

# Paradigm Shift in European Intellectual Property Law? From Microsoft to Linux

Lucie GUIBAULT\*

*Lex Electronica*, vol. 10, no 3, Hiver/Winter 2006  
<http://www.lex-electronica.org/articles/v10-3/guibault.htm>

<b>INTRODUCTION.....</b>	<b>2</b>
<b>1. OPEN SOURCE AND COPYRIGHT LAW .....</b>	<b>4</b>
1.1. FREEDOM TO USE.....	5
1.2. FREEDOM TO REPRODUCE .....	9
1.3. FREEDOM TO MODIFY .....	10
1.4. FREEDOM TO (RE)DISTRIBUTE.....	13
1.5. ROYALTY FREE DISTRIBUTION .....	18
<b>2. OPEN SOURCE AND PATENT LAW .....</b>	<b>20</b>
2.1. OPEN SOURCE AND PATENTED SOFTWARE.....	21
2.2. OPEN SOURCE PATENTING STRATEGY .....	25
2.3. OPEN SOURCE LICENSING STRATEGY .....	27
<b>3. ENFORCEMENT OF OPEN SOURCE LICENCES .....</b>	<b>29</b>
<b>CONCLUSION.....</b>	<b>32</b>

---

\* (LL.M. Montréal en LL.D. Amsterdam) Senior researcher at the Institute for Information Law (IViR) of the University of Amsterdam (L.Guibault@uva.nl). This paper is adapted from O. van Daalen and L. Guibault, *Unravelling the Myth around Open Source Licences – An Analysis from a Dutch and European Perspective*, The Hague, T.M.C. Asser Press, forthcoming end 2005 or beginning 2006.

## Introduction

Open source or free software<sup>1</sup> is actually as old as the software industry, but its use is becoming more and more widespread among businesses, governments, and the public at large. Open source software licences are based on two fundamental principles: the possibility for users to use the software for any purpose and to modify and redistribute it without prior authorisation from the initial developer. Some open source software licences, like the General Public Licence (GPL), also impose a corollary obligation on the licensee: to make the source code available to other developers.<sup>2</sup> The idea behind this form of licensing is that when programmers can read, redistribute, and modify the source code for a piece of software, the software evolves.<sup>3</sup> Perhaps more than any other type of software, open source software is, as a result of its characteristic licensing scheme, the engine of collaborative creation. The very fact that the software may be freely used, modified and redistributed encourages subsequent developers to make their own contribution to an existing piece of software, by correcting errors, or by enhancing the software's capabilities and efficiency. Open source software may be developed in a closed setting, but it may also consist of a patchwork of different contributions originating successively from a number of unsupervised and unrelated developers, who are often scattered across different locations in the world. The modifications brought to the initial software can then either be distributed as a separate programme or be integrated into the original software.

Within a few years, the 'open' method of development and distribution of computer programs has imposed itself as a powerful social ideology. The philosophy behind open source licensing has also inspired the development of numerous other 'open' licences and 'open' projects, where the principles of open source are applied in the fields of music, media, encyclopaedia and science. The mechanism for achieving this goal is through a standardized licensing infrastructure. The open source movement is so powerful in fact that even the software giant Microsoft felt the pressure to offer open and royalty-free documentation and licences for the Microsoft Office 2003 XML Reference Schemas, which provide developers and representatives of business and government a standard way to store and exchange data stored in documents.<sup>4</sup> Microsoft's release of the Office 2003 XML Reference Schemas does not qualify as 'free' or 'open source' software, for the accompanying licence does not grant the user the required freedom to use, reproduce, modify and redistribute the software. Nevertheless, Microsoft's gesture does give an indication of the increasing pressure of disclosing software standards within the community of software developers. Other important 'proprietary' software companies are slowly following Microsoft's footsteps and disclosing certain components of their products to the open source community.<sup>5</sup>

The use of open source software licences has given rise to new, viable, and attractive business models for the distribution of software products. In view of its commercial potential, established companies are investing important capital and labour resources in the development of open source operating systems and applications. Open-source licences cover thousands of projects, including the heart of the Linux operating system, the Firefox Web browser, the Apache server software collection and soon, Sun Microsystems' Solaris version of Unix. Open source software owes its attractiveness to the very principles put forward by its

---

<sup>1</sup> In this text, the expression "open source" software licences encompasses "free" software. For a distinction between the two movements, see: Richard Stallman, 'The Free Software Definition', at <http://www.gnu.org/philosophy/freesw.html>.

<sup>2</sup> Free Software Foundation Europe, <http://fsfeurope.org/documents/freesoftware.html>.

<sup>3</sup> Open Source Initiative, <http://www.ossl.nl/opensource.org/>

<sup>4</sup> 'Microsoft geeft ontwikkelaars meer inzicht', 8 February 2005, WebWereld, online: <http://www.webwereld.nl/nieuws/20737.phtml> (visited October 13th 2005).

<sup>5</sup> S. Shankland, 'Adobe releases open-source interface software', 2 March 2005, CNET News.com.

proponents: software users and developers savour the political freedom granted under the licence to use and modify the software as they wish.<sup>6</sup> The principles underlying the open content movement have been embraced by a large and varied public worldwide, including in the Netherlands, ranging from governments, to businesses, individual users and institutions. To some extent, however, the open source ideology may be victim of its own success, for the number of different open source licences has dramatically increased over the past couple of years, giving rise to compatibility and transparency problems.

The increasing popularity of open source software licences could be the indication of a gradual paradigm-shift in the (use of the) law: from a strict protection regime for computer software following the Microsoft model, the trend moves towards an open and permissive form of protection, following the Linux model. While the protection computer software under copyright and patent law is premised on the idea that the grant of exclusive rights actually encourages creativity and innovation, the open source movement has emerged precisely in reaction to the perceived failure of traditional intellectual property law to do just that. In the opinion of the adherents to the open source software movement, protecting computer software under copyright and patent law leads to a contrary result primarily because of the constant expansion of intellectual property law and of the way rights holders license their rights to users and subsequent developers. Open source software licences offer an alternative to the traditional licensing model with the view of enabling access to computer information and of encouraging further software development. The subject of the impact of the open source software movement on law and practice appears particularly well suited for the *Lex Electronica*'s tenth anniversary special issue in which authors are invited to explore various trends with regard to the impact of new technologies on the Law. This paper therefore presents the main elements of the most commonly used open source software licences, focusing on the General Public Licence (GPL), the Berkeley Software Distribution (BSD) and the Mozilla Public Licence (MPL), examined from a European and Dutch law perspective.

This paper is divided into three parts. Section 1 focuses on issues of copyright law. The open source software ideology, far from rejecting the rules of copyright law, relies on the application of these rules to set their own 'open' terms of use of protected software. The key terms in open source software licences have been designed to take account of the fact that the traditional distinction between creators and users of works has essentially vanished thanks to the digital networked environment: users are creators and vice-versa. To accommodate the incremental development of creative works, the licences grant users the freedom to use, reproduce, modify the software, and the freedom to distribute or re-distribute the work. How do these freedoms fit in with rules on copyright? Section 2 examines the implications of the recognition of the patentability of software-implemented inventions for the development of open source software. To this end, we consider the patent protection as it is currently granted in Europe with respect to computer-implemented inventions. We also take a look at the reaction of some open source software developers in order to counter potential patent infringement claims from third parties. This includes the development of a patent strategy and the drafting of specific language such as the one appearing inside the GPL, and the Mozilla Public Licence. Section 3 takes a brief look at the enforcement of these licences. Finally, section 4 sets out a number of concluding remarks on the emergence of a new paradigm for the licensing of creative works.

---

<sup>6</sup> Pearson, H.E., 'Open Source — The Death of Proprietary Systems?', 3 *Computer Law & Security Report* (2000), pp. 151-156, p. 152.

# 1. Open Source and Copyright Law

Software developers across the European Union enjoy copyright protection on their programs since the adoption, in May 1991, of the Directive 91/250/EEC on the legal protection of computer programs.<sup>7</sup> As a result, all computer programs, whether in object code or in source code,<sup>8</sup> are subject to copyright protection in Europe, provided that they meet the habitual criterion of originality. Open source software does not differ in this respect from any other proprietary software. Open source software does, however, depart from proprietary software in the manner in which it is created and distributed to the public. The modes of creation and distribution of open source software have emerged in reaction to those of proprietary software, where the use of copyright law was seen as an impediment to the further development of software.<sup>9</sup> Far from rejecting the rules of copyright law, the open source movement relies on the application of these rules to set their own ‘open’ terms of use for protected software. The key terms in open source licences have been designed to take account of the fact that the traditional distinction between creators and users of software has essentially vanished within the open source community: users are creators and vice versa. In practice, the most widely used open source licences have been developed from an American law perspective, which shows important differences with European copyright law. In order for European users to be able to fully take part in the open source movement, it is paramount that each software developer knows precisely what his rights and obligations are under the law and the licence.

Since the adoption of the Computer programs directive, the right of reproduction is considered to include the permanent or temporary reproduction of a computer program by any means and in any form, in part or in whole. Given this very broad exclusive right of reproduction, some limitations had to be introduced to allow the lawful user to execute certain acts with respect to protected software without the former’s prior authorisation. Nevertheless, most provisions in the national legislation concerning the use of computer programs are merely default rules from which contracting partners may derogate. In practice, copyright owners usually grant users permission to accomplish certain acts with respect to their copyright protected work by means of a licence. The main difference between open source licences and proprietary software licences lies first and foremost in the freedoms that the former type of licence grants to users and, in certain cases, in the corollary obligation to make the source code available to fellow developers. As we will discuss below, the user of open source software enjoys an extended freedom to use, reproduce, modify and (re)distribute the software. In return, the licensee who undertakes to modify and redistribute new software based on an open source program must agree, at least under the GPL, to renounce receiving royalty payments for the use of the software. The following pages concentrate exclusively on the distinctions shown between the terms of the GPL, the BSD, and the Mozilla licences and the rules on copyright currently in force at the European level, as transposed into Dutch law.<sup>10</sup>

---

<sup>7</sup> Council Directive 91/250/EEC of 14 May 1991 on the legal protection of computer programs, *OJ L* 122 17.05.1991 p. 42 [hereafter the Computer programs directive].

<sup>8</sup> Computer programs directive, recital 7, which reads as follows: “Whereas, for the purpose of this Directive, the term ‘computer program’ shall include programs in any form, including those which are incorporated into hardware; whereas this term also includes preparatory design work leading to the development of a computer program provided that the nature of the preparatory work is such that a computer program can result from it at a later stage”.

<sup>9</sup> Azzaria, G., ‘Les logiciels libres à l’assaut du droit d’auteur’, 16 *Les cahiers de propriété intellectuelle* (2004), pp. 405-428, p. 409.

<sup>10</sup> For a more exhaustive account of the legal protection of proprietary software, we refer the reader to Meijboom, A. in Prins, J.E.J. et al., *Recht en Informatietechnologie – Handboek voor Rechtspraak en Beleid*, (Den Haag, SdU Uitgevers 1999), pp. 71/1- 71/36, pp. 71/1- 71/36; and Verkade, D.W.F., ‘Intellectuele Eigendom’, in H. Franken, H.W.K. Kaspersen and A.H. De Wild, *Recht en computer*, 5<sup>th</sup> ed., (Deventer, Kluwer 2004), pp. 227-289, pp. 242-258.

At the outset, it is important to note that, contrary to what is often asserted in the literature, the freedom of use afforded under most open source licences does not, as such, entail a waiver of right on the part of the “Rights Owner”. In our opinion, the grant of a permission to execute certain acts with respect to a copyright protected work falls within the scope of the “Rights Owner’s” exclusive right to authorise or prohibit the reproduction and communication to the public of his work and must be distinguished from a waiver of right. Admittedly, the line between a very broad licence to use a work and a waiver of right is in practice hard to draw. This may explain the confusion in terminology found in the literature. A waiver of right implies that the “Rights Owner” forsakes his right to exercise in the future one or all of his exclusive prerogatives, with respect to one or more individuals.<sup>11</sup> If the licensor waived his right of reproduction and communication to the public, there would logically be no consequence attached to the non-respect of the conditions by the licensee. However, by granting users the freedom to use, reproduce, modify and (re)distribute the open source software, the licensor does not relinquish his right to institute infringement proceedings should the licensee fail to honour the conditions set out in the licence. The failure by the licensee to meet the requirements of the GPL, or the Mozilla Public licence regarding the distribution of new versions of the software may result in the termination of the licence. This, in our opinion, supports the qualification of the open source licence as a broad permission, rather than a waiver of right.<sup>12</sup>

### 1.1. Freedom to use

Traditionally, copyright owners have never held absolute control over the use of their works. Everyone is in principle free to read, listen to, or view a work for his or her own learning or enjoyment. In theory, copyright never protected against acts of consumption or reception of information by individuals.<sup>13</sup> With the adoption of the Computer programs directive, this is no longer true, however, with respect to computer software.<sup>14</sup> In the case of software, the execution of even the most trivial operation constitutes a restricted act, since it involves making at least one temporary reproduction of the software in the RAM memory of the computer. Article 45i of the Dutch Copyright Act 1912 indeed specifies that “without prejudice to the provisions of article 13, the reproduction of a work as referred to in article 10, paragraph 1, sub 12°, shall include the loading, displaying, running, transmission and storage, in so far as these acts are necessary for the reproduction of the said work.” According to recital 17 of the Computer programs directive however, the exclusive rights of the author to prevent the unauthorized reproduction of his work must be subject to a limited exception in the case of a computer program to allow the reproduction technically necessary for the use of that program by the lawful acquirer. According to article 5(1) of the Computer programs directive, in the absence of specific contractual provisions, the acts of reproduction referred to in article 4a) and b) do not require authorization by the right holder where they are necessary for the use of the computer program by the lawful acquirer in accordance with its intended purpose, including for error correction. This provision has been incorporated into article 45j of the Copyright Act 1912, which reads as follows:

---

<sup>11</sup> Spoor, J.H., D.W.F. Verkade and D.J.G. Visser, *Auteursrecht*, 3rd ed., (Deventer, Kluwer 2004), p. 552.

<sup>12</sup> See: Clément Fontaine, M. *La licence publique générale gnu [logiciel libre]*, (Montpellier, Mémoire de D.E.A Droit des Créations Immatérielles, Université de Montpellier I, Faculté de Droit, 1999), online: <http://www.freescape.eu.org/biblio/IMG/pdf/gpl.pdf>, § 59.

<sup>13</sup> Guibault, L.M.C.R., *Copyright limitations and contracts. An analysis of the contractual overridability of limitations on copyright* (The Hague/ London/Boston: Kluwer Law International 2002), p. 48.

<sup>14</sup> Jaeger, T. & A. Metzger, *Open source software : Rechtliche Rahmenbedingungen der Freien Software*, (München, Verlag C.H. Beck, 2002), p. 21.

“Unless otherwise agreed, the reproduction of a work as referred to in article 10, paragraph 1, sub 12° by the lawful acquirer of a copy of said work, where this is necessary for the use of the work for its intended purpose, shall not be deemed an infringement of copyright. Reproduction, as referred to in the first sentence, in connection with loading, displaying or correcting errors cannot be prohibited by contract.”

From the text of the directive and its implementing provision in the Dutch Copyright Act, it follows that the minimum rights of use are conferred only to the ‘lawful acquirer’ of a computer program. When is a person to be considered the ‘lawful acquirer’ of a computer program? More importantly for our purpose, can the person who downloads or otherwise obtains free of charge a copy of an open source program be seen as a ‘lawful acquirer’? How must one interpret the ‘lawful’ character of the acquisition? Should the ‘lawfulness’ be assessed in relation to the authorisation to use the software granted under licence by the copyright holder, or in relation to the acquisition of the copy of the software, where the lawfulness is considered from a property law perspective.<sup>15</sup> In the first case, a user who acquires in good faith an infringing copy of the software would not be considered a ‘lawful’ acquirer of the program in the sense of the Copyright Act, while it could be true in the second case.

Van Schelven and Struik argue that, in view of the copyright dimension of the Computer programs directive, the ‘lawfulness’ of the acquisition should logically be evaluated from the perspective of the authorisation of the copyright holder rather than from a property law perspective. As a logical consequence of this, it is also generally accepted that a subsequent acquirer of the same copy of the software would be a ‘lawful acquirer’, even in the absence of a licence, according to the doctrine of exhaustion of rights.<sup>16</sup> In a preliminary ruling, the district court of Zutphen (The Netherlands) confirmed this interpretation of the expression ‘lawful acquirer’.<sup>17</sup> The court ruled in this case that the simple fact that a copy of the program had been obtained legally, i.e., without having been stolen, it did not imply that the acquirer had the right to pose the acts of a ‘lawful acquirer’ in the sense of article 45j of the Copyright Act. On appeal, the Court of Arnhem overruled the decision, arguing among other things that since the software had been acquired in good faith the acquirer could be regarded as lawful within the meaning of article 45j of the Act.<sup>18</sup> This part of the court’s ruling received severe criticism in the literature: first, the good faith character of the acquisition was irrelevant from a copyright law point of view, since infringement done in good faith is still an act of infringement. Moreover, the appeals decision went against the majority opinion which considers the ‘lawful acquirer’ solely to be the one who is authorised to use the software in accordance with a purchase or licence contract from the copyright owner or his assignee.

The European Commission would seem to agree with the majority of opinion in the Netherlands, which considers the ‘lawful acquirer’ to be the one who is authorised to use the software in accordance with a purchase or licence contract from the copyright owner or his assignee and not to be the one who legally obtained a copy of the program. In its report on the implementation and effects of Directive 91/250/EEC on the legal protection of computer programs, the European Commission observes that divergences of views subsist however as

---

<sup>15</sup> Van Schelven, P.C. and H. Struik, *Softwarerecht : bescherming en gebruik van programmatuur sedert de Richtlijn Softwarebescherming*, (Deventer, Kluwer 1995), p. 79-82.

<sup>16</sup> Van Schelven and Struik 1995, *supra* note 15, p. 81; HR 25 January 1952, *NJ* 1952No. 95 (*Leesportefeuille*); and HR 20 November 1987, *NJ* 1988No. 82, with annotation from Wichers Hoeth (*Stemra/Free Record Shop*).

<sup>17</sup> Arrondissementsrechtbank Zutphen, 29 April 1999 (*Deurwaarders Software Services*), *Computerrecht* 1999/4, § 9.19.

<sup>18</sup> Gerechtshof Arnhem 11 december 2001 (*Blomsma/ Deurwaarders Software Services (DWSS) BV*), *Computerrecht* 2002/2, with annotation from E. Thole.

to the meaning of 'lawful acquirer'. Several Member States have transposed this notion by using the term 'lawful user' i.e., a person having a right to use the program. The Commission shares the view of some commentators that 'lawful acquirer' does in fact mean a purchaser, licensee, renter or a person authorised to use the program on behalf of one of the above. This argument also draws from Articles 6 and 8 of the Database Directive (Directive 96/9/EC)<sup>19</sup>, which use the term 'lawful user', and which were modelled along the lines of Article 5 (1) of the Computer Programs Directive. In the view of the Commission, what was intended by Article 5 (1) and recital 18 was that it should not be possible to prevent by contract a "lawful acquirer" of a program doing any of the restricted acts that were required for the use of the program in accordance with its intended purpose or for correcting errors. It is, however, possible for a contract to include specific provisions that "control" the restricted acts, which may be carried out by the user of the computer program.<sup>20</sup>

With this definition of a 'lawful acquirer' in mind, one could reasonably argue that anyone having a licence to use an open source computer program is a 'lawful acquirer' of that program within the meaning of the Copyright Act, provided that the licence accompanying the product was validly entered into. Whether the user of the open source software obtained the copy free of charge or not should make no difference, since the majority opinion considers that the lawfulness of the acquisition should not be assessed from a property law perspective. In other words, whether the transaction would qualify as a donation rather than a purchase is irrelevant for the determination of when a user is a 'lawful acquirer' of open source software pursuant to article 45j of the Act.

According to some commentators, a literal interpretation of article 45j of the Act would suggest that a person who downloads an electronic version of the software instead of acquiring a tangible copy of the same software could not be regarded as a lawful acquirer of 'a copy' of the software. As a result, the acquirer of a computer program downloaded from the Internet would not be entitled to benefit from the minimum rights of use.<sup>21</sup> In our opinion, it cannot have been the intent of the legislator to limit the application of the provision according to the medium upon which the software is distributed to the public. Arguably, at the time the directive was adopted in 1991, neither the European legislator nor the national legislators of the Member States may have been aware of the possibility to distribute software on-line, as an economically viable mode of exploitation.<sup>22</sup> Today, on-line distribution has become one of the most important modes of exploitation of both proprietary and open source software, including for popular programs like Microsoft Windows and Linux. Such a restrictive and technology dependent interpretation would, in our opinion, be inconsistent

---

<sup>19</sup> Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the Legal Protection of Databases, O.J. L 077, 27 March 1996, pp. 20-28.

<sup>20</sup> Report from the Commission to the Council, the European Parliament and the Economic and Social Committee on the implementation and effects of Directive 91/250/EEC on the legal protection of computer programs, COM/2000/199 final, Brussels, April 10<sup>th</sup> 2000, online: [http://europa.eu.int/eur-lex/lex/LexUriServ/site/en/com/2000/com2000\\_0199en01.pdf](http://europa.eu.int/eur-lex/lex/LexUriServ/site/en/com/2000/com2000_0199en01.pdf).

<sup>21</sup> Koelman, K.J., "Brothers in arms: open source en auteursrecht", 5 *Computerrecht* (2004), pp. 230-233, p. 230; Thole, E. and W. Seinen, 'Open Source Software licenties: een civielrechtelijke analyse', 5 *Computerrecht* (2004), pp. 221-225, p. 221; and M.W. Scheltema and E. Tjon Tjin Tai, "Overeenkomsten sluiten door openen en klikken?", 4 *Computerrecht* (2003) pp. 244-248, p. 248.

<sup>22</sup> See: Proposal for a European Parliament and Council Directive on the harmonisation of certain aspects of copyright and related rights in the Information Society, Explanatory Memorandum, December 1997, § II.A.5, where the Commission writes: 'The Computer Programs Directive, in its Article 4, however, only protects "any form of distribution to the public" of computer programs, not expressly addressing its on-line transmission over the networks. Indeed, at the time of adoption of the Directive the usual form of distribution took place on the basis of floppy discs and not on-line'. And Meijboom, 'The EC Directive on Software Copyright Protection', in Jongen, H.D.J. and A.P. Meijboom (eds.), *Copyright Software Protection in the EC*, (Deventer, Kluwer Law and Taxation Publishers 1993), p. 11 where the author writes: "Usually, software is transferred, in whole or part, from the medium on which it is supplied (diskette, tape, hard disk) to the computer's memory in order to execute the program".

with the more common interpretation of the provision according to which ‘lawful acquirer’ means the person who is authorised to use the program. From a practical point of view, this interpretation would also frustrate the reasonable expectations of use of all licensees who acquire software on-line, consequence, which can hardly be justified under the law.

Article 45j of the Dutch Copyright Act also implies that while “Rights Owner’s” may contractually regulate the running, transmitting or storing of a computer program, they may not prohibit lawful acquirers from performing such acts as the loading, displaying or correcting of errors. The last sentence makes it clear that, in view of the unprecedented expansion of the copyright protection, the Dutch legislator wanted to guarantee a minimum right of the lawful acquirer of a copy of a computer program to perform those acts that are necessary for the normal use of the computer program.<sup>23</sup> Apart from the limited acts of loading, displaying, or correcting errors, a lawful acquirer may, however, only execute those acts that are necessary for the use of the work for its intended purpose. When can an act be deemed necessary for the use of the program for its intended purpose? Verkade notes on this subject that the circular formulation of articles 45i and 45j is the result of a political compromise and that it certainly cannot have been the intention of the legislator to include any and all technically possible acts of reproduction within the scope of protection of the “Rights Owner”.<sup>24</sup> For Meijboom, the intended purpose of the software is a question of fact that can be assessed in relation to the software’s nature, functionality, or capacity. The decisive factor in establishing what the intended use of a particular program is consists in looking at the common intention of the parties at the time they concluded the licensing agreement.<sup>25</sup> When the interpretation of the licence contract offers no concrete solution, the intended purpose of the software can be estimated in function of the use that the average purchaser could reasonably have expected to make of the software.<sup>26</sup>

It is worth pointing out in this context that the Directive on copyright and neighbouring rights in the information society<sup>27</sup> introduced a mandatory exception for temporary acts of reproduction. Article 13a of the Copyright Act, which transposed this last provision into Dutch law, provides that:

“The reproduction of a literary, scientific or artistic work does not include temporary acts of reproduction which are transient or incidental and an integral and essential part of a technological process and whose sole purpose is to enable: (a) a transmission in a network between third parties by an intermediary, or (b) a lawful use, of a work or other subject-matter to be made, and which have no independent economic significance.”

According to recital 50 of the InfoSoc directive, however, ‘such a harmonised legal protection does not affect the specific provisions on protection provided for by Directive 91/250/EEC. (...) Articles 5 and 6 of that Directive exclusively determine exceptions to the exclusive rights applicable to computer programs.’ In other words, the temporary reproduction of any other type of copyright protected work than computer software is excluded from the scope of the “Rights Owner’s” exclusive right, provided that the conditions of application of article 13a of the Act are met. As some commentators have argued, the Dutch legislator would have been

---

<sup>23</sup> Guibault 2002, *supra* note 13, p. 220.

<sup>24</sup> Verkade 2004, *supra* note 10, p. 253.

<sup>25</sup> Meijboom 1999, *supra* note 10, p. 71 – 25.

<sup>26</sup> Verkade 2004, *supra* note 10, p. 254.

<sup>27</sup> Directive 2001/29/EC of the European Parliament and of the Council of 22 May 2001 on the harmonisation of certain aspects of copyright and related rights in the information society, L 167, 22/06/2001, p 10-19 [hereafter InfoSoc directive].

wise to review article 45i in the light of new article 13a of the Act in order to avoid any possible ambiguity.<sup>28</sup>

Under Dutch copyright law, pure consumptive uses of computer programs such as loading, displaying and correcting errors on the software cannot be prohibited by contract, even if they technically fall under the scope of the owner's exclusive right. However, the licensor is allowed to contractually regulate the running, transmitting or storing of a computer program. In this sense, open source licences grant users a much greater freedom of use than article 45j of the Act. This is particularly evident from article 0 of the GPL, which specifies that "Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does." The BSD licence states that redistribution and use in source and binary forms, with or without modification, are permitted provided that certain conditions regarding the redistribution of software are met. Similarly, article 2.1 of the Mozilla Public Licence grants the user the following rights: "a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims: (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work".

Contrary to what article 5 of the GPL states, we believe that consumers should obtain a valid licence for the use of the software. For, without a licence, consumers are in principle restricted to the acts mentioned in the Copyright Act. For example, article 45j of the Act permits a lawful acquirer to perform only those acts that are necessary for the use of the work for its intended purpose, apart from the limited acts of loading, displaying, or correcting errors. Moreover, even the making of private copies of software, let alone their distribution among friends and family, is strictly prohibited under the law. The freedom of use and reproduction granted under a typical open source licence is generally much broader than what is allowed under copyright law, making the need to obtain a valid licence if not necessary at least recommendable.

## **1.2. Freedom to reproduce**

As soon as a user wants to do more with his software, than merely loading and displaying it on his computer, he must as a matter of course make a reproduction of the program. This is true not only for running, transmitting, or storing a computer program, but also for ensuring its maintenance<sup>29</sup> and for translating or adapting it. Unless these acts are covered by a limitation on copyright, the user is obligated to obtain permission from the rights holder prior to making any kind of reproduction. The Dutch Copyright Act, on the model of the Computer programs directive, grants the lawful user<sup>30</sup> only limited rights to make unauthorised reproductions of protected computer programs. Article 45k of the Act allows the lawful user of a program to make a copy of that program to serve as a back-up copy, where this is necessary for the use of the work for its intended purpose. The making of private copies of a

---

<sup>28</sup> Kleve, P., *Juridische iconen in het informatietijdperk*, (Deventer, Kluwer 2004), p. 210.

<sup>29</sup> Rb's-Gravenhage 23 April 2003, *Computerrecht*, 2004/5 (*Faco Informatisering BV/ Haley Software BV*), § 8.

<sup>30</sup> According to Meijboom [1999, *supra* note 10, p. 71 – 28], the expression 'lawful user' should be understood in the same terms as the expression 'lawful acquirer' since a person may only be a lawful user if he is a lawful acquirer in the sense of articles 45i and 45j of the Act.

program is strictly prohibited under article 45n of the Copyright Act.<sup>31</sup> Article 45l states that a person who is entitled to perform the acts referred to in article 45i shall also be entitled, while performing them, to observe, study or test the functioning of the work concerned in order to determine the ideas and principles underlying it. Article 45m permits the making of a copy of a program and the translation of the form of its code, provided that these acts are indispensable for obtaining information necessary to achieve the interoperability of an independently created computer program with other programs, and provided that a number of conditions are met. In the Explanatory Memorandum to the Implementation Act, the Dutch government did indicate that the limitations on the exclusive right, such as those set out in Articles 45k, 45l, and 45m of the Act, were imperative. However, according to the government, there was no need to specify this in the Act.<sup>32</sup> Although it would certainly have been clearer to spell it out in the Act, the Dutch courts cannot ignore the mandatory character of these provisions, since they too must interpret these provisions in compliance with the directive.<sup>33</sup> A great deal has been written concerning the scope of these limitations with respect to proprietary software.<sup>34</sup> Suffice to say, here, that the general limitations on the owner's exclusive rights, such as the right to quote and the right to use work for educational purposes are also applicable with respect to the reproduction of a computer program.<sup>35</sup>

In light of these provisions, it is clear that once again the GPL, the BSD, and the Mozilla Public licence all offer the licensee much greater freedom to reproduce the computer program, without restriction as to the number of copies realised or to the purpose for making these copies. From the perspective of the licensor(s), these licences are valid, as "Rights Owner's" are entitled to licence their rights as they see fit. Nothing in the Dutch Copyright Act prevents "Rights Owner's" to license their rights broadly to a third party, whether on an exclusive or non-exclusive basis, for a fee or for free. The only restrictions on the freedom of contract of the "Rights Owner" would be set by the imperative character of the limitations relating to the making of a back-up copy, of a reproduction for purposes of observing, studying and testing the software, as well as to the decompilation of the program for purposes of interoperability. These restrictions are irrelevant in the context of open source software, since all types of open source licences grant the user much broader rights of use than the law normally does.

### **1.3. Freedom to modify**

Generally speaking, the right to modify, adapt, or transform a protected work falls under the exclusive right of reproduction of the owner. This principle is derived from article 13 of the Dutch Copyright Act, which provides that "the reproduction of a literary, scientific or artistic work includes the translation, arrangement of music, cinematographic adaptation or dramatization and generally any partial or total adaptation or imitation in a modified form, which cannot be regarded as a new, original work". The rights holder's exclusive right of reproduction entails more than just the right to authorise or prohibit the making of exact or substantially similar copies. It also extends to the making of arrangements, adaptations, and modifications to an existing work, otherwise called 'derivative works'. Any arrangement, adaptation, or modification of an existing work is subject to the prior authorisation of the

---

<sup>31</sup> Report from the Commission to the Council, the European Parliament and the Economic and Social Committee on the implementation and effects of Directive 91/250/EEC on the legal protection of computer programs, COM/2000/0199 final.

<sup>32</sup> D.W.F. Verkade, "Computerprogramma's in de Auteurswet 1912: het vierde regime...", 3 *Computerrecht* (2003), pp. 86-97, p. 95.

<sup>33</sup> Guibault 2002, *supra* note 13, p. 218.

<sup>34</sup> Verkade 2004, *supra* note 10, p. 252 et seq.; Meijboom 1999, *supra* note 10, p. 71 – 18 et seq.

<sup>35</sup> Groenenboom, M.M., 'Software licenties: van closed source tot open source', 1 *Computerrecht* (2002), pp. 21-29, p. 22.

rights holder.<sup>36</sup> With respect to computer programs, recital 20 of the Computer programs directive states: “the unauthorized reproduction, translation, adaptation or transformation of the form of the code in which a copy of a computer program has been made available constitutes an infringement of the exclusive rights of the author”. Besides the generally applicable limitations on copyright, such as the right to quote and to make reproductions for purposes of research and private study, no specific limitation tempers article 45i of the Copyright Act, with respect to the translation, adaptation, or transformation of software. In other words, a computer program may not be translated, adapted, or transformed without the rights holder’s prior authorisation. Moreover, proprietary licensing contracts are usually adamant in requiring that the licensee refrain from bringing any modification to the software without prior authorisation from the rights holder.

Perhaps more than the freedom to use or to reproduce a computer program, the freedom to modify the software constitutes the cornerstone of the open source movement. As one can read on the home page of the Open Source Initiative (OSI):

“The basic idea behind open source is very simple: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing. We in the open source community have learned that this rapid evolutionary process produces better software than the traditional closed model, in which only a very few programmers can see the source and everybody else must blindly use an opaque block of bits.”<sup>37</sup>

As a rule, users of open source software have, pursuant to the GPL, the BSD, or the MPL, the right to modify the software and to prepare derivative works based upon the original work. Indeed a particular computer program may be qualified as ‘open source software’ only if the licence allows modifications and derivative works, and allows them to be distributed under the same terms as the license of the original software. To facilitate the modification and the evolution of computer programs, most open source licences require that the source code be distributed along with the object code of the program, or that it at least be made available to the public. Only by having access to the source code of an existing computer program, are software developers in a position to build upon that existing program in order to improve it or to develop compatible software.<sup>38</sup> Proprietary software manufacturers are usually very protective of their source code, for it may embody competitive trade secrets, and only gives users access to it in rare circumstances and then, only under controlled conditions.<sup>39</sup> In the same vain, proprietary software suppliers are generally highly reluctant to provide interested parties with interface information. Without the proper interface information or the possibility to decompile a program, computer programmers are absolutely unable to develop any kind of software that is interoperable with existing software. At the time of the adoption of the Computer programs directive, the question of whether the decompilation of a program should be allowed led to heated debates. As a result, the directive contains a mandatory limitation allowing under strict conditions lawful users to decompile a program for purposes of interoperability, which the Dutch legislator has transposed in article 45m of the Copyright Act.<sup>40</sup>

---

<sup>36</sup> Van Lingen, N., *Auteursrecht in hoofdlijnen*, 5<sup>th</sup> ed., (Groningen, Martinus Nijhoff 2002), p. 77.

<sup>37</sup> See <<http://www.opensource.org/>>, site visited on 23 November 2004.

<sup>38</sup> Guadamuz Gonzalez, A., ‘Viral Contracts Or Unenforceable Documents? Contractual Validity Of Copyleft Licences’, *E.I.P.R.* (2004), pp. 331-339, p. 332.

<sup>39</sup> Hoeren, T., ‘Die Pflicht zur Überlassung des Quellcodes’, 10 *Computerrecht* (2004), pp. 721-724, p. 721.

<sup>40</sup> Van Lingen 2002, *supra* note 36, p. 62; De Cock Buning, M., ‘Auteursrecht en reverse engineering’, 5 *IER* (1993), p. 129-137, p. 129; and Meijboom in Jongen and Meijboom 1993, *supra* note 22, p. 14.

According to article 3 of the GPL, for an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable code. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable code runs, unless that component itself accompanies the executable code. If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source code along with the object code.

Decompilation of a program becomes no more than a useless intellectual challenge in the context of open source software. The first principle laid down in the Open Source Definition (OSD) holds that ‘the program must include source code, and must allow distribution in source code as well as in a compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost—preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a pre-processor or translator are not allowed.’ The OSD’s second principle concerns derivative works, whereby ‘the license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.’ The rationale behind this principle is that the mere ability to read the source code is not enough to support independent peer review and rapid evolutionary selection. For rapid evolution to happen, people need to be able to experiment with and redistribute modifications. What constitutes a modification of an open source computer program must be evaluated, in the same way as for any other type of work, according to the criterion of originality.

While neither the GPL nor the BSD licence gives any definition of what must be understood by ‘modification’, the Mozilla license defines ‘modifications’ as follows: ‘any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is: A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications. B. Any new file that contains any part of the Original Code or previous Modifications.’ Arguably, even if they do not expressly define it, the GPL and the BSD licences should make reference to a similar notion of ‘modification’, presumably derived from the American case law on the notion of ‘derivative works’.<sup>41</sup> The BSD and the Mozilla licences grant the user comparable freedom to make modifications to existing open source software. However, article 10 of the GPL warns the user that if he wishes to incorporate parts of the licensed program into other free programs whose distribution conditions are different, he must write to the author to ask for permission. In the case of software licensed by the Free Software Foundation, exceptions can be made, each case being assessed following the double objective of preserving the free status of all derivative works based on free software and of promoting the sharing and reuse of software generally.

---

<sup>41</sup> See Title 17 U.S.C. § 101, definition of ‘derivative work’: A "derivative work" is a work based upon one or more pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications, which, as a whole, represent an original work of authorship, is a "derivative work".

As we shall see in the next section, the freedom to modify open source software under all types of licences is further accompanied by strict obligations as soon as the user wishes to distribute software based on software originally distributed under an open source licence.

#### **1.4. Freedom to (re)distribute**

The freedom to redistribute copies of the software or to distribute a modified version of the software is, just as the freedom to make modifications, one of the key features of any open source licence. A particular computer program will fall under the Open Source Definition only if it complies with the first principle laid down by the Open Source Initiative (OSI), according to which the ‘license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources.’ As a rule, open source licences therefore afford users much greater freedom than article 12 of the Copyright Act normally would with regard to the right to distribute a copy of a copyright protected work, since the exclusive right of the rights holder includes any form of distribution to the public, including the rental, of the original computer program or copies thereof.<sup>42</sup> The only exception to this rule is the one provided for in article 15b of the Act, which concerns the further communication to the public or the reproduction of a literary, scientific or artistic work communicated to the public by or on behalf of the public authorities. According to this provision, unless the copyright has been explicitly reserved, either in a general manner by law, decree, or ordinance, or in a specific case by a notice on the work itself or at the point of communication to the public, a work that is communicated by or on behalf of the public authorities may be freely distributed. This exception would be applicable for example in the case of software distributed by the government.

Under the three types of open source licences examined here, the exercise of this freedom is accompanied by a number of strict conditions of application. The BSD licence states for instance that redistributions of source code must retain the copyright notice, the list of conditions and a disclaimer. Redistributions in binary form must reproduce the copyright notice, the list of conditions and a disclaimer in the documentation and/or other materials provided with the distribution.

As we have seen in the previous subsection, one of the main conditions under the GPL is that the source code be distributed along with the program or that it be made available to any third party who requests it. Article 2 of the GPL regulates essentially the same elements as the BSD licence, i.e., the placement of a copyright notice, of a list of conditions and a disclaimer of warranty and liability, but in much greater detail. As the GPL explains, “it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.” The GPL licence goes further than the BSD licence in that it requires the user to cause any work that he or she distributes or publishes, that in whole or in part contains or is derived from the program, to be licensed as a whole at no charge to all third parties under the terms of the GPL License.

According to article 2 of the Mozilla licence, the initial developer grants the user a licence to among other things distribute the original code (or portions thereof) with or without modifications, and/or as part of a larger work. Article 3 of the Mozilla licence sets out “distribution obligations”, which are comparable in length and complexity to those of the GPL. The provision requires among other things that the user distribute any copy of the software or any work derived from the original code only under the terms of this licence and that a copy of the licence be included in the distribution. The user must also make any

---

<sup>42</sup> Van Lingen 2002, *supra* note 36, p. 87.

modification that he creates available in source code. He must also document the changes made to the original software and duplicate the prescribed copyright notice.

In this sense, the GPL and the Mozilla licences follow the OSI's foremost precept, according to which "the rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties". This clause, known as the 'copyleft' clause, is intended to forbid closing up software by indirect means such as requiring a non-disclosure agreement. To this end, the distribution obligations are placed not only on the initial licensee, but also on any subsequent licensee. In practice, the 'copyleft' clause varies in scope from one type of open source license to another and, as the BSD licence demonstrates, not every type of open source licence contains such a clause.<sup>43</sup>

Under the GPL and the Mozilla licences, the copyleft clause is applicable to the distribution of the original code with or without modification. In the case of the distribution of modified code, the question can arise, however, whether the product involved actually does constitute a 'derivative' work based on the original work, or if it rather constitutes an entirely new work, in the sense of article 13 last sentence, of the Dutch Copyright Act 1912.<sup>44</sup> For, if the new software qualifies as a new work under the copyright act, the developer is in principle not bound by the copyleft term of the licence. As McGowan rightfully observes, there are problems with the proposition that a person creates a work derivative of a program just by writing another program that interacts with it.<sup>45</sup> The last part of article 2 of the GPL does specify the following:

"These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License".

In the open source context, the discussion has focused essentially on the question of whether static and dynamic linking to a computer program distributed under the GPL entail, as a consequence, that the linking program must also be distributed under the GPL. The Free Software Foundation (FSF) has adopted the position that both static and dynamic linking to a computer programme may result in a "derivative" work.<sup>46</sup> Whether other combinations of code modules must also be regarded as an arrangement depends, according to the FSF, on the

---

<sup>43</sup> O'Sullivan, M., 'The Pluralistic, Evolutionary, Quasi-legal Role of the GNU General Public Licence in Free/Libre/Open Source Software ("FLOSS")', *EIPR* 2004, pp. 340-348, p. 343.

<sup>44</sup> Verkade 2004, *supra* note 10, p. 251.

<sup>45</sup> See for example: McGowan, D., 'Legal Aspects of Free and Open Source Software', in J. Feller, B. Fitzgerald, S. Hissam and K. Lakhani (ed.), *Perspectives on Free and Open Source Software*, (MIT Press, Boston, MA, 2005), pp. 361-392, online: <http://www.law.umn.edu/FacultyProfiles/articles/McGowanD-OpenSource.rtf>. (p. 23).

<sup>46</sup> Free Software Foundation, <<http://www.gnu.org/licenses/gpl-faq.html#MereAggregation>>. Incidentally, this rather broad interpretation of the term "derivative work" may have as a consequence that emulators like Wine, which use the libraries of Windows to function, would infringe Microsoft's copyright.

communication mechanism and the semantics of the communication between the modules. The fact that modules are combined into the same file suggests that we are in presence of an adaptation or an arrangement. If the modules are developed in order to be executed together in a combined address location, we can speak of an adaptation or an arrangement. However, collective communication mechanisms, which are normally used by two separate programmes, will not easily be seen as an adaptation. On the other hand, whenever the semantics of the communication are sufficiently “proximate”, one can conclude that an adaptation has been realised. In all cases, the question of whether the new code forms a derivative work or an entirely new work is a matter of fact that should be decided on a case-by-case basis.

The copyleft clause may raise problems from the point of view of copyright law. A potential problem concerns the application of the doctrine of exhaustion of rights. According to article 4c) of the Computer programs directive, “the first sale in the European Community of a copy of a program by the right holder or with his consent exhausts the distribution right within the Community of that copy, with the exception of the right to control further rental of the program or a copy thereof”. As a consequence of this rule, the author of a work loses control over the further dissemination of the copy after it has been made public by him or with his consent. Therefore, selling, lending, leasing and hiring of a copy of a computer program by the lawful acquirer cannot be prohibited.<sup>47</sup> The application of this doctrine has been reaffirmed in slightly different words in article 4(2) of the InfoSoc Directive. This provision states that “the distribution right shall not be exhausted within the Community in respect of the original or copies of the work, except where the first sale or other transfer of ownership in the Community of that object is made by the right holder or with his consent”. Does the distribution of computer programs under the terms of an open source licence constitute a ‘sale’ in the sense of the Computer Programs Directive, which would have as a consequence the effect of exhausting the distribution right of the rights holder? A corollary question to this is whether the distribution of software free of charge entails a ‘transfer of ownership’, which would lead to an exhaustion of right under Community law? Another corollary question is whether, for the purposes of the exhaustion doctrine, there is a difference between the off-line or on-line distribution of a ‘copy’ of a computer program.

Software manufacturers often maintain that the distribution right is not exhausted through the grant of a licence of use of the software, because the licensing of rights does not constitute “the first sale in the European Community of a copy of a program by the right holder or with his consent”. This argument could be inferred from the wording of article 4c) of the Computer programs directive, which would seem to limit the application of the exhaustion doctrine to the ‘sale’ of a computer program, whereas any other form of distribution would not give rise to the application of the doctrine.<sup>48</sup> This theory has been, in our opinion, rightfully contested.<sup>49</sup> Along with Neppelenbroek, we believe that the exhaustion doctrine does not so much focus on the concept of ‘sale’, but rather on that of ‘transfer of ownership’. As Grosheide explains with respect to the general principle of exhaustion of rights: “it is not limited to the first sale of the copy but encompasses other forms of distribution such as donation and first rental. Exhaustion assumes assignment of title with regard to the copy (i.e., the content carrier).”<sup>50</sup>

---

<sup>47</sup> Jongen, in Jongen and Meijboom 1993, *supra* note 22, p. 174. See HR 25 January 1952, *NJ* 1952No. 95 (*Leesportefeuille*); and HR 20 November 1987, *NJ* 1988No. 82, with annotation from Wichers Hoeth (*Stemra/Free Record Shop*).

<sup>48</sup> Van Schelven and Struik 1995, *supra* note 15, p. 70-71; and Grosheide, F.W., ‘Mass-market exploitation of digital information by the use of shrink-wrap and click-wrap licenses’, in F.W. Grosheide & K. Boele-Woelki (ed.), *Opstellen over Internationale Transacties en Intellectuele Eigendom*, (Lelystad, Koninklijke Vermande 1998), Molengrafica Series, pp. 263-319, p. 308.

<sup>49</sup> Neppenlenbroek, E.D.C., ‘Software en de uitputtingsregel’, 6 *AMI* (2001), pp. 125-132, p. 126.

<sup>50</sup> Grosheide 1998, *supra* note 48, p. 307.

This interpretation of the exhaustion doctrine would, in our opinion, be more consistent with the interpretation of the doctrine as it is set out in other European Directives in the field of copyright. It follows from this that in any case, the mere labelling of a transaction as a licence is insufficient as such to circumvent the exhaustion doctrine.

The question of whether the grant of a licence can amount to a sale or to another form of distribution giving rise to the application of the exhaustion doctrine is a matter of fact that should be decided on a case-by-case basis. In the Netherlands, opinions are divided on whether the distribution to the public of a computer program on a tangible medium (i.e., floppy disc or CD-ROM) for an unlimited term and an outright fee is more akin to a sale than a licence, understood in the strict meaning of the word. Such a transaction would entail, in our opinion, a transfer of ownership of the physical embodiment of the work, which would lead to the application of the exhaustion doctrine. In this sense, the court of appeal of The Hague once ruled that the view, according to which the further distribution of the software can be blocked through a clause prohibiting further transfers, would unduly restrict the working of the exhaustion doctrine.<sup>51</sup> In Germany, it is generally accepted that the distribution right is exhausted as soon as a computer program is put into circulation following the terms of a licence and against the payment of a one-time fee, a position that was confirmed by the Federal Supreme Court in the OEM-Version case.<sup>52</sup> The situation would be different if the licence to use the software was limited in time and if the licensee was obligated to periodically pay a fee during the entire duration of the licence. In this case, there would be no transfer of ownership of the physical embodiment of the work, and the distribution right of the rights holder would not be exhausted. On the other hand, the licensing of a copy of the software for an indefinite term, but free of charge, would probably qualify as a donation, thereby implying a transfer of ownership of the physical embodiment of the work.<sup>53</sup> As a result, the distribution right of the rights holder would be exhausted as soon as a tangible copy of the work is put into circulation, even if this occurs free of charge. As Spindler observes, the application of the exhaustion doctrine does not depend on whether the copy of the work is distributed for a price or free of charge. The important factor is that, through the granting of a licence, the distributor operates a definitive transfer of ownership of the software in favour of the licensee.<sup>54</sup>

The above remarks concern the distribution of physical copies of computer programs, i.e., on floppy discs, CD-ROM's, and the like. To the question formulated above, of whether, for the purposes of the exhaustion doctrine, a difference must be made between the off-line or on-line distribution of a 'copy' of a computer program, the answer is yes. While the non-application of the exhaustion doctrine to the electronic delivery of computer programs could already be inferred from the wording of article 4c of the Computer programs directive, which refers only to the "first sale in the European Community of a 'copy' of a program", the question would seem to have been resolved at the European level. According to Recital 29 of the InfoSoc directive:

"The question of exhaustion does not arise in the case of services and on-line services in particular. This also applies with regard to a material copy of a work or other subject matter made by a user of such a service with the consent of the right holder. Therefore, the same applies to rental and lending of the original and

---

<sup>51</sup> E.D.C. Neppelenbroek, annotation by Hof 's-Gravenhage 20 November 2003 (*1st Flight Training*), *Computerrecht* 2004/3.

<sup>52</sup> BGH, 6 July 2000, I ZR 244/97 (*OEM-Version*), *Computer und Recht* 2000, p. 654; see also: Jaeger and Metzger 2002, *supra* note 14, p. 22; and Spindler, G., *Rechtsfragen der Open Source Software*, (Göttingen, Verband Software Industrie 2003), available at: [http://www.vsi.de/inhalte/aktuell/studie\\_final\\_safe.pdf](http://www.vsi.de/inhalte/aktuell/studie_final_safe.pdf), p. 48.

<sup>53</sup> See Dutch Civil Code, art. 7:175.

<sup>54</sup> Spindler 2003, *supra* note 52, p. 51, footnote 302.

copies of works or other subject matter, which are services by nature. Unlike CD-ROM or CD-I, where the intellectual property is incorporated in a material medium, namely an item of goods, every on-line service is in fact an act which should be subject to authorisation where the copyright or related right so provides.”

The notion that the electronic distribution of works does not give rise to the exhaustion doctrine because it falls under the scope of the right of making a work available to the public, rather than under the right of distribution, is now part of the *acquis communautaire*.<sup>55</sup> For more certainty, the European Commission clearly stated, in its report on the implementation of the Computer programs directive, that community exhaustion only applies to the sale of copies, i.e., goods, whereas supply through on-line services does not entail exhaustion.<sup>56</sup> Although this distinction may be unfortunate in the eyes of some commentators,<sup>57</sup> we will not dwell on the issue any further. Nevertheless, it could be argued that the exhaustion doctrine could apply to the tangible copy made from a digital version of a computer program downloaded from the Internet. It would indeed not be unreasonable to think that, if the lawful acquirer of an electronic version of computer program burned the software on a CD, he would be able to transfer that specific CD to a third party without infringing the owner’s copyright, provided that the initial copy of the programme is deleted from his computer.

In light of all this, let us now consider how the exhaustion doctrine applies in the case of an open source licence. We will recall that, under the GPL and the Mozilla licences, the ‘distribution obligations’ are applicable to the distribution of the original code with or without modification. One must realise at this point that the doctrine of exhaustion applies only to the distribution right, not to the right of reproduction or to the right to distribute derivative works.<sup>58</sup> The distribution right is therefore subject to exhaustion only in the case where the original software is distributed on a tangible medium, i.e., on floppy discs or CD-ROM’s, where the licence terms can be interpreted as operating a transfer of ownership in the software and where the licensee further distributes exact and unmodified copies of that software. In such circumstances, the ‘distribution’ obligations of the GPL, the BSD, or the Mozilla Licence would not be binding upon the licensee. On the other hand, these obligations would be binding upon the licensee whenever the open source software is delivered on-line, or when the licensee creates and distributes a derivative work based on an open source computer program. In practice, the exhaustion doctrine would come into play only in limited circumstances, since the large majority of open source software is distributed over the Internet and since a software developer has little interest in distributing exact copies of a program that is otherwise freely available elsewhere. A computer programmer will be much more inclined to put improved versions of the software into circulation. In that case, he must comply with the requirements of the licence, namely to distribute the source code along with the object code of the program, or at least to make it available to the public, to put the proper copyright notices and, in the case of the GPL, to distribute the modified software under the same licence terms.

---

<sup>55</sup> Walter, M.M. (ed.), *Europäisches Urheberrecht – Kommentar*, (Wien, New York, Springer Verlag 2001), p. 1053; and Neppelenbroek 2001, *supra* note 49, p. 127.

<sup>56</sup> Report from the Commission to the Council, the European Parliament and the Economic and Social Committee on the implementation and effects of Directive 91/250/EEC on the legal protection of computer programs, COM/2000/0199 final.

<sup>57</sup> Bolcher in Walter 2001, *supra* note 55, p. 171 and ff.; Tjon Tjin Tai, E., ‘Exhaustion and Online Delivery of Digital Works’, *EIPR* (2003), pp. 207-210, p. 207.

<sup>58</sup> Walter 2001, *supra* note 55, p. 1043.

## 1.5. Royalty free distribution

In addition to the threefold requirement mentioned above, open source licences generally demand that the software developer, who wishes to distribute a modified version of the open source software, agree not to require a royalty or other fee for the sale of open source software. By imposing this requirement, the Open Source Initiative hopes that the temptation to throw away many long-term gains in order to make a few short-term sales dollars will disappear.<sup>59</sup> Otherwise, the OSI fears that co-operators would find themselves under a lot of pressure to defect from the open source movement in favour of more lucrative activities. In practice, this condition means that the distribution of any software developed on the basis of an open source programme must not be subject to the payment of a royalty fee.<sup>60</sup> Not all licences contain this requirement. Whenever they do not, like the BSD and the Mozilla licences, they are not regarded as falling under article 1 of the Open Source Definition which states that ‘the license shall not require a royalty or other fee for such sale’. However, the prohibition to charge royalties for the use of the software does not exclude the possibility of charging a fee for other aspects of the distribution. Article 1 of the GPL stipulates, for example, that “you may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.”

The obligation to license open source software free of charge raises little concern from a copyright law perspective. Pursuant to the Dutch Copyright Act, an author may transfer his exploitation rights in whole or in part; or grant them under licence, exclusive or non-exclusive, for a fee or free of charge. Nothing in the Copyright Act precludes rights holders from agreeing by contract not to ask for royalties for their copyright protected work.<sup>61</sup> From Microsoft to small software developers, the free distribution of software is in fact very common.<sup>62</sup> Most of all, the GPL does not prevent “Rights Owner’s” to make money, for example by charging for support and for guarantee. As an illustration of this, the Linux operating system is distributed, free of charge, over the Internet under the GPL terms. However, for any user who feels unable to download the free version of the software from the Internet and install it, potential users have the possibility of purchasing from several vendors a boxed set with the Linux system. In this case, the cost of Linux system increases, but on the other hand, the user will be provided with full documentation, step-by-step installation instructions and in many cases free technical support for up to 90 days by phone or e-mail.<sup>63</sup>

While the obligation to distribute software on a royalty free basis would probably pose no difficulty from a copyright law point of view, such an agreement may, in certain circumstances, raise competition law concerns. A first concern relates to the obligation of open source licensees to distribute the software free of charge. It has been suggested that this practice could, in certain circumstances, be regarded as an unlawful imposition of a vertical restraint in the form of resale price maintenance, contrary to article 6 of the Dutch Competition Act or article 81(1) of the European Union Treaty.<sup>64</sup> Both provisions prohibit all agreements between undertakings, which have as their object or effect the prevention, restriction, or distortion of competition within the common market, including agreements that directly or indirectly fix purchase or selling prices. However, a restrictive agreement may still be allowed, under article 6(3) of the Dutch Competition Act and article 81(3) of the European

---

<sup>59</sup> See <<http://www.opensource.org/docs/definition.php>>, site visited on 2 December 2004.

<sup>60</sup> Gonzalez 2004, *supra* note 38, p. 333.

<sup>61</sup> Spindler 2003, *supra* note 52, p. 48.

<sup>62</sup> Microsoft Corp (COMP/C-3/37.792) (Unreported, March 24, 2004) (CEC).

<sup>63</sup> See: Linux website, Download Information, at <[http://www.linux.org/dist/download\\_info.html](http://www.linux.org/dist/download_info.html)> (site visited on 21 October 2004).

<sup>64</sup> Koch, F.A., ‘Urheber- und kartellrechtliche Aspekte der Nutzung von Open-Source-Software’, 6 *Computer und recht* (2000), pp. 333-344, p. 341.

Treaty, if it contributes to improving the production or distribution of goods or to promoting technical or economic progress, while allowing consumers a fair share of the resulting benefit, and which does not: (a) impose on the undertakings concerned restrictions which are not indispensable to the attainment of these objectives; (b) afford such undertakings the possibility of eliminating competition in respect of a substantial part of the products in question.

In the case of open source software distributed under the GPL, it could be argued that it is precisely this obligation of distributing software on a royalty free basis that ensures that the open source community has the incentive to produce and distribute improved software, which is certainly to the advantage of the consumers. In other words, this argument would probably fail in our opinion, because the restrictive clause would likely pass the test of article 6(3) of the Dutch Competition Act and article 81(3) of the European Treaty. The distribution on a royalty-free basis would probably also be able to benefit from the application of the Block Exemption on technology transfer agreements<sup>65</sup>, which allows setting maximum prices under certain conditions.

The obligation to distribute open source software on a royalty free basis could also be seen as an abuse of dominant position arising from the predatory pricing on the part of the licensor,<sup>66</sup> contrary to article 24 of the Dutch Competition Act or article 82 of the European Union Treaty. Generally speaking, predatory pricing occurs, *inter alia*, where a dominant firm sells a good or service below cost for a sustained period of time, with the intention of deterring entry, or putting a rival out of business, enabling the dominant firm to further increase its market power and later its accumulated profits.<sup>67</sup> To amount to an abuse of dominant position under article 24 of the Dutch Competition Act,<sup>68</sup> two conditions must be met: first, the undertaking must occupy a dominant position in the market or a substantial part thereof; second, the undertaking must abuse its dominant position. Currently, hardly any open source project would meet these two conditions. Although this should be confirmed by exact figures, none of the open source projects has to our knowledge reached the degree of market share necessary to be in the presence of a dominant position, even if certain software is very successful on specific markets. Moreover, the open source software having the biggest market share is not necessarily the one distributed under the GPL, but it is often software distributed under the BSD, the Mozilla Public Licence or any of their variant which does not contain such an obligation to distribute open source software on a royalty-free basis. In addition, the evidence of an abuse of dominant position arising from predatory pricing would be very difficult to establish since this obligation is part of an ideological movement, aimed at improving technological progress and not at eliminating the competition. Finally, the obligation to refrain from asking for royalties does not prevent companies, like Red Hat and SuSe, to charge the small and medium-sized business market substantial sums of money for the support of the Linux server.<sup>69</sup> Far from representing a form of predatory pricing, these high prices may come with consequences, especially in a market where free alternatives are available for those who don't want as much support, software updates and certification as Red Hat offers. Competition is alive and well, also in the open source market!

---

<sup>65</sup> Commission Regulation (EC) No. 772/2004 of 27 April 2004 on the application of Article 81(3) of the Treaty to categories of technology transfer agreements, *Official Journal* L 123, 27.04.2004, pages 11-17, art. 4(1).

<sup>66</sup> Heath, C., 'Open Source Software: Law, Politics and Economics', 1 *Journal of Digital Property Law* (2002), pp. 216-252, p. 251.

<sup>67</sup> Case C-62/86 *AKZO Chemie BV v Commission*.

<sup>68</sup> Article 82 of the Treaty establishing the European Community states that: 'Any abuse by one or more undertakings of a dominant position within the common market or in a substantial part of it shall be prohibited as incompatible with the common market in so far as it may affect trade between Member States'.

<sup>69</sup> Stephen Shankland, 'Dell: Red Hat needs to lower prices', CNET News.com, 7 December 2004, available at <[http://news.com.com/Dell+Red+Hat+needs+to+lower+prices/2100-7344\\_3-5482234.html?tag=cd.top](http://news.com.com/Dell+Red+Hat+needs+to+lower+prices/2100-7344_3-5482234.html?tag=cd.top)>.

## 2. Open Source and Patent Law

The grant of patent protection with respect to computer programs has been a problematic issue worldwide for well over two decades. In the United States, the United States Patents and Trademark Office (USPTO) and the courts were initially very hesitant to grant patents with respect to software, since software was considered equivalent to mathematical algorithms or laws of nature, and thus not patentable.<sup>70</sup> A Supreme Court ruling in 1981 drastically changed software patenting in the United States. It held that, while software in isolation remained unpatentable, software innovations were patentable if they were claimed as part of a process.<sup>71</sup> The 1990's were marked by two important rulings from the Court of Appeals for the Federal Circuit,<sup>72</sup> which effectively extended the patent protection to cover software and business methods. Today, patents are granted regularly in the United States with respect to software, provided that the invention produces a 'concrete, useful and tangible' result and that it is new and non-obvious.<sup>73</sup>

In Europe, patent protection is granted pursuant to article 52(1) of the European Patent Convention (EPC) for any inventions, which are susceptible of industrial application, which are new and which involve an inventive step.<sup>74</sup> Although the EPC does not expressly require it, the constant practice of the European Patent Office (EPO) has also been to grant a patent only if the claimed subject-matter, considered as a whole, has a technical character.<sup>75</sup> While 'programs for computers' are included in the list of items that are not regarded as inventions within the meaning of the Convention, if the claimed subject-matter has a technical character, it is not excluded from patentability. Accordingly, the EPO has issued over the years an estimated 30,000 patents relating to computer-implemented inventions and a considerable body of case law on the subject has been built up by the appellate bodies of the EPO and the Member States' courts. It is important to point out that the EPC is entirely separate from the European Community and the EPO is not subject to Community law. Granted European patents form a 'bundle' of national patents which have to be validated, maintained and litigated separately in each Member State. The patent holder in any case obtains, for a period twenty years from the date of filing of the application, the exclusive right to make, use, put on the market or resell, hire out or deliver the patented invention, or otherwise deal in it commercially, or to offer, import or stock it for any of those purposes. Even more than copyrights, patent rights have the potential to confer on their owner a degree of monopoly power in the market. Patents therefore constitute a significant economic instrument in the competition process.

The appropriateness of granting to software-implemented inventions the same level of protection as other types of inventions is a hotly debated topic, namely in view of the very particular mode of development of software and in view of the fact that software also benefits from copyright protection.<sup>76</sup> The controversy is in fact so strong that the recent efforts of the European legislator towards the adoption of a European directive on the patentability of

---

<sup>70</sup> *Gottschalk v. Benson*, 409 U.S. 63 (1972); and Evans, D.S. & A. Layne-Farrar, "Software Patents And Open Source: The Battle Over Intellectual Property Rights", 9 *Virginia Journal of Law & Technology* (2004), pp. 10-38, § 8.

<sup>71</sup> *Diamond v. Diehr*, 450 U.S. 175, 186 (1981)

<sup>72</sup> *In re Allapat* 33 F.3d 1526 (Fed. Cir. 1994); *State Street Bank and Trust Co. v. Signature Financial Group Inc.* 149 F.3d 1368 (Fed. Cir. 1998).

<sup>73</sup> Bakels, R.B. & P.B. Hugenholtz, *The patentability of computer programs, study commissioned by the European Parliament* (Amsterdam, Institute for Information Law, April 2002), p. 13.

<sup>74</sup> European Patent Convention, art. 52(1).

<sup>75</sup> Guidelines for Examination in the EPO, C-IV, § 2.3.

<sup>76</sup> See: Evans and Layne-Farrar 2004, *supra* note 70, § 8.

computer-implemented inventions have until now remained unsuccessful.<sup>77</sup> Indeed, because the rules regarding the patentability of computer-implemented inventions and the interpretation of patent claims differ among the EU Member States, the European Commission has proposed the text of a directive intended to set clear borders to what would be patentable in the EU and what would not. While the European Commission argues that the harmonisation of the patent rules regarding computer-related inventions is necessary to remedy the current lack of legal certainty in the field, opponents maintain that the proposed directive may not only fail to achieve its intended objective, but may also have undesirable effect on software development.<sup>78</sup>

Open source software developers have consistently taken the position that software patents generally impede innovation in software development and that software patents are inconsistent with open source software ideology. The implications of the current patenting practice for the open source movement became very clear during the summer of 2004, when the news circulated that the Linux kernel could be infringing an estimated 283 patents worldwide, and 50 patents in Europe alone.<sup>79</sup> Soon after the results of the Open Source Risk Management (OSRM) survey were disclosed, the city of Munich announced that it would halt its 13,000-desktop migration to Linux in order to investigate whether software patent laws in the EU could impact the city's use of the open source operating system.<sup>80</sup> In the light of this incident, we will examine in the first subsection the implications of the recognition of the patentability of software-implemented inventions for the development of open source software, without however, putting the entire patent system into question. To this end, we briefly consider the patent protection as it is currently granted in the Netherlands with respect to computer-implemented inventions. We then take a look at the reaction of some open source software developers in order to counter potential patent infringement claims from third parties. This includes the development of a patent strategy and the drafting of specific language such as the one appearing inside the GPL, and the Mozilla Public Licence.

## **2.1. Open source and patented software**

With respect to the Netherlands, an inventor, or his assignee, may apply for a purely national patent to be issued pursuant to the Dutch Patent Act of 1995, or may choose to designate the Netherlands, as one of the territories for which patent protection is sought, to be issued as part of a bundle of national patents pursuant to the European Patent Convention. While the Dutch and the European patent regimes both impose similar substantive requirements, there exists a significant difference in their application and issuance procedures. The Dutch patent regime is generally referred to as a 'registration system', where a patent is granted as soon as the formal requirements are met, irrespective of whether the invention also meets the substantive criteria for patentability, such as novelty, inventivity and industrial application. Contrary to the European patent system, where the patentability of an invention is evaluated *ex ante* by the patent examiner in the course of the application procedure, the validity of a Dutch patent is assessed *ex post* by the judge, in the context of an infringement or an invalidation procedure. The Dutch Patent Act does require the production of a novelty search conducted by the Office for the Industrial Property (BIE) prior to the start of any infringement or invalidation proceeding. At the time of its implementation, the Dutch 'registration system' was believed to

---

<sup>77</sup> Proposal for a Directive of the European Parliament and of the Council on the Patentability of Computer-implemented Inventions, Brussels, 20.02.2002 COM(2002) 92 final 2002/0047 (COD).

<sup>78</sup> Bakels and Hugenholtz 2002, *supra* note 73, p. 43.

<sup>79</sup> Steven J. Vaughan-Nichols, 'Open-Source Insurance Provider Finds Patent Risks in Linux', *eWeek*, August 2, 2004, available at: <<http://www.eweek.com/article2/0,1759,1630082,00.asp>>.

<sup>80</sup> Linux Business Week News Desk, 'Concerned Over Patent Infringement, Munich Calls Halt to Linux Switch', *LinuxWorld*, August 5, 2004, available at <<http://www.linuxworld.com/story/45825.htm>>.

be simpler and more accessible to small and medium enterprises (SME's), than an 'examination patent system' like the European patent system. Whether this system has yielded the expected advantages is a question, which reaches far beyond the scope of this study.<sup>81</sup> The fact remains, however, that open source software developers must, in their developing process, take account of the possible existence of potentially conflicting patents on related computer-implemented inventions, whether issued under the Dutch or EPC patent system, and which their owners will undeniably want to enforce.

Let us briefly consider the workings of the difficulties posed to open source software developers under the current legal framework as well as under the proposed directive. Given the fact that the substantive requirements of the Dutch and European patent systems are fairly comparable, we will refer below primarily to the provisions of the EPC and to the case law of the EPO, since it is more extensive on this subject than the purely national jurisprudence. With respect to the scope of protection granted, we will refer to the Dutch Patent Act since article 64 of the European Patent Convention refers directly to the national legislation on this issue.

An invention can be a process, a machine, a product, or a composition of matter. In order to be patentable under the EPC, an invention must have a technical character. In particular, this requirement is not met if the patent application or the patent relates to mathematical methods, rules and methods for performing mental acts or doing business, presentation of information or computer programs as such. Assuming that a patent application is formulated so as to avoid claiming rights on a program for a computer 'as such', which would fall under the exclusion of article 52(2) of the EPC, the invention must also be susceptible of industrial application, be new, and involve an inventive step.<sup>82</sup> An invention is considered new if it does not form part of the state of the art. The state of the art comprises of everything made available to the public by means of a written or oral description, by use, or in any other way, before the date of filing of the European patent application, including pending patent applications (published or not) as well as any published innovations in industry or academic journals. An invention is considered as involving an inventive step if, having regard to the state of the art, it is not obvious to a person skilled in the art.<sup>83</sup> If the state of the art also includes patent applications that were filed prior to the date referred to in the application but which were published on or after that date, these documents are not to be considered when deciding whether there has been an inventive step. With respect to the evaluation of the technical character of a computer-implemented invention, the Guidelines for Examination in the EPO give patent examiners the following instructions:

'If a claimed invention does not have a prima facie technical character, it should be rejected under Art. 52(2) and (3). In the practice of examining computer-implemented inventions, however, it may be more appropriate for the examiner to proceed directly to the questions of novelty and inventive step, without considering beforehand the question of technical character. In assessing whether there is an inventive step, the examiner must establish an objective technical problem, which has been overcome. The solution of that problem constitutes the invention's technical contribution to the art. The presence of such a technical contribution establishes that the claimed subject-matter has a technical character and therefore is indeed an invention within the meaning of Art. 52(1). If no such objective technical problem is found, the claimed subject-matter does not satisfy

---

<sup>81</sup> See D. van Engelen, 'Het Nederlandse registratieoecrooi: een wolf in schaapskleren!', *IER* 2004/1.

<sup>82</sup> Bakels and Hugenholtz 2002, *supra* note 73, p. 8.

<sup>83</sup> European Patent Convention, art. 56.

at least the requirement for an inventive step because there can be no technical contribution to the art, and the claim is to be rejected on this ground.<sup>84</sup>

European patents have been granted with respect to all kinds of computer-implemented inventions, ranging from an activated anti-blocking-system (ABS), to a road-pricing system, a voice-recognition system, a data-compression (MP3) system, and a biometrical identification and access control system, to name but a few examples.<sup>85</sup> Most of these patents relate to a new process or machine. In practice, the requirement of a ‘technical effect’ has proved to be rather ambiguous and difficult to apply. The interpretation of the substantive criteria of ‘technical effect’, novelty, and inventiveness of computer-implemented inventions has led to a considerable body of case law from the appellate bodies of the EPO.<sup>86</sup> Over the years, the EPO has generally taken the position that the technical character of a computer-implemented invention cannot be acknowledged for the sole reason that a program causes physical modifications of the hardware (i.e., electrical currents) deriving from the execution of the program instructions. A technical character might however be found in further effects deriving from the execution by the hardware of the instructions given by the computer program. Where these further effects have a technical character or where they cause the software to solve a technical problem, an invention that brings about such an effect might be considered the subject-matter of a patent under the EPC.<sup>87</sup> In some commentators’ opinion, the criterion of the ‘technical effect’ has been interpreted rather loosely, while at the same time, the exclusion of article 52(2) EPC has been interpreted rather restrictively.<sup>88</sup> This, in combination to a poorly accessible body of prior art in the field of computer-implemented inventions, leads in turn to the grant of what some commentators refer to as ‘trivial patents’.<sup>89</sup> This problem, however, is not unique to patents on computer-implemented inventions.

Leaving the complex issue of the patentability of computer-implemented inventions to the appreciation of more expert scholars, let us concentrate here on the implications of granting patents on such inventions for the development of open source software. As mentioned previously, the open source community has consistently maintained that software patents are incompatible with the open source ideology. The foundation of the open source development model lies on the possibility for developers to share parts of the source code and to use the source code in one’s own work. This freedom is severely curtailed whenever a new piece of code ends up fulfilling the same function as that of a patented invention. In such circumstances, the manufacturing, use, and distribution of the potentially infringing code would be impossible without the patent holder’s authorisation, a requirement that goes against the philosophy of the open source development model. The open source community argues that patenting software would reduce the overall level of innovation in the field and may lead to a monopolisation of standards.<sup>90</sup>

---

<sup>84</sup> Guidelines for Examination in the EPO, C-IV, § 2.3.

<sup>85</sup> Tauchert, W., ‘Software-Patente und computerimplementierte Erfindungen’, JurPC Web-Dok. 6/2005, §§ 4-8.

<sup>86</sup> See Sedlmaier, R. and J. Gigerich, ‘Rechtliche Bedingungen und Risiken der Landeshauptstadt München für den Einsatz von Open-Source Software’, JurPC Web-Dok. 10/2005, § 80 and ff.; Case Law of the Boards of Appeal of the European Patent Office, Fourth Edition, Munich, EPO, 2002, § 1.1, pp. 2-6; and Singer, M. and D. Stauder, *European Patent Convention – A Commentary*, 3<sup>rd</sup> ed., (London, Sweet & Maxwell 2003), p. 73 and ff.; and the classic cases: T208/84, *O.J.E.P.O.* 1987, 14 (*VICOM*); T26/86, *O.J.E.P.O.* 1988 No. 19 (*Koch & Sterzel*); T769/92, *O.J.E.P.O.* 1995No. 525 (*SOHEI*).

<sup>87</sup> See T935/97 and T1173/97, *O.J.E.P.O.* 1999 No. 609 (*IBM patents*); T 931/95, *OJ* 10/2001 No. 441.

<sup>88</sup> Verkade 2004, *supra* note 10, p. 239.

<sup>89</sup> Tauchert 2005, *supra* note 85, § 45. The argument of ‘trivial patents’ is particularly strong in the United States, see: Evans and Layne-Farrar 2004, *supra* note 70, § 25.

<sup>90</sup> Valgaeren, E., ‘Open source-code en octrooien – van copyleft naar patentleft?’, 5 *Computerrecht* (2004), pp. 233-237, p. 234.

Unlike copyright protection, patent law generally protects the functionality of a computer program and not its expression. By conferring on its owner the exclusive right to manufacture, use, sell, and distribute the patented invention, the existence of a patent actually prevents any other computer programmer from independently developing a piece of software with a comparable functionality, even if the new software does not reproduce the lines of code of the patented software.<sup>91</sup> Moreover, several European national courts, including the Dutch Supreme Court, have recognised the general applicability of the doctrine of equivalents. This doctrine states that an element ('the equivalent element') can generally be considered as being equivalent to an element as expressed in a patent claim if, at the time of any alleged infringement, either of the following conditions is fulfilled in regard to the invention as claimed: 1) substantially the same function in substantially the same way and produces substantially the same result as the element as expressed in the claim; or 2) it is obvious to a person skilled in the art that the same result as that achieved by means of the element as expressed in the claim can be achieved by means of the equivalent element. Although this doctrine has yet to be applied in Europe to computer-implemented inventions, a computer programmer would not, according to this theory, be able to 'invent around' a patent, if the resulting computer code fulfilled substantially the same function in substantially the same way and produces substantially the same result as the patented invention.<sup>92</sup>

Since the core of the patent protection relates to the functionality of an invention, some commentators have maintained that, for the purposes of software development, a distinction should be made between object code and source code.<sup>93</sup> If the patent claim relates to a product or a machine, article 53(1)(a) of the Dutch Patent Act grants its owner the exclusive right to prohibit anyone from making, using, putting on the market or reselling, hiring out or delivering the patented product, or otherwise dealing in it commercially. A patented machine or product embodying software can only be infringed when the object code, not source code, is loaded into the memory of a computer to produce an equivalent functionality. If the patent claim relates to a process, article 53(1)(b) of the Dutch Patent Act grants its owner the exclusive right to prohibit anyone from using the patented process in or for his business or to use, put on the market, or resell, hire out or deliver the product obtained directly as a result of the use of the patented process, or otherwise deal in it commercially. Since the process patent primarily protects inventive technical methods, the prohibition right does not cover the production of a product, but rather the 'application' of the patented invention and the 'offering' for application of the invention.<sup>94</sup> As the authors Sedlmaier and Gigerich explain:

Das Programmieren von Software birgt stets das Risiko einer Patentverletzung. Die Gefahr einer Patentverletzung bezieht sich dabei aber weniger auf die Designstruktur oder die Kodierung selbst, als auf die Programmarchitektur und Funktionalität des jeweiligen Computerprogramms.<sup>95</sup>

In other words, the use, study, copy, or modification of the source code embodied in a computer-readable medium can hardly infringe a patent on a computer-implemented invention. The use of a patented computer-implemented invention in the development of new software also brings up the issue of interoperability. It could be argued that, since article 53(1) of the Patent Act does not prevent natural or legal persons from using a patented invention purely for internal or private research purposes, developers are in principle free to reverse

---

<sup>91</sup> Engelfriet, A., 'Open source software en octrooien: een moeilijke combinatie', 5 *BIE* (2003), pp. 204-208, p. 206.

<sup>92</sup> HR, 2 November 2001, *BIE* 2003/30 (*Kabelgeleidingsbuis*); HR, 29 March 2002, *BIE*, 2003, No. 14, blz. 99 (*Van Bentum/Kool*).

<sup>93</sup> Lin, D., Sag, M. and R.S. Laurie, 'Source Code versus Object Code: Patent Implications for the Open Source Community', 18 *Santa Clara Computer & High Tech. L.J.* (2002), pp. 235-254, p. 235.

<sup>94</sup> Sedlmaier and Gigerich 2005, *supra* note 86, § 166; Jaeger and Metzger 2002, *supra* note 14, p. 119.

<sup>95</sup> Sedlmaier and Gigerich 2005, *supra* note 86, § 148.

engineer a computer program for purposes of interoperability or otherwise, without the patent holder's authorisation. In this sense, the patent rules appear more flexible than the copyright rules on the subject. However, just as with copyright law, the private or internal use of a patented invention must not pursue commercial objectives.<sup>96</sup> The question is whether the resulting interoperable computer-implemented invention or software infringes the patented invention once it is put on the market. The answer, in our opinion, is a matter of factual appreciation.

The chance that a particular piece of code unwittingly infringes a patent is not purely theoretical.<sup>97</sup> The risk for a software developer of being involved in a patent infringement lawsuit and of having to start the development process from scratch is especially acute for small software firms or freelance developers who rarely have the sufficient resources to hire a patent lawyer to conduct a search prior to the development of new software. Of course, the fear of having to pay high damages as a result of a patent infringement suit may also play an important role in the software development process.<sup>98</sup> Contrary to most commentators however,<sup>99</sup> we believe that the risk of facing an infringement lawsuit may be greater for open source software developers than for developers of proprietary software, insofar as the disclosure of the source code that is typical for any open source project makes the detection of possible infringement much easier than would otherwise be the case. Nevertheless, the uncertainty comes above all from the fact that the law is still unclear on the patentability of computer-implemented inventions and that the quality of the patents delivered by the EPO or the national patent offices of the Member States often leaves something to be desired.<sup>100</sup>

## 2.2. Open source patenting strategy

Whether or not the European legislator will one day harmonise the rules on the patentability of computer-related inventions, the reality is that patents are actually being granted with respect to computer-implemented inventions both at the European and at national levels. Part of this reality is also that a relatively small number of very large companies hold the vast majority of patents issued with respect to computer-implemented inventions.<sup>101</sup> This means that, in order to avoid infringing another company's patent, (open source) software developers may be forced to obtain a licence on a patented invention before they can pursue their own development activities. Although no generalisation should be made in this regard, it may happen in practice that large companies will build-up impressive patent portfolios for strategic reasons, in order to gain leverage in cross-licensing negotiations. Patents may be used in an aggressive manner to fight competition by means of patents rather than by performance. Patents are said to be used in a 'strategic' way if the owner employs his patents merely to prevent competitors from using the invention, rather than to exploit the invention himself. In a broader sense, strategic use of patents could also be considered to include other actions specifically targeted at the obstruction of competitors.<sup>102</sup> As a result, smaller businesses and

---

<sup>96</sup> Jaeger and Metzger 2002, *supra* note 14, p. 117.

<sup>97</sup> Engelfriet 2003, *supra* note 91, p. 207; Jaeger and Metzger 2002, *supra* note 14, p. 113.

<sup>98</sup> Jaeger and Metzger 2002, *supra* note 14, p. 128.

<sup>99</sup> Knubben, B., *Software-octrooien – Stoppen of doorgaan met open source software?*, (The Hague, Programma OSOSS October 2004), p. 5; Tauchert 2005, *supra* note 85, § 49 and ff.; Sedlmaier and Giegrich 2005, *supra* note 86, § 164 and ff.; Engelfriet 2003, *supra* note 91, p. 207; Evans and Layne-Farrar 2004, *supra* note 70, § 69.

<sup>100</sup> Sedlmaier and Giegrich 2005, *supra* note 86, § 186.

<sup>101</sup> Bessen, J. and R.M. Hunt, *An Empirical Look At Software Patents*, (March 2004), p. 4 online: <http://swpat.ffii.org/papri/bessenhunt03/index.en.html>; and see the statistics held by the Free Information Infrastructure, available at <[http://swpat.ffii.org/patents/stats/app\\_stat.html](http://swpat.ffii.org/patents/stats/app_stat.html)>.

<sup>102</sup> Bakels and Hugenholtz 2002, *supra* note 73, p. 22.

individual freelance developers could be prevented from entering the market and from innovating further.<sup>103</sup>

In the context of open source software development, the fear of unwittingly infringing another company's patent by one's own developing activities is only as strong as the fear of incorporating another contributor's infringing code into a collective work. In both cases, the software developer(s) could be held liable for patent infringement at the close of a very costly litigation process. Although open source software developers are not often involved in patent infringement lawsuits, it does happen that distributors of CD-ROM's embodying open source programs are confronted with a patent holder's claim. One example is the Linux-distributor Red Hat, which had to remove all MP3-software from her products because it allegedly conflicted with a MP3 licensing scheme of Thomson Multimedia.<sup>104</sup> However, open source developers are not entirely helpless in front of holders of patents on conventional software. Besides taking an insurance policy against third party patent infringement claims, there are ways to minimise the risk of being confronted with the consequences of both 'strategic' patenting of conventional software developers and possible patent infringement lawsuits.

The first method consists, for the open source community, in developing a patenting strategy of its own. This includes the development of a patent portfolio, which would serve as an exchange item for cross-licensing and patent pools.<sup>105</sup> Many software companies, both open source and proprietary, pursue this strategy. As Bakels and Hugenholtz maintain, all patents serve to some extent a defensive purpose, since a patent owner can always prevent others from applying the technology he has developed. A purely defensive use of patents may be the filing of patents with the sole objective of creating an exchange item in negotiations with competitors. The patented software can be used to obtain a licence for another patent from a competitor who would otherwise be reluctant to do so or to create a patent pool with other companies.<sup>106</sup> Cross-licensing and patent pools are also an effective way to share technology.<sup>107</sup> Of course, developing a patent portfolio is not a realistic option for small businesses and independent programmers. And although the patenting of computer-implemented invention theoretically goes against the principles of the open source community, the competitive reality leaves the bigger players no other choice but to jump into the race and start developing their own patent portfolio.

The most important Linux-distributors, Red Hat Inc., have elected to adopt this same stance. It conceded to do so reluctantly because of the perceived inconsistency with the open source ideology. To the extent any party exercises a patent right with respect to open source, which reads on any claim of any patent held by Red Hat, Red Hat agrees to refrain from enforcing the infringed patent against such party for such exercise. The promise does not extend to any software, which is not open source, and any party exercising a patent right with respect to non-open source, which reads on any claims of any patent held by Red Hat, must obtain a license for the exercise of such rights from Red Hat. The promise does not extend to any party who institutes patent litigation against Red Hat with respect to a patent applicable to software (including a cross-claim or counterclaim to a lawsuit). No hardware per se is licensed hereunder.<sup>108</sup> In a similar vein, subscribing to the theory that the best defence is a good offence, the second largest seller of the Linux kernel, Novell, made it clear that any patent

---

<sup>103</sup> Evans and Layne-Farrar 2004, *supra* note 70, § 54.

<sup>104</sup> Engelfriet 2003, *supra* note 91, p. 207.

<sup>105</sup> Välimäki, M., 'A Practical Approach to the Problem of Open Source and Software Patents', *EIPR* (2004) 523-527, p. 526.

<sup>106</sup> Bakels and Hugenholtz 2002, *supra* note 73, p. 23.

<sup>107</sup> Välimäki 2004, *supra* note 105, p. 523; Evans and Layne-Farrar 2004, *supra* note 70, § 60.

<sup>108</sup> See Red Hat's patent policy, available at <[http://www.redhat.com/legal/patent\\_policy.html](http://www.redhat.com/legal/patent_policy.html)> site visited on 13 February 2005.

litigation against the Linux kernel or the open-source community would give Novell cause to check any accuser's own software against Novell's extensive portfolio of patents for possible retaliatory litigation.<sup>109</sup>

Similarly, Sun Microsystems announced that it would provide programmers free access to 1,600 patents as part of a plan to make an open-source version of its forthcoming Solaris 10 operating system.<sup>110</sup> The Solaris operating system is being released under the terms of the OSI-approved, CDDL (Common Development and Distribution License). One question that arises in this context is whether code released under terms of the CDDL can be used in combination with code released under the GPL. In January 2005, IBM has decided to let open-source developers use 500 software patents without fear of an infringement lawsuit, a new step in its encouragement of the collaborative programming philosophy. In August, the company had already pledged not to use its patent portfolio to attack Linux. IBM plans to grant royalty-free access to more patents in the future for open-source use. It also plans to release patents for use in open standards – a move that could make it easier to embrace such standards within open-source and proprietary software.<sup>111</sup> Other big software companies may decide to follow the trend and offer a portion of their patent portfolio to open-source developers.

### **2.3. Open source licensing strategy**

Another way to reduce the risks associated with the use of patented software is to regulate the consequences of the use of such software inside the open source licence. An open source licence could provide for example, for a guarantee against third party infringement claims, for a prohibition to further distribute patented software, or for a free non-exclusive licence to use any software patented by an open source developer. Not all open source licences contain such language however. The BSD licence is one of them, in contrast with the GPL and the Mozilla Public Licence. The details of each licence are given below, but it is worth noting, however, that the obligations laid down in the GPL and the Mozilla Public Licence are directed strictly at the licensee. The licensor makes under these licences no representation guaranteeing that the code does not infringe third party patents, nor does he undertake not to obtain patent protection on the software.

The GPL is mainly concerned with the consequences of the incorporation of patented software into code that is distributed under the terms of the GPL. It also discourages developers from obtaining a patent on their computer-related invention. The preamble of the GPL states that ‘any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.’ Accordingly, article 7 of the GPL stipulates the following:

“If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you

---

<sup>109</sup> L. Greenemeier, ‘Novell Warns Against Linux Patent Suits’, CRN, 12 October 2004. <<http://www.crn.com/sections/breakingnews/dailyarchives.jhtml?articleId=49901223>>.

<sup>110</sup> Stephen Shankland, ‘Sun's open-source gamble’, February 7, 2005, *CNET News.com*, available at <[http://news.com.com/Suns+open-source+gamble/2008-1082\\_3-5564283.html](http://news.com.com/Suns+open-source+gamble/2008-1082_3-5564283.html)> site visited on 13 February 2005.

<sup>111</sup> Stephen Shankland, ‘IBM offers 500 patents for open-source use’, 10 January 2005, *CNET News.com*, online: <[http://news.com.com/IBM+offers+500+patents+for+open-source+use/2100-7344\\_3-5524680.html](http://news.com.com/IBM+offers+500+patents+for+open-source+use/2100-7344_3-5524680.html)>, (visited on 13 February 2005).

cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License’.

As Rosen points out, article 7 could have been couched in much clearer terms. Uncertain is what the licensor means by ‘any other pertinent obligations’ and what ‘obligations under this Licence’ may be contradicted by the court judgment.<sup>112</sup> Article 7 of the GPL must be read in conjunction with article 6 of the GPL, which provides for the ‘copyleft’ effect of the licence. When read together, they reveal somewhat of a contradiction: a programmer cannot use the licence in circumstances where he is restricting the rights of those beneath him, yet he can! Along with O’Sullivan, we consider that the somewhat obscure drafting of this section would need to be improved in any future versions of the GPL. The purpose of the section is, of course, to ensure that the free software distribution system is maintained, which appears to be an appeal to customary practices underlying the GPL.<sup>113</sup> It is also apparent from article 7, however, that it takes more than the threat of patent infringement to invoke this provision. An actual patent dispute has to be alleged and either litigated or settled.<sup>114</sup>

The Mozilla Public Licence is much more detailed and complex. In addition to regulating the consequences of the incorporation of patented software into code that is distributed under the terms of the MPL, it also contains an obligation to inform subsequent contributors of the existence of a patent right on part or whole of the code. According to article 2.1(b) of the licence, the initial developer grants the licensee the right to make, use or sell the original code in a specific embodiment of software, without limiting the initial developer’s right to exclude others from making, using or selling other embodiments in other software. Article 2.2(b) of the licence deals with modifications submitted by contributors who are licensees of the original code. Each contributor grants a reciprocal licence for his patents to allow modifications to be made, used, or sold either alone or in combination with the original code.<sup>115</sup>

The Mozilla Public Licence also contains a provision dealing with third party patent claims. Pursuant to article 3(4) of the Mozilla Public Licence, if a contributor who has knowledge that

---

<sup>112</sup> Rosen, L., *Open Source Licensing – Software Freedom and Intellectual Property Law*, (Upper Sadle River, Prentice Hall, 2004), p. 134.

<sup>113</sup> O’Sullivan 2004, *supra* note 43, p. 344.

<sup>114</sup> Rosen 2004, *supra* note 112, p. 134.

<sup>115</sup> *Id.*, p. 152.

a license under a third party's intellectual property rights is required to exercise the rights granted under the licence, he must include a text file with the source code distribution, describing the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If the contributor obtains such knowledge after the modification is made available, the contributor must promptly modify the file in all copies he makes available thereafter and must take other reasonable steps (such as notifying appropriate mailing lists or newsgroups) to inform those who received the covered code that new knowledge has been obtained. Similarly, if the contributor's modifications include an application-programming interface (API) and he has knowledge of patent licenses, which are reasonably necessary to implement that API, he must also include this information in the file. Article 3(4)(c) of the Mozilla Public Licence states that the contributor represents that, except as disclosed pursuant to article 3(4)(a), he believes that his modifications are his original creation and that he has sufficient rights to grant the rights conveyed by this license. Note that the initial developer is not obliged under the licence to make a similar representation. Although a guarantee against third-party infringement claims says nothing about whether or not the software infringes the rights of a third party, it does give the licensee recourse against his contracting partner, should it be the case. The guarantee is, of course, only worth as much as the economic strength of the contracting partner.<sup>116</sup> Finally, article 8 of the Mozilla Public Licence states that, should a licensee file a patent infringement lawsuit against the initial developer or a contributor, all copyright and patent licences on the software shall be terminated prospectively.

### 3. Enforcement of Open Source Licences

The previous sections have highlighted the respective rights and obligations of the parties under the most commonly used open source licences. But a licence is of no use if the parties do not live up to their obligations or if it is not enforced in practice. As we have seen in this paper, the main difference between open source licences and proprietary software licences lies first and foremost in the freedoms that the former type of licence grants to users and in the obligation to make the source code available to fellow developers. As Rosen rightly points out in the context of open source licences, why would a licensor who grants everyone the permission to copy, modify, and distribute the software, complain about someone doing these things? And why would a licensee who receives software with essentially unlimited rights to it need to demand even more from the licensor? When the software is distributed free of charge and is devoid of any guarantee, why would anyone take the trouble to sue?<sup>117</sup> But the occurrence of a dispute is not unthinkable, taking the enforcement of the copyleft clause or of the obligation to make the source code available to fellow developers, as examples.

Relatively little case law has emerged so far with respect to the enforcement of open source software licences. The lack of relevant case law can to a large extent be attributed to the philosophy underlying the open source software movement: the GPL's enforcement mechanisms are informal and dependent on customary practices rather than on legal processes. There is significant social pressure on participants to comply with the licence, owing to the threat of negative publicity and litigation. As Moglen stresses: 'Our way of doing law has been as unusual as our way of doing software, but that's just the point'.<sup>118</sup> Moglen uttered this remark in reaction to the suggestion coming from certain circles that open source licences, such as the GPL, are not unenforceable in law. On the basis of the findings in this study, we believe on the contrary that the terms of a typical open source licence are valid

---

<sup>116</sup> Sedlmaier and Gigerich 2005, *supra* note 86, § 173.

<sup>117</sup> Rosen 2004, *supra* note 112, p. 269.

<sup>118</sup> Moglen, E., 'Enforcing the GPL, II', 10 *Linux User* (2001), available at: <http://emoglen.law.columbia.edu/publications/lu-13.html>.

and enforceable, provided of course that the agreement is validly concluded between the parties. Along with O’Sullivan, we think that if courts refused to uphold the validity of an open source licence, the consequences for the open source community which relies on such a licence for the continuity of its developing practices could be devastating. The strength of the GPL and other open source licences lies precisely in their moral force. If a court were to decide in favour of a violator of the licence, this moral force would be depleted, since much of the threat of negative publicity, which actually ensures compliance with open source licences, would simply fade away.<sup>119</sup>

The informal character of the enforcement process of the GPL is clearly illustrated by Moglen’s description of the manner in which alleged violations are dealt with in the United States:

“So what happens when the GPL is violated? With software for which the Free Software Foundation holds the copyright (either because we wrote the programs in the first place, or because free software authors have assigned us the copyright, in order to take advantage of our expertise in protecting their software's freedom), the first step is a report, usually received by email to [license-violation@gnu.org](mailto:license-violation@gnu.org). We ask the reporters of violations to help us establish necessary facts, and then we conduct whatever further investigation is required.

We reach this stage dozens of times a year. A quiet initial contact is usually sufficient to resolve the problem. Parties thought they were complying with GPL, and are pleased to follow advice on the correction of an error. Sometimes, however, we believe that confidence-building measures will be required, because the scale of the violation or its persistence in time makes mere voluntary compliance insufficient. In such situations we work with organizations to establish GPL-compliance programs within their enterprises, led by senior managers who report to us, and directly to their enterprises' managing boards, regularly. In particularly complex cases, we have sometimes insisted upon measures that would make subsequent judicial enforcement simple and rapid in the event of future violation.’

In view of the broad freedom granted to the licensee under most open source software licences, the disputes most likely to arise between parties will concern either the licensee’s failure to distribute modified code under the terms of the GPL, thereby ‘closing-up’ the code, or his failure to make the source code available. Both types of omissions were actually at the heart of the decision of the District Court of Munich in the case *Netfilter v. Sitecom*,<sup>120</sup> which represents the first judgment ever upholding the validity of the GPL. In this case, the plaintiff, Mr. Harald Welte, as the ‘maintainer’ of the open source project ‘netfilter/iptables’, offered the software for download under the conditions of the GPL to which reference was made on the corresponding website. The defendant was the German subsidiary of the Dutch corporation Sitecom Europe B.V., which advertised and distributed a Wireless Network Broadband Router through its website. The firmware offered for download along with the Router contained the Linux kernel, as well as the ‘netfilter/iptables’ in object code, the software modules ‘PPTP helper for connection tracking and NAT’ and ‘IRC helper for connection tracking and NAT’, all of which were developed by the plaintiff. The litigation arose mainly from the failure by Sitecom to mention the fact that the firmware contained software licensed under the GPL, to join the text of the licence and to distribute the source

---

<sup>119</sup> O’Sullivan 2004, *supra* note 43, p. 344.

<sup>120</sup> District Court of Munich I, 19 May 2004, (*Netfilter v. Sitecom*), *Computerrecht* 2004/10, pp. 774- 776, with annotation by T. Hoeren and A. Metzger.

code of the software ‘netfilter/iptables’. The plaintiff’s counsel argued that the defendant infringed plaintiff’s copyright by offering the software ‘netfilter/iptables’ for download and promoting its distribution without abiding by the licence conditions of the GPL. In its final ruling, the District Court of Munich concluded that the distribution of the software without complying with the conditions of the GPL constitutes an infringement of copyright give rise to a claim for injunctive relief pursuant to the German Copyright Act.<sup>121</sup> In appeal of an injunctive order, the Court upheld articles 2, 3, 4 of the GPL as valid general conditions of sale that are in compliance with the provisions of the Copyright Act.

While this court decision has been acclaimed as finally bringing legal certainty for the open source community, is it really the case? The decision of the District Court of Munich is only a first instance ruling, which brings clarity only with respect to the specific points raised by the case. Moreover, several clauses of the GPL remained unaddressed, such as the clauses governing the warranty disclaimer and the limitation of liability.<sup>122</sup> Several legal issues raised by the enforcement of the GPL under German law were also left unexamined: for example, the question of the applicable law, of the use of general conditions in a foreign language and of the automatic termination clause in case of breach of some of the obligations under the contract. Nevertheless, this first decision is important insofar as it contributes in a significant way to the acceptance of the phenomenon of open source licensing.<sup>123</sup> With respect to general awareness and acceptance of open source licences, it is interesting to note that, at the close of the legal dispute between Netfilter and Sitecom, the ‘maintainer’ of the ‘netfilter/iptables’ project, Mr. Harald Welte, started a website called [www.gpl-violations.org](http://www.gpl-violations.org). The site’s main purpose is to raise public awareness of the infringing use of free software, thereby putting pressure on the infringers; to give users who detect or suspect that GPL-licensed software is being misused, a way to report it to the copyright holders, as this is the first step in enabling the copyright holders to push for license compliance; and to assist copyright holders in any action against GPL infringing organisations.

In the United States, while the GPL and other open source licences have been the object of some consideration by the courts – mainly in the context of interim orders and injunctive relief actions – their enforceability must still be confirmed in a court decision. Most disputes are settled informally out of court. They concern either the licensee’s failure to distribute modified code under the terms of the GPL, thereby ‘closing-up’ the code, or his failure to make the source code available. One of these disputes opposed Progress Software Corporation to MySQL AB, in which MySQL AB claimed, among other things, that Progress Software had allegedly violated MySQL’s programs licensed under the GPL, by distributing them without giving access to the source code. The parties settled out of court in 2002. In a similar case, Drew Technologies filed a lawsuit against the Society of Automotive Engineers in October 2003 for allegedly having distributed software programs without reference to the GPL and without making the source code available.<sup>124</sup> Upon completion, the software was claimed by a standards organization as its own copyrighted work although it had been developed by third party engineers who had released it under the GPL. In the end, by way of settlement and not judgment, the standards group accepted that the GPL governed, and it paid the engineering group \$75,000.<sup>125</sup>

<sup>121</sup> Höppner, J., ‘The GPL prevails: An analysis of the first-ever Court decision on the validity and effectivity of the GPL’, 1 *SCRIPT-ed* (2004), pp. 1-5, p. 2; Visser, E.N.M., ‘Netfilter versus Sitecom. Noot bij de uitspraak van Rechtbank te München 19 mei 2004’, 5 *JAVI* 2004, pp. 186-188, p. 186;

<sup>122</sup> Visser 2004a, *supra* note 121, p. 188.

<sup>123</sup> Schultz, C., ‘Open Source Software vor Gericht’, 9 *Multimedia und Recht* (2004), pp. 573-574, p. 574; Visser 2004a, *supra* note 121, p. 188.

<sup>124</sup> *Drew Technologies, Inc. v. Society of Automotive Engineers, Inc., et al.*, No. 03-CV-74535-NGE-PJK (E.D. Mi.).

<sup>125</sup> See ‘A GPL Win in Michigan – DrewTech v. SAE’, *Growklaw*, March 20 2005, available at <<http://www.groklaw.net/article.php?story=20050225223848129>>, site visited on April 05 2005.

It may also happen that a proprietary software company will institute proceedings against an open source software developer for having included proprietary code into an open source project, thereby 'opening up' closed software. This is essentially the core of the legal saga between the SCO Group ('SCO') and IBM. In March 2003, SCO introduced an action against IBM, alleging breach of contract, unfair competition and misappropriation of trade secrets. SCO's claim was essentially that IBM improperly destroyed the economic value of their intellectual property in UNIX by transferring portions of the proprietary UNIX source code into Linux, which is open-source. IBM launched a counter-suit against SCO in August 2003. They alleged, among other things, breach of contract, unfair competition, and infringement of IBM's patents and copyrights. The heart of IBM's cross-complaint was that SCO breached the GNU General Public License. SCO answered IBM's cross-complaint in October 2003, averring that the GPL is unenforceable and inapplicable. The legal battle between SCO and IBM is still raging and new developments in the case are reported almost daily.<sup>126</sup> It is therefore still premature to say what impact the dispute between SCO and IBM will have on Linux or on the development of open source software. Along with other commentators, we believe however, that this case merits close attention for it has the potential of derailing the momentum that has been building behind the open source movement.<sup>127</sup> It is also clear that the uncertainty regarding the chain of title of the UNIX code is at the root of the whole dispute, for in SCO's case, the issue of copyright ownership depends largely on contract interpretation. The scale of the litigation between SCO and IBM should serve as a lesson to anyone, who assigns or licenses rights with respect to software, to always carefully identify the object of the transfer inside the agreement so as to avoid future disputes.

## Conclusion

Are open source software licences the sign of a gradual paradigm-shift within the world of intellectual property? We believe so. This form of licensing reflects the dire need of part of the software community to enable each member to share knowledge and to actively build upon one another's programs, with the belief that better programs will be created more rapidly and at lower cost. Interestingly, the paradigm-shift brought about in the context of open source software licences lies not so much in the urge to reform the law, but rather in the manner in which rights owners exercise the exclusive rights conferred to them under the law. Indeed, open access licences would seem to successfully make use of the legal rules on copyright. Indeed, far from rejecting the rules of copyright law, the open source movement relies on the application of these rules to set their own 'open' terms of use for protected software. The key terms in open source licences have been designed to take account of the fact that the traditional distinction between creators and users of software has essentially vanished within the open source community: users are creators and vice versa. In practice, the power to grant users the freedom to use, reproduce, and modify the software, and the freedom to distribute or re-distribute the work falls under the prerogatives of the exclusive rights holder would appear to be fully compatible with existing copyright law. Even the prohibition to charge a royalty for the distribution of the software poses, in our opinion, no problem under copyright law.

The situation is somewhat different with respect to patent law. The open source community has consistently maintained that software patents are incompatible with the open source ideology. The foundation of the open source development model lies on the possibility for

---

<sup>126</sup> See <<http://www.groklaw.net>>.

<sup>127</sup> Goettsch, K.D., 'SCO Group v. IBM: The Future of Open-Source Software', 2003 *University of Illinois Journal of Law Technology and Policy* (2003), pp. 581-588, p. 588; Determann, L. and A. Coan, 'Spoiled Code', *Computer Law Review International* (2003), pp. 161-167, p. 167.

developers to share parts of the source code and to use the source code in one's own work. This freedom is severely curtailed whenever a new piece of code ends up fulfilling the same function as that of a patented invention. Moreover, the risk of facing an infringement lawsuit may be greater for open source software developers than for developers of proprietary software, insofar as the disclosure of the source code that is typical for any open source project makes the detection of possible infringement much easier than would otherwise be the case. Nevertheless, the uncertainty comes above all from the fact that the law is still unclear on the patentability of computer-implemented inventions and that the quality of the patents delivered by the EPO or the national patent offices of the Member States often leaves to be desired. This problem, however, is not unique to patents on computer-implemented inventions.

A unique feature of open source software licences is probably their method of enforcement, which relies so heavily on the good will of their users, as a direct consequence of the philosophy underlying the open source software movement: the GPL's enforcement mechanisms are informal and dependent on customary practices rather than on legal processes. So far, this form of enforcement seems to be efficient and sufficient. Time will tell if software developers will remain so well behaved, as the community engaged in open source software licensing grows and as the commercial success of their products increases.

This paradigm-shift is not confined exclusively to the software sector, for the need to share knowledge can equally be felt within the scientific and artistic worlds. Indeed, the open source movement has inspired a variety of similar distribution models in the realms of science and art, which are commonly referred to as 'Open Access' or 'Open Content'. The principles of Open Access have been enshrined in three declarations: the Declaration of the Budapest Open Access Initiative (February 2002), the Bethesda Statement on Open Access Publishing (June 2003), and the Berlin Declaration on Open Access to Knowledge in the Sciences and Humanities (October 2003). A highly successful application of the open content model in the realm of the arts is the Creative Commons initiative ([creativecommons.org](http://creativecommons.org)). Creative Commons has developed a series of standard-form licenses that enable authors of literary, musical or audiovisual works to permit wide dissemination and transformative uses of their works, without forfeiting copyright. While copyright law creates the default rule of *All Rights Reserved*, making permission necessary for each and every use of a work, Creative Commons seeks to facilitate an environment in which *Some Rights Reserved* or even *No Rights Reserved* become the norm. The Creative Commons licensing scheme, which has been greatly inspired by the open source movement, is designed to meet the diverse preferences of authors, and at the same time keep it simple and easy to employ. The mechanism for achieving this goal is through a standardized and automated licensing infrastructure.

Of course, a number of additional legal issues need to be addressed in order to ensure the most efficient deployment of open content licences in the market place. Issues to be resolved include the importance of open standards and the problem of the proliferation of licenses, which threatens the compatibility and transparency of the open content model. Another question that arises relates to the perpetuity clause ('copyleft'), which imposes the obligation on the licensee to distribute modified versions of the work subject to the same conditions of use. Is such a clause valid under civil law?