid proposition.

The possible applicability of third-party rights to copyleft can be better understood in the famous Scottish case of *Beta Computers v Adobe Systems*.[60] In this case, Beta Computers provided a copy of software authored by a third party called Informix, for which they had a licence. The court in this case found that Informix—although not part of the contract between Adobe and Beta—had a third-party right. This position has been adequately criticised by MacQueen, who says that when the subject of a software transaction is a licensing agreement, third-party rights cannot possibly apply as a licence grants rights *by* the third party, it does not create rights *to* the third party, which is the doctrinal requirement of third-party rights.[61] There cannot be much doubt that in the case of copyleft licences, the author's rights arise from the licence itself and the contractual provisions contained within. It will be seen later whether the author could sue under copyright, providing the code has been copied without a licence, but it would be more difficult to state that the author could sue for a broken contractual term contained in the licence. The contractual validity of the copyleft clause would then work on a one-to-one basis, where only the two parties involved could sue each other and there would be no possibility of involving third parties, even if the third party is the author.

Copyright infringement

The analysis above would seem to indicate that the author or owner of a work that has been licensed using copyleft will find it difficult to sue subsequent users of the software down a distribution chain for contract breach. Yet the question still remains whether the author can sue for copyright infringement. The answer to this is much more straightforward than the contractual analysis.

Using the same example cited above, let us assume that A is the software owner and B is the copyleft licensee, and that B licenses the software to C using the GPL. C modifies the software and releases a proprietary version of it by closing the source code to subsequent users. Could A sue C for copyright infringement? The answer is a definitive yes, as copyright is less preoccupied with who licensed the software to C, but the emphasis would be whether or not C is committing actions that would be considered as infringing A's copyright. The question then would become one of infringement and originality, possibly hinging on the question of whether or not C has done enough work to the original source code to be considered an original work.

This is a much better explored area of copyright law. Computer software is awarded copyright protection as a literary work if it is considered to be an original work.[62] The question of originality has been long discussed by the courts, but there is agreement that an original work is one that demonstrates the use of skill and labour by the author, in short, that "that it should originate from the author".[63] Even though the originality requirement states that the work should not be copied in its entirety, courts have recognised that a certain amount of copying is acceptable. For example, copying of the drawing of existing designs has been deemed to be original in some instances.[64] When copying exists, the copying must fulfil the long-standing qualitative test to determine whether the copying has been substantial.[65]

In computer software, the courts have been following the general qualitative test in cases of copying from another work. In both *Richardson Computers v Flanders*[66] and *Ibcos v Barclays*,[67] the courts found that if there had been any copying from a protected original work, there had to be an analysis of whether such copying had been substantial. It is important to stress that the test is for qualitative copying, not quantitative. There will be some consideration about the quantity of the work copied,[68] but even if this is minimal it may result that the copying may be deemed to be substantial. This is evident in the case of *Cantor v Tradition*,[69] where copying of original source code took place by former employees of a financial services company. In this case, expert witnesses found that only 2 per cent of the original source code had been copied, accounting for only 2,952

58 And in mixed legal systems such as Scotland.

59 In France, for example, privity of contract is qualified by Art.1121 of the *Code Civil*, which allows third-party rights. In Germany, Art.328 of the *Burgerliches Gesetzbuch* allows for the performance of rights by third parties. Another example can be found in Art.2.115 of the Principles of European Contract Law; see European Commission on Contract Law, *Principles of European Contract Law: Part 1: Performance, Non-performance, Remedies* (O. Lando and H. Beale ed., 1995).

60 1996 S.C.L.R. 587.

61 For an excellent attack to this ruling, see: H. L. MacQueen, "Software Transactions and Contract Law", *Law and the Internet: Regulating Cyberspace* (Edwards and Waelde ed., 1997).

62 s.3(1)(b) UK CDPA 1988.

63 *University of London Press Ltd v University Tutorial Press Ltd* [1916] 2 Ch. 601.

64 For examples of this see *The Duriron Co Inc v Hugh Jennings & Co Ltd* [1984] F.S.R. 1; and *Interlego v Tyco Industries* [1989] A.C. 217; [1988] 3 All E.R. 949.

65 Existing in common law since *Bleistein v Donaldson Lithography Co* 188 U.S. 239, 250 (1903).

66 *John Richardson Computers Ltd v Flanders and Chemtec Ltd* [1993] F.S.R. 497.

67 *Ibcos Computers Ltd v Barclays Mercantile Highland Finance* [1994] F.S.R. 275.

68 For which software may be helpful in analysing the number of lines of code copied, for instance software such as MOSS: *www.cs.berkeley.edu/%7Eaiken/moss.html*.

69 *Cantor Fitzgerald International v Tradition (UK) Ltd* [1999] Masons C.L.R. 157.

lines of code out 77,000.[70] The lines of code were deemed to be of importance for some modules in the resulting software, but the copying was not considered substantial enough to grant the infringement case, but was enough for the copier to agree to take financial responsibility for the infringed code and offer to pay for it. Nevertheless, the fact that some of the copying was even considered in the ruling must send signals to potential copiers of non-proprietary software about their chances in court.

Given the state of the rulings in software copyright infringement, it appears that if a copyright author or owner can prove to a court that a proprietary copy of their original software has been infringed, then it will not matter just how they obtained the software, and it will certainly not matter if they are further down in a chain of distribution. If a programmer uses substantial sections of code belonging to a copyleft program, that programmer will still be subject to legal action by the author. There may also be a question about moral rights, but these considerations fall outside of the scope of the present article.[71]

On a side note regarding enforcement of copyright, it is interesting to point out that the FSF recommends to all those programmers using the GPL that they should assign copyright ownership of their works to the FSF because in that way they can enforce the licence better in case of infringement.[72]

Competition law

There is one final area that may provide validity problems for copyleft licences. Even though these licences do not impose obligations to third parties as the licence is passed to a single licensee at the time, it is less clear whether such restrictions could be considered anti-competitive in accordance to European competition rules, as it could be found that the imposition of the copyleft clause, even if done on a one-to-one basis, could be found to be anti-competitive.

EC competition rules have a set of provisions that impose certain restrictions on the passing of obligations through a distribution chain which may create anti-competitive restrictions on the recipient; this is evident in the regulation and implementation of competition law in the area of licensing and vertical agreements. Vertical agreements in the competition sense "are those entered into between undertakings whose relationship is complementary, such as manufacturer and distributor or licensor and licensee".[73] An example of a regulated vertical agreement is the existing set of restrictions in the area of technology transfer licensing, where a number of impositions down a distribution chain are blacklisted.[74]

There is an ongoing debate about the seriousness of vertical agreements that impose restrictions through a distribution chain, as economists in the 1980s started seeing vertical restraints in a positive light[75] despite some early emphasis by the European courts on clamping down on these types of agreements.[76] The debate has continued, with the official position steadily moving towards a less restrictive approach towards vertical restrictions. In fact, a Green Paper by the European Commission found that

> "distribution agreements raise special difficulties because they are usually something of a two-edged sword. They can be a useful way for a firm to penetrate a new market and to sell its products effectively. But they can also be used to prevent outsiders from entering a market, and so perpetuate the compartmentalization of the Community".[77]

National implementation of the European rules seems to vary as well. It has been generally commented that the United Kingdom has a less strict application of vertical restrictions than the rest of Europe, with the emphasis being placed on whether there will be a sanction for such practices being placed on undertakings with considerable market dominance that is used in detriment to the consumer.[78] Having said this, licensors of copyleft software are not likely to posses the market share to be considered dominant by any stretch of the imagination. It is also very unlikely that these licences would be considered to impose a considerable damage on the consumer, as they always have the option to purchase non-copyleft software. Another important consideration is that copyleft licences do not fall into the four main types of vertical agreements listed by the European Commission in their Green Paper (exclusive distribution, exclusive purchasing, selective distribution and franchising).[79]

In this light, it seems unlikely that copyleft licences will be considered anti-competitive by the courts and regulators, but this is an area that demands more scrutiny.

Conclusion

An initial look at the problem of the validity of copyleft licences (particularly the GPL), seems to provide a positive response to this novel and ingenious software distribution model. There are some unanswered questions, in particular with regards to privity of contracts, but as long as the contractual chain is kept at the most simple relationship between licensor and licensee, the validity of the copyleft clause appears to be sound. Software authors interested in making sure that their works are distributed to the largest number of people without fear

70 I. Lloyd, *Information Technology Law* (3rd ed., 2000), p.411.

71 For an excellent look at moral rights and OSS, see A. Metzger and T. Jaeger, "Open Source Software and German Copyright Law" (2001) 32 I.I.C. 52–74.

72 E. Moglen, "Why the FSF gets copyright assignments from contributors", *www.gnu.org/copyleft/why-assign.html*.

73 D. G. Goyder, *EC Competition Law* (3rd ed., 1998), p.13.

74 See: Commission Regulation 240/96 on the application of Art.85(3) of the EC Treaty to certain categories of technology transfer agreements [1996] O.J. L31/2. For more on vertical restraints, see M. Furse, *Competition Law of the UK & EC* (1999), pp.104–112.

75 Furse, *ibid.* at p.105.

76 See for example *Consten and Grundig v Commission*, Joined Cases 56 and 58/64 [1996] E.C.R. 299; C.M.L.R. 418.

77 European Commission, *Green Paper on Vertical Restraints in EC Competition Policy*, COM (96) 721, *http://europa.eu.int/en/record/green/gp9701/vrtocen.htm*.

78 Furse, n.74 above, at p.105.

79 European Commission, n.77 above.

of commercial interests placing a fence over their works should definitely consider the copyleft model as a successful example, but some reservations may still be healthy until copyleft licences are finally tested in court. Authors may also be willing to consider other types of licences because the GPL, although apparently valid, suffers from drafting errors and too many revisions. In this respect, a good look at licences offered by other suppliers, such as the Creative Commons project, may be advisable.

Something else that would be welcome is to see more European versions of copyleft licences. Although this article has concluded that the GPL seems to be valid in accordance to UK law, country-specific licences will have less problem in being considered valid in different jurisdictions.