Spring 2013

The Granting Clause and Intellectual Property Rights Management in Open-Source Software Licensing

Vikrant N. Vasudeva
George Washington University, Washington D.C., VIKRANTVASUDEVA@GMAIL.COM

Follow this and additional works at: http://www.repository.law.indiana.edu/ipt

Part of the Antitrust and Trade Regulation Commons, Commercial Law Commons, Computer Law Commons, Contracts Commons, Intellectual Property Law Commons, International Law Commons, Internet Law Commons, Law and Economics Commons, Legislation Commons, and the Litigation Commons

JEROME HALL LAW LIBRARY
INDIANA UNIVERSITY
Maurer School of Law
Bloomington
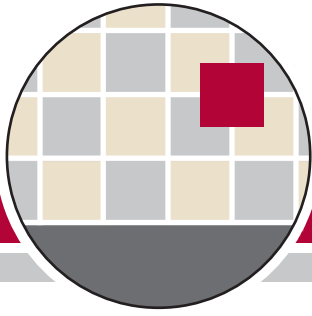
The Granting Clause and Intellectual Property Rights Management in Open-Source Software Licensing

Vikrant Narayan Vasudeva[*†‡§]

Introduction

Licensing law is extremely diverse, complex, and with its fair share of abstract zones. The granting clause is tasked with navigating this confusing and polemic landscape. The sum total of the terms in the granting clause determines the scope of the license. "The net result of such arrangements is that the grantee can, vis-à-vis the grantor, engage in conduct that the grantor might—if its rights are valid and the conduct violates them—otherwise prevent."[1]

In software licensing, the major thrust is on the intellectual property rights involved. The traditional copyright grant based on the rights granted by copyright law controls the rights to copy, distribute, and prepare derivative works.[2] Similarly, the traditional patent grant based on the rights granted by patent law covers the rights to make, use, sell or offer for sale, and import.[3] A trademark grant covers goods or services associated with a particular mark.[4]

---

1. 2 Roger M. Milgrim & Eric E. Bensen, Milgrim on Licensing § 15.00 (Matthew Bender & Co. 2012).
2. *See* 17 U.S.C. § 106 (2011); *Copyright Act 1968* (Cth) s 31 (Austl.).
3. *See* 35 U.S.C. § 154 (2011); *Patents Act 1990* (Cth) s 13 (Austl.).
4. *See* 15 U.S.C. §1127 (1946).

The open-source movement designed a counterintuitive licensing system based on the same legal premise as for closed source software but to different ends.

I. The Granting Clause

The granting clause should highlight the flow of rights and obligations from the licensor to the licensee and vice-versa for the various intellectual property involved. The character of the license should be made clear—whether exclusive, non-exclusive, sole, etc. Whether this grant includes subsidiaries, controlled companies, and affiliates should also be made clear. It is advisable to structure separate sub-classes for the different intellectual property under consideration in the license. Mentioning the term of the license is also preferable. Scope of the license should be clarified viz. geographic regions and markets, fields of use and products, sublicensing, royalties, etc. Furthermore, the regulation of improvements should be ascertained.

Each open-source license's grant of rights is drafted according to its respective affiliation's ideology. However, all open-source licenses have a common thread running through them. As Robert W. Gomulkiewicz states:

> Hackers seem to agree on the basic objectives of open source licensing. There are four fundamental rights that an open source license needs to grant: First, access to source code; second, the right to run the software for any purpose; third, the right to change the software in any way; fourth, the right to redistribute the original software and any derivatives.[5]

A. Intellectual Property Grant

From the perspective of open-source software development and intellectual property rights, GPL has several interpretative ambiguities. It contains a general grant of rights; so does the BSD License. The MPLv1.1 on the other hand has addressed the grant of rights at length, segregating it into "The Initial Developer Grant"[6] and the "Contributor Grant."[7] The Artistic License, the Open Software License, and the Apache License have taken this grant of rights a degree further by specifically segregating the grant of rights by fields in the intellectual property arena.

1. Copyright Grant

The BSD License is markedly abrupt in its grant of rights. It simply grants the right to redistribute and use. The right to modify is granted as a follow-on concession and is

---

5. Robert W. Gomulkiewicz, *De-Bugging Open Source Software Licensing*, 64 U. Pitt. L. Rev. 75, 81 (2002).

6. *Mozilla Public License Version 1.1*, Mozilla § 2.1, http://www.mozilla.org/MPL/1.1/ (last visited Apr. 25, 2011) [hereinafter MPLv1.1].

7. *Id*. at § 2.2.

not an affirmative grant. Distribution is an exclusive right under copyright law but not redistribution. Thus, redistribution would be interpreted to include the right to create copies and the right to distribution. Lawrence Rosen speculates as to the license thus: "[S]ince the word *modification* later in the sentence implies *derivative work*, I assume that the license allows the copying and distribution of both the original and derivative works."[8]

GPLv2 has scripted its license grant in a manner that creates unnecessary ambiguity. The grant stretches over three sections. While section 0 has been drafted in the negative,[9] sections 1 and 2[10] are in the positive. Essentially GPLv2 grants the rights to copy, distribute, and modify. These three rights conform to rights granted by copyright law; the other rights granted by copyright law but not mentioned in the license are not that relevant to software and particularly open-source software development.

The right to modify under GPLv2 poses a singularly unclear scenario. The right is primarily based on the concept of what constitutes a "work based on the Program"[11] and derivative works for the purpose of GPLv2. Irregular drafting practice has given rise to unending discussions as to whether GPLv2 grants the right only to create modifications, the right to create any derivative works, or even more. The modification right's deep linkage to the copyleft concept further aggravates the issue.[12]

Setting aside the vexatious discussion of what constitutes a "work based on the Program," the GPLv2's intention is explicit when it states that it seeks to "control the distribution of derivative or collective works based on the Program."[13] GPL has an expansive scope as compared to the rights granted under copyright statutes. Copyright statutes simply prohibit preparation of derivative works without approval; GPL is more invasive, seeking control over derivative works too. This might open the GPL to a copyright misuse action.

---

8. Lawrence Rosen, Open Source Licensing: Software Freedom and Intellectual Property Law 78 (2004), *available at* http://www.rosenlaw.com/oslbook.htm.

9. Free Software Found., *GNU General Public License, Version 2*, GNU Operating Sys. § 0 (June 1991), http://www.gnu.org/licenses/gpl-2.0.html [hereinafter GPLv2] ("Activities other than copying, distribution and modification are not covered by this License; they are outside its scope.").

10. *Id*. at § 1 ("You may copy and distribute verbatim copies . . . ."); *id*. at § 2 ("You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work . . . .").

11. *Id*. at § 0 (a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications).

12. Reciprocal obligation is known by various informal terms, *viz*. "Viral", "Taint", "Infectious", and "Copyleft". Copyleft entails propagation of code and its modifications only according to the specified directions and none other. Copyleft is a way of using of the copyright on the program. It is a general concept, one of whose specific and most prominent implementation is in GNU GPL: developers can take advantage of the contributions reflected in an existing piece of GPL code, provided they make a reciprocal contribution of their modifications under the GPL; *see* Ronald J. Mann, "Commercializing Open Source Software: Do Property Rights Still Matter?", 20 *Harv. J. Law & Tec* 1, 16 (2006).

The MPLv1.1 splits its grant of rights, segregating it into "The Initial Developer Grant"[14] and the "Contributor Grant."[15] It refers to copyright by exclusion, citing it as "intellectual property rights (other than patent or trademark)."[16] However, it grants the full set of rights under copyright law.[17]

The OSLv3.0 and the Apache License, v2.0 are well-structured licenses with separate copyright grants. They grant all the rights under copyright law in an unambiguous manner.[18] While the Apache License, v2.0 scripts its own definition of Derivative Works,[19] the OSLv3.0 adopts that of the copyright law.[20] In both the cases, adherence to licensing practice (unlike as in GPLv2) greatly eases the uncertainty in determining what constitutes derivative works, Furthermore, OSLv3.0 recognizes external deployment as equivalent to distribution.[21] External deployment entails making available the licensed code to a third party, essentially over a network.[22] This clause ensures that entities are not able to escape the application of OSL by avoiding a physical distribution. Rosen notes that this clause is a waivable license condition.[23]

GPLv3 maintains the same set of rights as granted under GPLv2 albeit under new terminology; copy, distribute, and modify have been replaced by propagate and convey.[24] The Free Software Foundation made these changes as an attempt at achieving universal standardization and clearing doubts that had created interpretative ambiguity in GPLv2.[25] Propagation[26] is the central aspect on which the open-source policy—especially the GPL—hinges and is essential to its theme. The term "propagate" was used to restrict the scope of rights available under copyright law.[27] The definition enumerates two exceptions which

---

13. *Id*. at § 2.
14. MPLv1.1, *supra* note 6, at § 2.1.
15. *Id*. at § 2.2.
16. *Id*. at §§ 2.1(a), 2.2(a).
17. *Id*.
18. *See* Lawrence Rosen, *The Open Software License 3.0 (OSL-3.0)*, Open Source Initiative § 1, http://opensource.org/licenses/OSL-3.0 (last visited Apr. 25, 2011) [hereinafter OSLv3.0].
19. *Apache License, Version 2.0*, Apache Software Found. § 1 (Jan. 2004), http://www.apache.org/licenses/LICENSE-2.0 [hereinafter Apache License, v2.0].
20. OSLv3.0, *supra* note 18, at § 1(b).
21. *Id*. at § 5.
22. *Id*.
23. Lawrence Rosen, OSL 3.0: A Better License for Open Source Software 10 (2007), http://www.rosenlaw.com/pdf-files/OSL3.0-explained.pdf.
24. *See* Free Software Found., *GNU General Public License, Version 3*, GNU Operating Sys. § 2 (June 29, 2007), http://www.gnu.org/licenses/gpl-3.0.html [hereinafter GPLv3].
25. *See* Free Software Found., *Frequently Asked Questions About the GNU Licenses*, GNU Operating Sys., http://www.gnu.org/licenses/gpl-faq.html (last visited Apr. 25, 2011).
26. *See* GPLv3, *supra* note 24, at § 0 ("To 'propagate' a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.").

do not constitute propagation—the first being "executing it on a computer" and second, "modifying a private copy."[28] Ordinarily, under copyright law, even these two exceptions would constitute copyright infringement if done without initial permission.[29] However, via this contractual license, the scope of rights available under copyright law has been restricted.

This definition is not without ambiguity; what constitutes a private copy is not defined.[30] Presumably, a private copy is a copy personal to the licensee. However, the scope of such a license is dependent on the classification of the licensee. It could be an individual, a corporation, or a group of related companies.

By mentioning only copyright law, GPLv3 makes it clear that the concept of propagation does not govern patent law under the license, for which a separate section 11 has been structured.

"Convey" in GPLv3 is essentially a substitute for the term "distribute" in GPLv2.[31] Conveyance of software is directly linked to propagation. GPLv3 states that "[t]o 'convey' a work means any kind of propagation that enables other parties to make or receive copies."[32] Thus, of the examples of propagation mentioned, distribution and making available to the public would amount to conveyance amongst certain other activities.[33] Learning its lessons from GPLv2, the Free Software Foundation clarifies that merely making copies for self-use[34] or for intra-company use[35] would not amount to conveyance. However, this still does not clarify situations such as mergers, acquisitions, joint ventures, etc.[36] As regards digital transmission, GPLv3 qualifies that "[m]ere interaction with a user through a computer network, with no transfer of a copy, is not conveying."[37]

GPLv3 acknowledges fair use rights unlike GPLv2.[38] Taking a page from MPLv1.1, GPLv3 has included a definition which states that "'Copyright' also means copyright-like laws that apply to other kinds of works, such as semiconductor masks."[39] This seemingly corresponds with MPLv1.1's exclusionary reference to "intellectual property rights (other

---

27. *Id*.
28. *Id*.
29. *See* 17 U.S.C. §§ 106, 117 (2011).
30. GPLv3, *supra* note 24, at § 0.
31. *See* Free Software Found., *supra* note 25.
32. *Id*.
33. *Id*.
34. *See* Free Software Found., *supra* note 25.
35. *Id*.
36. *See* Comm. on Info. & Tech. Law, N.Y.C. Bar Ass'n, *Open Source Software and the Free Software Movement*, 61 Rec. Ass'n B. City N.Y. 325, 328-30 (2006) (comments of the New York City Bar Association to GPL Version 3, sent to the Free Software Foundation).
37. GPLv3, *supra* note 24, at § 0.
38. *Id*. at § 2.

than patent or trademark)."[40] GPLv3 tries to reduce the ambiguities that plagued GPLv2. However, it still refrains from being completely explicit.

2. Patent Grant

While the MPLv1.1, the OSLv3.0, and Apache License, v2.0 contain explicit patent grants, GPLv2 entirely lacks a patent grant.

GPLv2's Preamble voices Richard Stallman's opposition to patents.[41] Section 7 of GPLv2 further elucidates the concept highlighted in the Preamble, stating that nothing excuses the licensee from the license's conditions. The subsequent clauses further reemphasize that, "[i]f [the Licensee] cannot distribute so as to satisfy simultaneously [its] obligations under this License and any other pertinent obligations, then as a consequence [it] may not distribute the Program at all."[42]

Though GPLv2 lacks an express patent license, some scholars have interpreted the Preamble and S.§ 7 of GPLv2 to create an implied patent license.[43] However, scholars also agree that the GPL's Preamble lacks legal effect,[44] or that the scope of such an implied license would be difficult to determine.[45]

GPLv2 allows its users to copy, distribute, and modify the licensed code. The right to run the program, is mentioned separately as being "not restricted";[46] worse still, the clause not restricting the "act of running the Program"[47] is preceded by a clause which explicitly states that "[a]ctivities other than copying, distribution and modification are not covered by this License; they are outside its scope."[48] However, the scenario where the act of running is not allowed is highly unlikely; after all, it is one of the founding principles of the Free Software Definition.[49] "The ambiguity here might be just a matter of phrasing the sentence in the negative rather

---

39. *Id*. at § 0.
40. MPLv1.1, *supra* note 6, at §§ 2.1(a), 2.2(a).
41. GPLv2, *supra* note 9, at pmbl. ("[A]ny 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.").
42. *Id*. at § 7.
43. *See* Clark D. Asay, *The General Public License Version 3.0: Making or Breaking the FOSS Movement?*, 14 Mich. Telecomm. & Tech. L. Rev. 265, 288 (2008); Rosen, *supra* note 8, at 126; Terry J. Ilardi, *Mass Licensing—Part 2: Open Source Software Licensing*, *in* Patent and High Technology Licensing 2005, at 279, 295 (PLI Intellectual Prop., Course Handbook Ser. No. G-831, 2005).
44. *See* Rosen, *supra* note 8, at 109, 111.
45. Richard Fontana et al., *A Legal Issues Primer for Open Source and Free Software Projects*, Software Freedom Law Ctr. 25 (Mar. 3, 2008), http://www.softwarefreedom.org/resources/2008/foss-primer.pdf.
46. GPLv2, *supra* note 9, at § 0.
47. *Id*.
48. *Id*.
49. *See* Free Software Found., *The Free Software Definition*, GNU Operating Sys., http://www.gnu.org/philosophy/free-sw.html (last visited Apr. 25, 2011).

than the positive."[50] However, it exists and unnecessarily clouds the issue. Such ambiguous licensing language can lead to interpretations that certain rights may not have the same level of assurance attached to them, or that there may exist unnamed or implied rights or obligations.

Rosen is of the view that right to run might actually be a right to use.[51] The BSD License in its grant of rights too grants the right to use.[52] However, a predicament with such a reading is that the right to use is not an exclusive right granted under copyright law; it is one of the rights granted under patent law. Both the GPLv2 and the BSD license are asserted to be primarily copyright licenses.[53]

GPLv3 is much more explicit than GPLv2 and has shifted its language from the negative to the positive. It affirmatively grants the right to run.[54] Furthermore, it also grants the right to make.[55] GPLv3 also includes a patent clause that grants as far as any contributions are concerned all the rights governed by patent law.[56] These changes mark GPL's transition from a copyright license to an intellectual property license.

The MPLv1.1 makes a distinction between patent-trademarks grant and copyright grant. The copyright grant extends in the case of the Initial Developer Grant to "Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work,"[57] and in the case of the Contributor Grant to "Modifications . . . (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work."[58] In comparison, the patent grant extends only to "the Original Code (or portions thereof)"[59] in the case of the Initial Developer Grant, and to "1) Modifications . . . (or portions thereof); and 2) the combination of Modifications . . . with a Contributor Version (or portions of such combination)"[60] in the case of the Contributor Grant. Thus the patent and trademarks grant is much more restrictive in nature than the copyright grant.

Barring the GPL licenses, most open-source licenses explicitly grant the have made right in their patent grants.[61] A have made right essentially allows the licensee to have a

---

50. Gomulkiewicz, *supra* note 5, at 85 n.74.
51. *See* Rosen, *supra* note 8, at 225.
52. Regents of the Univ. of Cal., *The BSD 3-Clause License*, Open Source Initiative, http://opensource.org/ licenses/BSD-3-Clause (last visited Apr. 25, 2011) [hereinafter BSD License].
53. *Id*. ¶ 2; GPLv2, *supra* note 9, at pmbl.
54. GPLv3, *supra* note 24, at § 2.
55. *Id*.
56. *Id*. at § 11.
57. MPLv1.1, *supra* note 6, at § 2.1(a).
58. *Id*. at § 2.2(a).
59. *Id*. at § 2.1(b).
60. *Id*. at § 2.2(b).
61. OSLv3, *supra* note 18, at § 2; MPLv1.1, *supra* note 6, at § 2.1(b), § 2.2(b); Apache License, v2.0, *supra* note 19, at § 3; *Artistic License 2.0*, Perl Found. § 13, http://www.perlfoundation.org/artistic_license_2_0 (last updated 2006) [hereinafter Artistic License v2.0].

third party manufacture a licensed product for it without the third party or the licensee being subject to an infringement action. Though GPLv3 does not grant a have made right in its patent grant, it addresses this situation by explicitly allowing the licensee to "convey covered works to others for the sole purpose of having them make modifications exclusively for [the Licensee], or provide [the Licensee] with facilities for running those works."[62] Given the nature of open-source licensing, it is desirable to elucidate how the open-source license will be applicable to the third party where the have made right is granted. Not doing so allows leeway to both the licensor and the licensee to attempt to circumvent the provisions of the license. GPLv3, when it allows conveyance of covered works to third parties for modification purposes, also limits the third parties' rights and lays out the rules governing the scenario.[63]

3. Trademarks Grant

Trademarks acquire special importance as regards open-source software. The source code being available for copying, it is essentially the trademarks of a particular entity that establish its goodwill in the market. Users and developers alike tend to gravitate towards products that have acquired a certain reputation in software circles e.g. Linux, Mozilla, Red Hat, etc. That is why most open-source software developers guard their respective trademarks. However, several open-source software licenses do not contain any trademark grant at all or prohibit only specific behaviors with extremely restricted and controlled grants. In *Planetary Motion, Inc., v. Techsplosion, Inc.*,[64] a U.S. case, the court made the following observation as regards GPLv2:

> That the Software had been distributed pursuant to a GNU General Public License does not defeat trademark ownership, nor does this in any way compel a finding that [the licensor] abandoned his rights in trademark. Appellants misconstrue the function of a GNU General Public License. Software distributed pursuant to such a license is not necessarily ceded to the public domain and the licensor purports to retain ownership rights, which may or may not include rights to a mark.[65]

GPLv2 does not refer to trademarks at all. In fact, it explicitly states that "[a]ctivities other than copying, distribution and modification are not covered by this License; they are outside its scope."[66] GPLv3 on the other hand expressly allows inclusion of Additional Terms "[d]eclining to grant rights under trademark law for use of some trade names,

---

62. GPLv3, *supra* note 24, at § 2.
63. *See id.*
64. 261 F.3d 1188 (11th Cir. 2001) (It is worth noting that the Federal Reporter 3d.'s drafters misspelled "Techsplosion" as "Techplosion" in writing the case citation in the actual reporter pages. This has resulted in the case being cited with two different spellings.)
65. *Id.* at 1198.
66. GPLv2, *supra* note 9, at § 0.

trademarks, or service marks."[67] This marks the evolution of the GPL into a holistic intellectual property license.

The Apache License, v2.0 explicitly prohibits use of "trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work."[68] Furthermore, the license requires that the licensee "retain, in the Source form of any Derivative Works that [the licensee] distribute[s], all copyright, patent, trademark, and attribution notices from the Source form of the Work."[69] The Artistic License, v2.0[70] and OSLv3.0 too carry similar clauses.[71]

Likewise, the MPLv1.1 excludes trademark rights from its intellectual property grant. Furthermore, it requires a modified version of the license not to have any reference to Mozilla, Netscape, or other confusingly similar phrases.[72]

The BSD License specifically mandates that neither the name of the licensor nor the contributors "be used to endorse or promote products derived from this software without specific prior written permission."[73]

B. Other Aspects of the Granting Clause

1. Character

The nature of the rights determine the character of the license. The open-source philosophy necessitates a license which is non-exclusive in nature. Accordingly most licenses mention it to be so. However, certain licenses like the BSD License, GPLv2, and the Artistic License v2.0 are silent on this point. Others are silent as regards some of the intellectual property involved. For example, GPLv3 only mentions the non-exclusive character as regards the patent grant but is silent about the copyright grant. The basic assumption in cases where nothing is specified as to the character of the license is that they are non-exclusive.[74]

2. Sublicensing

Management of incremental innovation in open-source software mandates proper transfer of rights to ensure an effective chain of title. Open-source licenses primarily adopt two

---

67. GPLv3, *supra* note 24, at § 7(e).
68. Apache License, v2.0, *supra* note 19, at § 6.
69. *Id.* at § 4.
70. Artistic License v2.0, *supra* note 61, at § 12.
71. OSLv3.0, *supra* note 18, at § 4.
72. MPLv1.1, *supra* note 6, at § 6.3.
73. BSD License, *supra* note 52.
74. Kenneth L. Port, *et. al. Licensing Intellectual Property in the Information Age*, 293 (Carolina Academic Press, 2005 2d Edn.).

modes of doing so: either through sublicensing or through direct allocation of rights from the original author to each licensee. The latter mode allows centralization and hence greater control over the project relative to sublicensing. In the case of sublicensing, a sub-licensor is allowed to grant only the rights received, nothing more, and ensures the passage of obligations too. Regarding sublicensing, Milgrim and Bensen state:

> The current state of law suggests that a patent licensor that does not grant a right to sublicense, or confers only a limited right to sublicense (for example to defined entities, such as subsidiaries and other affiliates), thereby precludes the nonexclusive licensee from further licensing except to the extent contractually permitted.[75]

While the BSD License is completely silent as regards sublicensing, OSLv3.0 expressly allows sublicensing of both copyrights and patents.[76]

GPLv2 allows sublicensing subject to license conditions.[77] However, the procedure and parameters of sublicensing are unclear. To add to the confusion, GPLv2 also grants a license to each and every licensee directly from the original licensor, hence making the need for sublicensing unnecessary.[78] GPLv3 attempts to address this issue by prohibiting sublicensing entirely.[79] Concurrently, GPLv3 has introduced a new section 10, which provides for automatic licensing of downstream recipients, thus rendering the requirement of sublicensing unnecessary.[80]

Conversely, the Apache License, v2.0, while allowing sublicensing of copyrights,[81] does not expressly allow sublicensing of patents, though it does allow the licensee to "otherwise transfer the Work"[82] MPLv1.1 too treads a similar path.[83]

3. Territorial Extent

Theoretically, a licensor can license rights only as to the territory where it has valid and subsisting intellectual property rights. A certain degree of intellectual property harmonization, more so for copyright than for patents and trademarks however, allows for international licensing. Furthermore, open-source software licenses, being based primarily on copyright law, are likely intended to benefit from the automatic protection route of

---

75. 2 Milgrim & Bensen, *supra* note 1, at § 15.13.
76. OSLv3.0, *supra* note 18, at §§ 1-2.
77. GPLv2, *supra* note 9, at § 4.
78. *Id*. at § 6.
79. GPLv3, *supra* note 24, at § 2.
80. *Id*. at § 10.
81. Apache License, v2.0, *supra* note 19, at § 2.
82. *Id*. at § 3.
83. MPLv1.1, *supra* note 6, at §§ 2.1-.2.

copyright law which requires no registration for IP rights to be protected.[84] Moreover, it is prudent licensing practice to attempt to seek the broadest possible territorial grant.

Consequently, several open-source licenses mention the extent of the license as worldwide.[85] GPLv2 and the BSD License on the other hand are completely silent about the territoriality aspect, leaving to assumption that considering their propagating philosophies they intend the grant to be worldwide. The same assumption applies to the copyright grant of GPLv3, though for the patent grant, GPLv3 does mention the extent as "worldwide."[86] GPLv2, however, does allow for a territorial distribution limitation to be included in the license excluding countries where "the distribution and/or use of the Program is restricted . . . either by patents or by copyrighted interfaces."[87]

II. EXCLUSIONS FROM THE GRANTING CLAUSE

It is advisable that a software license specifically mention the rights excluded from the granting clause. Doing so avoids uncertainty regarding implied rights and obligations. However, several prominent open-source licenses are lacking in a specific exclusion clause, thus leaving the exclusions to implication or spread intermittently throughout the license. The most prominent exclusion from most open-source software licenses, as mentioned earlier, is that of trademark rights. The MPLv1.1 contains certain patent exclusions both from the Initial Developer Grant[88] and the Contributor Grant.[89] OSLv3.0 has a specific exclusion clause addressing exclusions covering Licensor's and Contributor's names, trademarks, copyrights, patents, trade secrets and any other intellectual property.[90]

III. TERM, REVOCATON, AND TERMINATION OF GRANT OF RIGHTS

The clause defining the term of a license is usually a part of the granting clause while the termination clause is entirely different. However, the term of a license is closely related to the termination aspect and has therefore been grouped herein for analysis purposes.

A. Term

Good licensing practice mandates that the term of the license be specified in the Grant of Rights. As Milgrim and Bensen state:

> While in practice the term of the license grant is often equivalent to the term of the license agreement, this is not always the case. For example, in

---

84. 1-4 MELVILLE B. NIMMER AND DAVID NIMMER, NIMMER ON COPYRIGHT § 4.08 (Matthew Bender, Rev. Ed. 2009).
85. *E.g.*, *id*.; Apache License, v2.0, *supra* note 19, at §§ 2-3.
86. GPLv3, *supra* note 24, at § 11.
87. GPLv2, *supra* note 9, at § 8.
88. MPLv1.1, *supra* note 6, at § 2.1(d).
89. *Id*. at § 2.2(d).
90. OSLv3.0, *supra* note 18, at § 4.

many licenses the parties expressly or impliedly provide that certain rights and duties survive the term of the actual license. A common way of dealing with this distinction between the term of the license for the intellectual property and the term of the agreement under which the rights and duties are created is to make the term of the license specific, either for a defined period or until a certain event occurs (e.g., expiration of a patent).[91]

The BSD License, MPLv1.1, and GPLv2 fail to specify a license term. Rosen is of the belief that this implies that the GPLv2 is perpetual.[92] The general perception under U.S. law is that perpetual licenses or licenses which do not state any term are licensed for the duration of the intellectual property involved.[93] However, in the U.S., there have been decisions holding that a license that states no term makes the license one at will, and thus subject to termination by either party on notice.[94] The court has also concluded that a license not stating any term had an implied term of thirty-five years.[95] This holding was based on a statutory right in the U.S. Copyright Act granting authors a five-year window from the thirty-fifth year onwards, to terminate a license agreement regardless of the license terms or state contract law.[96] "At least in theory, the potential ability to terminate at will increases the risk of opportunistic behavior by rights holders."[97] Alternatively, it could open the license to a copyright misuse action. This ambiguity has been rectified in GPLv3, which specifies the license term as the term of copyright,[98] but again does not mention a term for the patent grant. The OSLv3.0[99] and the Apache License, v2.0[100] both specify the terms of their licenses explicitly.

B. Revocation of License

The conditions for revocation of license, the method of revocation of license, and recourse available to the licensee on revocation are certain parameters that should be addressed in the license agreement.

The GPLv2 does not state any explicit position on revocation. GPLv3, responding to the internationalization challenge, specifically states that "[a]ll rights granted under this License

---

91. 3 MILGRIM & BENSEN, *supra* note 1, at § 27.02.

92. ROSEN, *supra* note 8, at 127.

93. *See, e.g.*, Poles, Inc. v. Estate of Beecker, 461 F. Supp. 878 (E.D. Pa. 1978); TV Globo Ltda. v. Braz. Up-Date Weekly, Inc., No. 98 Civ. 3911 (RPP), 1999 WL 163378 (S.D.N.Y. 1999).

94. *See, e.g.*, Korman v. HBC Fla., Inc., 182 F.3d 1291 (11th Cir. 1999); Walthal v. Rusk, 172 F.3d 481 (7th Cir. 1999).

95. *See, Rano* v. *Sipa Press, Inc.*, 987 F.2d 580 (9th Cir. 1993).

96. *Id*.; *see* The Copyright Act 2011, 17 U.S.C. § 203 (2011).

97. David McGowan, *Legal Implications of Open-Source Software*, 2001 U. ILL. L. REV. 241, 299.

98. GPLv3, *supra* note 24, at § 2.

99. OSLv3.0, *supra* note 18, at §§ 1-2.

100. Apache License, v2.0, *supra* note 19, at §§ 2-3.

are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met."[101] However, it does not mention irrevocability in its patent license. The Apache License, v2.0[102] uses the term "irrevocable" in both its patent and copyright grant.

Section 7 in GPLv2 and section 12 in GPLv3 can be interpreted to be implied references to revocation. These sections essentially revoke the right of distribution, but not the other rights granted under the license, if due to conditions imposed by third parties, the licensee is not able to simultaneously satisfy the obligations imposed upon him by the license.[103] However, if there is an option of simultaneously satisfying both, or any other cure, then the right remains or is reinstated.[104]

Stating the parameters governing revocation creates mutual expectations and reduces the risk of unanticipated revocation. However, a license essentially amounts to a "revocable permission."[105] The same premise has been recognized for software licenses, where "nonexclusive licenses are revocable absent consideration."[106] Some experts are also of the view that "merely stating that the license is irrevocable does not make it so,"[107] and that "promissory estoppel could be used to bridge this gap, via a failed contract theory."[108]

C. Termination Issues

A termination clause should address in detail the events that would trigger termination, the consequences of termination, and what rights or obligations would continue post-termination.

U.S. copyright law permits copyright termination under certain conditions; it allows authors "to terminate grants of copyright assignments and licenses that were made on or after January 1, 1978."[109] "[T]he provision was a safeguard 'against unremunerative transfers.'"[110] "The copyright termination provision is an unequivocal right of reversion

---

101. GPLv3, *supra* note 24, at § 2.
102. Apache License, v2.0, *supra* note 19, at §§ 2-3.
103. GPLv2, *supra* note 9, at § 7; GPLv3, *supra* note 24, at § 12.
104. *Id.*
105. *See* 25 Am. Jur. 2d *Easements and Licenses* § 122 (2011); 53 C.J.S. *Licenses* § 143 (2011).
106. Sapna Kumar, *Enforcing the GNU GPL*, 2006 U. Ill. J.L. Tech. & Pol'y 1, 13 (quoting 3 Melville B. Nimmer, Nimmer on Copyright § 10.02(B)(5) (2000)) (internal quotation marks omitted); *see also* Carson v. Dynegy, Inc., 344 F.3d 446 (5th Cir. 2003); Keane Dealer Servs., Inc. v. Harts, 968 F. Supp. 944 (S.D.N.Y. 1997); Johnson v. Jones, 885 F. Supp. 1008 (E.D. Mich. 1995); 53 C.J.S. *Licenses* § 146 (2011).
107. Kumar, *supra* note 107, at 14.
108. *Id.*
109. *Termination of Transfers and Licenses Under 17 U.S.C. §203*, U.S. Copyright Off., http://www.copyright.gov/docs/203.html (last visited Apr. 25, 2011); *see* 17 U.S.C. § 203 (2006).
110. Grant C. Yang, *The Continuing Debate of Software Patents and the Open Source Movement*, 13 Tex. Intell. Prop. L.J. 171, 201 (2005) (quoting H.R. Rep. No. 94-1476, at 124 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5740).

which can neither be waived nor contracted away; thus no form of open-source license would be able to protect against it."[111] Open-source projects are primarily dependent on copyright law, and the existence of such a termination provision could prove to be their Achilles' heels. Even if the copyright termination was exercised on a small amount of source code, it could still deal a tremendous blow to development of the software of which it was a part. Lydia Pallas Loren argues against the applicability of such termination provisions in copyright law to open-source licenses:

> While it is clear that the termination provisions were not designed to address this type of licensing regime, it might be tempting for a court to construe these licenses as grants subject to termination under the statute. Such a construction would undermine the reliability of the semicommons status of a work . . . . [C]ourts should understand these types of licenses as a type of limited abandonment not subject to the termination provision.[112]

Open-source licenses also contain a termination clause contingent on certain conditions. The GPLv2 requires strict adherence to the terms of the license, failing which would result in termination of the license.[113] OSLv3.0[114] and the MPLv1.1[115] too have similar clauses. The MPLv1.1 allows an opportunity to cure the breach.[116] Furthermore, the MPLv1.1 termination clause is considerably broader in scope than other licenses. Firstly, it is triggered by any act of noncompliance with any of the terms. Other licenses' termination clauses are usually triggered in response to certain actions. Secondly, the MPLv1.1 provides for the termination of the complete license, unlike some other licenses which provide for termination of certain rights. Compared to GPLv2, GPLv3 elucidates much more on the termination aspect. The scope of the termination clause is much more diverse, encompassing the patent portfolio too.[117] GPLv3 also makes provision for reinstating the license both temporarily and permanently, hence providing solace to inadvertent violators. However, rights not permanently reinstated cannot yield a license for the same material.

The GPLv2 provides protection to downstream users even if an upstream user's license is revoked.[118] The MPLv1.1 too has a similar but more expansive clause,[119] as does GPLv3.[120]

---

111. *Id*. (footnote omitted).
112. Lydia Pallas Loren, *Building a Reliable Semicommons of Creative Works: Enforcement of Creative Commons Licenses and Limited Abandonment of Copyright*, 14 GEO. MASON L. REV. 271 (2007).
113. GPLv2, *supra* note 9, at § 4.
114. OSLv3.0, *supra* note 18, at § 9.
115. MPLv1.1, *supra* note 6, at § 8.1.
116. *Id*.
117. GPLv3, *supra* note 24, at § 8.
118. GPLv2, *supra* note 9, at § 4.
119. MPLv1.1, *supra* note 6, at § 8.1.
120. GPLv3, *supra* note 24, at § 8.
121. *Id*. at § 10.

A large number of open-source developers lack resources to negotiate or face patent litigation. Hence, the licenses include defensive strategies; for example, an important implication of the GPLv3 termination clause is termination of the license if a licensee "initiate[s] litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed."[121] The threat of withdrawal of intellectual property rights in the program and ouster from that particular open-source development model is used as a suitable deterrent against initiation of infringement action. However, this is a working option; it would be a restraint only if the aggressive party has deep interests at stake in the software.

Though GPLv2 lacks such a provision, similar provisions exist in other open-source licenses.[122] The OSLv3.0 termination provision does not, however, "apply for an action alleging patent infringement by combinations of the Original Work with other software or hardware."[123] The MPLv1.1 is quite elaborate in its termination clause. The MPLv1.1 excludes declaratory judgment actions from applicability of the termination clause. The MPLv1.1 terminates *prospectively* all copyright and patent licenses granted by the Initial Developer or a Contributor (Participant) for someone suing the Participant for infringement.[124] On the other hand, it terminates *retrospectively* only the patent licenses granted by the Participant for someone suing for infringement by any software, hardware, or device, other than such Participant's Contributor Version.[125] The MPLv1.1 also makes provision for a 60 days notice and opportunity for the parties to reach a settlement in the former scenario by payment of royalty or withdrawal of the suit.[126]

IV. THE GRANTING CLAUSE AND INTELLECTUAL PROPERTY LAW ISSUESS

Software licensing is complex, as the tangible cannot be separated from the intangible. It encompasses licensing of the underlying technology at the primary level, then the licensing of the complete program at a secondary level. Furthermore, the program requires a physical medium of transfer and a physical medium in which to function.

A. First Sale Doctrine and Principle of Exhaustion Under Copyright Law

1. First Sale Doctrine

Copyright law, via the first sale doctrine, allows the purchaser of an authorized copy certain limited rights as to that particular copy, primarily the right to distribute, sell, or otherwise dispose of that copy, without the permission of the copyright owner.[127] However,

---

121. *See, e.g.*, The Apache License, v2.0, *supra* note 19, at § 3; OSLv3.0, *supra* note 18, at § 10.
123. OSLv3.0, *supra* note 18, at § 10.
124. MPLv1.1, *supra* note 6, at § 8.2(a).
125. *Id*. at § 8.2(b).
126. *Id*. at §§ 8.2(a), 8.3.
127. *See* 17 U.S.C. § 109(a) (2011).

the purchaser of a copy of a copyright protected work cannot make and distribute derivative works of the copy.[128]

Such an exception would be devastating for the proprietary software industry. Hence, software developers resort to licensing instead of selling their software, thus circumventing the first sale doctrine.

However, there are certain chinks in the armor. The Free Software Foundation asserts that the GPLv2 is a license, yet concurrently it expressly allows sale of copies under the GPL.[129] This could bring into effect the first sale doctrine. Though it would not allow creation of derivative works, it would allow circumvention of open-source licenses, bringing the software under the general application of copyright laws. Thus, an avenue becomes available to avoid the copyleft-reciprocity provision. Hence, the possibility exists of a developer preparing add-on software, or utilities or upgrades, while avoiding triggering the derivative works right, and thus being in conformity with copyright law. With non-applicability of the copyleft-reciprocity provision, there would be no compulsion to contribute to the open-source project.

The dithering judicial stance further complements the possibility of such an occurrence. In some cases, courts have disregarded the "license" label to find that the transaction was a sale. In the U.S., in *Vernor v. Autodesk, Inc.*,[130] the plaintiff obtained legitimate copies of Autodesk, Inc.'s AutoCAD software package from a third party. Autodesk, Inc. had transferred the software to the third party pursuant to a license, which recited a "nonexclusive, nontransferable license to use" and prohibited "rent, lease, or transfer [of] all or part of the Software."[131] The court concluded—based on the fact that the third party was allowed "to retain possession of the software copies in exchange for a single up-front payment" and that "the Settlement Agreement and License imposed onerous restrictions on transfer of the AutoCAD copies"[132]—that the transaction between Autodesk and the third party was a "sale with restrictions on use," which fell under the first sale doctrine.[133]

2. Principle of Exhaustion

Similar to the U.S. first sale doctrine, albeit more extensive in ambit, is the principle of exhaustion in European copyright law. This section will focus on the principle's application in Germany. According to this statutorily embodied principle, the distribution right is exhausted whenever a copy of a computer program is put into circulation by way of sale, or

---

128. NIMMER, *supra* note 84, at § 8.08.
129. *See* Free Software Found., *supra* note 25.
130. 555 F. Supp. 2d 1164 (W.D. Wash. 2008).
131. *Id.* at 1166 (alteration in original).
132. *Id.* at 1170.
133. *Id.* at 1170-71; *see also* SoftMan Prods. Co. v. Adobe Sys. Inc., 171 F. Supp. 2d 1075 (C.D. Cal. 2001).

otherwise brought into traffic, in the European Union or the European Economic Area with the consent of the copyright owner.[134]

As in the U.S., an issue is raised regarding the enforceability of the license on third parties who are not directly bound by the terms of the license agreement. Moreover, there have been instances where German courts have ignored, not only labels of license and sale, but most of the content of the software licenses, and have held perpetual software transfers for fixed fees as sales.[135]

A question under German copyright law exists as to whether exhaustion applies to online distribution of software instead of only when software is distributed on data storage media. This issue arises because a possible interpretation of section 69c(3) of the Copyright Act is that for exhaustion to apply, the rights holder should itself put a copy of the computer program into physical distribution. The argument against applying the doctrine to online distribution is that it is not the rights holder but the recipient who creates a copy when he downloads the software.[136] This view, of course, has a lot of dissenters.[137]

Another issue with deep implications for open-source software is that entailed in section 69d(1) of the Copyright Act, which allows users of a computer program rights of reproduction and alteration without obtaining a license, if the acts are necessary for the use of the program in accordance with its intended purpose.[138] Ambiguity in interpretation could arise due to the initial language of the clause which allows exercising of the rights subject to the condition that no specific contractual provision stipulates otherwise.[139]

These issues have already been raised in the context of open-source software. In *Welte v. D-Link Deutschland GmbH*,[140] a German case, the defendant asserted that the conditions of GPL did not apply because of the principle of exhaustion. The District Court of Frankfurt am Main held that:

> Defendant cannot invoke a claim of exhaustion of the right to distribute (Section 69 c, No. 3, Sentence 2 of the German Copyright Act (UrhG)), even though the three programs are available to the public on the Internet. The principle of exhaustion only applies to the individual physical data carrier

---

134. Gesetz über Urheberrecht und verwandte Schutzrechte [Urheberrechtsgesetz] [UrhG] [Copyright Act], Sept. 9, 1965, BUNDESGESETZBLATT, Teil I [BGBL. I] at 1273, as amended, § 69c(3) (Ger.).

135. Lothar Determann, *Dangerous Liaisons—Software Combinations as Derivative Works? Distribution, Installation, and Execution of Linked Programs Under Copyright Law, Commercial Licenses, and the GPL*, 21 BERKELEY TECH. L.J. 1421, 1468-69 (2006).

136. *See* Tim Engelhardt & Till Jaeger, *Germany*, INT'L FREE & OPEN SOURCE SOFTWARE L. BOOK n.14, http://ifosslawbook.org/germany/ (last updated 2011) (collecting cases in support).

137. *Id.* at n.15.

138. Urheberrechtsgesetz [Copyright Act], Sept. 9, 1965, BGBL. I at 1273, as amended, § 69d(1) (Ger.).

139. *Id.*

140. Landgericht Frankfurt Am Main [LG] [District Court of Frankfurt] Sept. 6, 2006, No. 2-6 O 224/06 (Ger.), *available at* http://www.jbb.de/judgment_dc_frankfurt_gpl.pdf.

onto which the software is copied during the downloading process. With respect to the right to copy, no exhaustion takes place; thus, Defendant is not entitled pursuant to Section 69 c, No. 3, Sentence 2 of the German Copyright Act (UrhG) to [freely] copy the software onto the individual data storage units.[141]

B. Exhaustion Doctrine Under Patent Law

The exhaustion doctrine, the patent law analog to copyright law's first sale doctrine, lays down that an unconditional sale of a patented device embodying the patentee's invention extinguishes the patentee's right to control the purchaser's use of the device. The patentee conveys a property interest in the device along with the right to use. Thus, the purchaser can transfer the device without the patentee's consent without exposing either himself or the purchaser to liability for infringement.[142]

Again, as under the first sale doctrine, the limitations on open-source software of the exhaustion doctrine are largely negated. However, possibilities always abound. One possibility is where, as in the first sale doctrine, the court interprets the license as a sale and similar consequences as under the first sale doctrine ensue for open-source software.

Another possibility is where, subsequent to *Quanta Computer, Inc. v. LG Electronics, Inc.*,[143] a U.S. decision, it is possible to combine a licensed product with a non-licensed product and cite patent exhaustion.

C. Implied License

Major software corporate houses essentially employ several techniques to contribute to the open-source model. Firstly there is the option of pledging patents to the patent commons, where software companies commit not to enforce their patents against open-source products.[144] Secondly there is the option of licensing patents subject to the open-source license.

The scope of possible conspiracy theories is immense. Firstly, the enforceability of the pledges to the patent commons is suspect.[145] Secondly, open-source developers may rely on an open-source license only to be informed of infringement. Thirdly, certain contributors to

---

141. *Id.* at 6-7 (alteration in original) (unofficial translation).
142. *See* Quanta Computer, Inc. v. LG Elecs., Inc., 553 U.S. 617 (2008).
143. *Id.*
144. *See* Press Release, IBM, IBM Pledges 500 U.S. Patents to Open Source in Support of Innovation and Open Standards (Jan. 11, 2005), http://www-03.ibm.com/press/us/en/pressrelease/7473.wss; Press Release, Open Invention Network, Open Invention Network Formed to Promote Linux and Spur Innovation Globally Through Access to Key Patents (Nov. 10, 2005), http://www.openinventionnetwork.com/press_release11_05.php; *Red Hat, Inc. Statement of Position and Our Promise on Software Patents*, Red Hat, http://www.redhat.com/legal/patent_policy.html (last visited Apr. 25, 2011).
145. *See* Joe Mutschelknaus, Note, *Spillover Effect: Investigating Patent Implications to Open-Source Software Copyright Licensing*, 19 Fed. Cir. B.J. 409, 412, 429-31 (2010).

the patent commons allow restrictive patent usage, only under particular open-source licenses.[146] Thus, if another open-source software developer of an incompatible open-source license was to proceed, basing his stand on those licenses, he would be open to an infringement action.

The doctrine of implied license poses an interesting intellectual property complication. Primarily, the doctrine might arise equitably or legally.

The various pledges made to the patents commons reflect an affirmative grant of consent to open-source software developers and users. That implied licenses were not outside the contemplation of the drafters is clear, as they find explicit mention in open-source licenses.[147] It has often been argued that the Preamble and section 7 of GPLv2 contain an implied license.[148] Hence, an argument subsists that the circumstances in toto establish consent constituting an equitable implied license.

Legally, an implied license arises by way of sale. "That purchaser may be deemed to have an 'implied license' under the patent or copyright with respect to the use and sale of an article that purchaser brought from an authorized person."[149] The rights holder cannot license or assign a right, receive consideration, and then seek to derogate from the right granted.[150]

Open-source licenses have already been adjudged to be enforceable contracts with valuable consideration.[151] Hence, it can be argued that a rights holder under an open-source license grants an implied license to its patents, even if it is not mentioned as such in the license. Joe Mutschelknaus believes it to be so when he affirmatively argues that "the question is whether the copyright grant in an open-source license 'carries with it, by necessary implication, a license under any . . . patent of the licensor which would be infringed by operation under the grant'?"[152]

A major implication of the implied license would be in the field of trademarks. As mentioned earlier, there are several open-source software licenses that do not contain a trademark grant at all, or prohibit only specific behaviors with no or extremely restricted and controlled grants. In such a situation, the court could interpret the grant of an implied license.[153]

---

146. *See* Bruce Perens, *The Open-Source Patent Conundrum*, CNET NEWS (Jan. 31, 2005), http://news.cnet.com/The-open-source-patent-conundrum/2010-1071_3-5557340.html.

147. *See, e.g.*, GPLv3, *supra* note 24, at § 11.

148. *See* Asay, s*upra* note 43, at 288; *see also* Rosen, *supra* note 8, at 126; Ilardi, *supra* note 43, at 295.

149. 2 MILGRIM & BENSEN, *supra* note 1, at § 15.48.

150. *See generally* Mutschelknaus, *supra* note 146, at 424 citing Winbond Elecs. Corp. v. Int'l Trade Comm'n, 262 F.3d 1363, 1375 (Fed. Cir. 2001).

151. *See* Jacobsen v. Katzer, 535 F.3d 1373 (Fed. Cir. 2008).

152. Mutschelknaus, *supra* note 146, at 424 (alteration in original) (quoting AMP Inc. v. United States, 389 F.2d 448, 453 (Ct. Cl. 1968)).

153. *See* McCoy v. Mitsuboshi Cutlery, Inc., 67 F.3d 917 (Fed. Cir. 1995); Dep't of Parks & Recreation v. Bazaar Del Mundo Inc., 448 F.3d 1118 (9th Cir. 2006) (holding that it is the entire course of conduct of the involved parties that determines whether an implied license exists).

V. Conclusion

The institution of open-source has its roots in an ethical rebellion of sorts by software programmers, expressed in the form of sharing source code of computer programs. The cause was subsequently taken up by academics and practitioners alike, and eventually this model snowballed into a parallel regime in the software development landscape. Now, open-source has gained prominence as an institution that earmarks a philosophy—a social movement that developed new facets of software development and its commercialization. The popularity of the movement has invited critiques from all involved stakeholders, viz. economists, lawyers, engineers, and social commentators.

The legal tool to this ethical and technological rebellion is the license. However, unlike the technological aspect, where trained technical persons are involved, in the complementary legal facet, the involvement of trained legal personnel is comparatively reduced. Several open-source licenses were drafted by persons unaware or limitedly aware of legal intricacies. Hence, drafting irregularities, inconsistencies, and ambiguous terminology and usages exist, which pave the way for an ambiguous assumption as to the scope of the license. With time, open-source software has evolved and the legal facet too has been properly structured. Hence, newer versions of licenses are more in tandem with the law.

Software development is a creative process involving numerous conscious choices by the programmer based on user feedback. The open-source model, as opposed to the proprietary model, ensures the optimal mode of feedback and ensures tailoring of software to the specific needs of the society. The major emphasis of the granting and associated clauses is to ensure that such a feedback model is maintained and not corrupted. ■