

Italy

author:[Piana,Carlo] author:[Aliprandi,Simone]

Body of law

Copyright protection of software is regulated in Italy under a few articles added to the general Italian Copyright Law (precisely “Legge n. 633 del 22 aprile 1941”) under the Legislative Degree no. 518 dated 29 December 1992. This reform transposes the Council Directive of 14 May 1991 on the Legal Protection of Computer Programs (91/250/EEC) into Italian national law (hereinafter referred to as the “Software Directive”).

Now the Italian Copyright Law has a section in chapter IV specifically on software: namely, Section VI, comprising articles 64 bis, 64 ter, and 64 quarter.

Software Act: Object of protection

Computer programs (including the preparatory material) are protected by copyright and are equivalent to literary works within the meaning of the Berne Convention for the Protection of Literary and Artistic Works. According to article 2, no. 8, of the Italian Copyright Law software programs are protected in whatsoever form, insofar as they are original and result from the intellectual creative activity of the author. The ideas and principles on which software programs are based, including those on which their interfaces are based, are excluded from protection. The concept of “program” includes also the preparatory material of the same.

Authors/Beneficiaries

The general principle of Italian copyright law provides that the copyrights in a work belong to the author. Article 64 ter however provides that — unless stipulated otherwise — the employer holds the exclusive right to exploit the program (or of the data base) created by the employee while performing his/her job or working under instructions from the employer. According to a very reliable doctrine, however, also in case of “work for hire”, i.e. when software development occurs under the performance of a development agreement and is paid by the client, the exploitation rights are assigned to the contracting party.¹

Exclusive rights

According to article 64 bis of the Italian Copyright Law the exploitation rights comprise the exclusive right to perform or authorize:

¹L. C. UBERTAZZI, ‘Diritto d’autore, estratto da Commentario breve alle leggi sulla Proprietà Intellettuale e Concorrenza, 4th Edition, CEDAM, Milano, 2007, pp. 62-63

(a) the temporary or permanent reproduction of the computer program by any means or in any form. Insofar as acts like the uploading, displaying, execution, transmission or storage of a software program require its reproduction, those acts are also subject to the authorization of the copyright holder; (b) the translation, adaptation, transformation and any other modification of the computer program, including the reproduction of the resulting program [in other words, creating a derivative work], without prejudice to the author of the modification; (c) any form of public distribution, including lending the original computer program or copies thereof. The first sale within the [the European Union] exhausts the right to further control the distribution of such copy within the [European Union], with the exception of the right to control the further lending of the program or of a copy thereof.

Exceptions to exclusive rights

Article 64 ter of the law expressly provides that:

in the absence of any contrary stipulation, authorization from the copyright holder shall not be required for the activities as per sub-headings a) and b) of article 64 bis, whenever such activities are necessary for the program to be used, for its designated purpose, by the person who legally purchased it, including for the correction of errors. Whoever is entitled to use a copy of the computer program may not be prevented by contractual means from making a backup copy of that program, where this is necessary for its use. Whoever is entitled to use a copy of the computer program may, without being authorised by the copyright holder, evaluate, study or test the operation of such program in order to identify the ideas and principles underlying each component of that program, provided he carries out such acts during the course of loading, visualization, execution, transmission or storage of the program which he/she is entitled to perform. Any contractual stipulation conflicting with the provisions of this paragraph is null and void.

Moral rights

The Italian copyright system is a *droit d'auteur* one, where moral rights are particularly strong. Worth noting are:

- The right to be acknowledged as the author of the work (Art. 20, 21)
- The right to object to modifications or alterations which are prejudicial to the honour or reputation of the author (Art. 20)
- The right to withdraw the work from distribution, when high moral reasons exist, subject to indemnification of those who have acquired the right

to exploit the work (Art. 164).

Article 20 provides for two separate rights:

independently of the exclusive rights of economic utilization of the work referred to in the provisions of the preceding Section, and even after the transfer of such rights, the author shall retain the right to *claim authorship* of the work and to *object* to any distortion, mutilation or any other modification of, and other derogatory action in relation to, the said work, which would be prejudicial to his honour or reputation.

While the right to claim authorship does not raise any practical concern, the right to object to any distortion has been cited sometimes as a possible restriction to the working of a FOSS license, because the author at any point in time could *de facto* revoke his or her permission to modify the program, in stark contradiction to the provisions of any FOSS license. However, on the one hand this right is limited to very serious alterations of the work, as they must be detrimental to the honor or reputation of the author. In addition, according to Art. 22, if the author is aware of and accepts the modifications he has no right to object to them. Finally, we argue that this provision is not applicable to software because of its rationale, which is to protect the “sprit” of the artist which lives in artistic works, spirit which are far less arguable in a software work. There is no known case in Italy dealing with the application of such rights to software. Many authors are inclined to think that since there is no specific exception for software, and since the rules are of general application, they also apply to software. But this rather formal argument does not seem convincing, as the same is quite patently inapplicable in certain instances of copyright law, such as the *sui generis* database rights. Besides, because the copyright provisions in the Italian copyright law are formally an implementation of a Directive for the harmonization of the internal market, it is arguable that such an impediment to harmonization — that could prejudice the exploitation of software works — may only be based on grounds that supersede the European legislative powers and should receive restrictive interpretation.

Article 22 provides that these moral rights are non-transferable and Art. 23 provides that they have no time limitation, as after the author’s death they may at any point in time be exerted by the heirs of the author.

A further moral right is the right to *withdraw the work* from distribution when high moral reasons exist (Art. 142). This provision has the same rationale as the one in Art. 20, and again it reflects the fact that an author can have very serious moral involvement with his or her own works, so that the distribution of said works can be highly prejudicial to his or her reputation or other aspects of the *droit moral* of the author. It is therefore arguable that the right to withdraw the work does not apply to software for the same reasons we put forward for Art. 20. At any rate, the need to indemnify the rightsholders (including licensees) who would be damaged by the withdrawal, and the cumbersome procedure provided

for it by the law, relegates this hypothesis to the realm of intellectual exercise without practical bearings.

Term of protection

The same term as for works of literature and art applies: 70 years as of January 1 following the death of the author.² In case of co-authorship the death of the latest surviving author is taken into account.

Copyright assignment

Copyright assignment is not expressly regulated by Italian Law for any copyrightable subject, except for works created under an employer/employee relationship, as mentioned above. By the way, this is not considered a case of assignment, but a case where the rights to exploit the work are vested in the employer by virtue of the law. We have also already mentioned that in the case of work for hire the copyright of the resulting work has to be attributed — depending on circumstances and unless different stipulations exist — to the contracting entity, as this is the normal working of the rules of contracts (*“contratto d’appalto”*). In the case of unpaid contributors, without an express agreement the copyrights will be vested in the authors, as the general rules apply.

Moral rights cannot be assigned and any agreement to the contrary is null and void. All other rights can be assigned and transferred without limitation (Art. 107 of Italian Copyright Law), save for certain express provisions under a the publishing agreement. The only requirement is that the agreement must be proved in writing (Art. 110).

Special measures

The Italian Copyright Law refers to the general copyright principles regarding measures for enforcing software copyrights. Besides these general measures, a specific criminal sanction has been created for those who bring into trade or possess for commercial purposes a copy of a computer program. Article 171 bis, subsection 1, of the Italian Copyright Law provides that:

whoever unlawfully duplicates computer programs for purposes of gain, or imports, distributes, sells, detains for commercial purposes or leases the said programs, for the selfsame purposes and being aware or having reason to be aware of the fact that such copies were not authorized, shall be punished by imprisonment for between three months and three years and by a fine of between 500,000 and 6,000,000 Liras [1 euro = 1936.27 Liras]. The same punishment shall be not less than imprisonment for six months and a fine of 1,000,000 Liras if the offence is serious or if the program which was

²Italian Copyright Law, Art. 25 and 32-ter of the Italian Copyright Law

unlawfully duplicated, imported, distributed, sold, detained for commercial purposes or leased had been previously distributed, sold or leased on media bearing the mark of the Italian Authors' and Publishers' Association [SIAE] pursuant to this law and to the relevant Regulations as per Royal Decree no. 1369 of May 18, 1942.

Unprotected software and public domain software

Italian Copyright Law lays down that, in order to be protected, a work must have a minimal creative character, which must have at least two components: novelty and originality. This applies also to software, although as a matter of fact — excluding utterly simple computer programs or very basic scripts, as well as interface definition files — it would be hard for a judge to rule that a software application fails to have sufficient creative character.

A public domain status exists *ex lege* when the copyright expires, i.e., 70 years after the death of the longest surviving author. In practical terms this is so far ahead of it becoming relevant that discussing it would be futile at the time of writing. In theory, nothing prevents a rightholder from relinquishing her rights, but as we have seen, some of the moral rights are actually non-disposable; therefore it is impossible for a copyrighted item in Italy to have a truly public domain status. If for the purpose of our analysis we redefine “public domain” for this purpose into “devoid of any exploitation rights”, then a public domain status is achievable by an act of dedication to the public domain of the rightholder.

Analysis of FOSS under Italian law

Copyrights

Although FOSS can be written by a single person or be owned by a single legal entity,³ after some time software is the result of the work of several authors who can lay claim to it. The question is whether later additions create a collaborative work (a work created by collaborating authors), or whether the original software is the final work and every contribution created during the further development of the software is a derivative work. The legal consequences are different in each case.

Qualification of FOSS

There is no way to determine whether a FOSS work is a collective work, a complex work or else, and each version of the software can be classified differently depending on how the work was made. Many combinations can ensue. The simplest scenario is the one in which the software was made by author A, taken by author B and then by author C. A, B and C have each modified and extended the software. This is a collective effort, but technically it consists of a series

³Such as software developed by employees (Italian Copyright Law, Art. 12 bis), and software developed for hire under a contractual transfer of copyrights

of derivatives, each one being technically copyright of the respective author, who relies on the upstream permission in the chain of development. Usually the scenario is more complex, as the contributions of the individual authors are committed into the release by means of a code management system through which concurrent development is made.

Whereas the first version of the software, if written by several people, can in many cases be qualified as a collaborative work where the different contributions are indistinguishable, this seems much less the case for the later versions, which are based on the original work, without, however, there being any “consultation” between the authors. These later versions will be qualified as derivative works. Therefore, in terms of the legal consequences, a distinction needs to be made between the rights of the original co-authors and the rights of people who carry on developing the original work.

According to a sound theory in Italy, this way of working constitutes a “complex work”, in other words a work in which “[...] the individual contributions have their own autonomy, which enables their separate exploitation, and nevertheless they are shaped in a way that, as a final result of the cooperation, they result as essential elements of an organic *ensemble* where the individual contributions express themselves as a whole, creating an unitary artistic effect”. This definition is geared towards artistic works (such as movies) where the contributions are different in nature (like the photography, the direction, the story, the writing of the screenplay, etc.), while in software programming the elements are more of the same nature.

Rights of the original co-authors

Unless the components of the software can be clearly distinguished *and* separate, software made collaborative is usually considered an “indivisible work”⁴. This involves works whereby it cannot be concluded clearly what the individual contribution of every author is, e.g. when two authors write a text together. In the case of indivisible works the authors are free to regulate the exercise of the copyrights by agreement. This freedom is very far reaching. The co-authors can agree how the program is made public (e.g. as “FOSS”) and how decisions regarding the copyrights are made, e.g. by normal or special majorities, or by giving one of them the right to take all the decisions regarding this work (limited to acts of ordinary administration).⁵

If the co-authors have not reached an agreement as to how decisions are made (insofar as the law allows them to regulate the co-authors’ decision-making process) the rules laid down by Art. 1105-1110 of the Italian Civil Code apply. The main rule is that any act that does not involve disposing of the copyright and that does not prevent the co-owners to exert their rights is allowed, but acts of

⁴“Indivisible works” are governed by art. 10 of the Italian Copyright law, which makes extensive reference to the rules of the “*Comunione*” (shared ownership), Art. 1100 onwards of the Italian Civil Code

⁵Art. 1106 of the Italian Civil Code

“extraordinary administration” must be voted according to the majorities laid down by the law or agreed upon by the parties. Parties disagreeing can oppose the decisions of the majority in Court.⁶

FOSS and derivative works under Italian law

Italian law does not expressly mention “derivative works”, but it deals with “creative elaborations”. It may be only a nuance, a reflection of the nature of the Italian law in the *droit d’auteur* system, but it is worth noting.

Art. 4 of the Italian Copyright Law provides that

Without prejudice to the rights in the original work, elaborations of a creative character of any such work, such as translations into another language, transformations into any other literary or artistic form, modifications and additions constituting a substantial recasting of the original work, adaptations, reductions, abridgements and variations which do not constitute an original work, shall also be protected.

The assignment of copyrights

Assignment of copyrights in FOSS licensing does not pose any particular challenge or differ greatly from the general working concept of assignment. The virtual absence of distributed all-Italian collaborative FOSS projects makes it difficult to predicate any characteristic of the assignment of copyrights in practice, for lack of examples. A field where all-Italian FOSS projects exist is the public administration sector, which is however governed by certain specific rules (such as the reuse principle and compulsory clauses in ad-hoc software development) meaning that it would require a much wider and separate discussion.

Moral rights

Moral rights could play a dangerous role in FOSS, which for instance prohibits of field-of-use restrictions. In discussion groups arguments can be found in favour of applying moral rights in order to object to military or nuclear power-related use of software) and the Freedom to improve and adapt the software, which empowers anybody to make changes to the software. If these change are made to port an application to a field of use that can affect the reputation or the morals of the original author, arguably this could trigger the use of moral rights to object to such modifications of the work. Reference is made to the section on moral rights above above for an argument against the applicability of the moral rights to software.

⁶Article 1109 of the Italian Civil Code

Enforcing FOSS licenses

There is little doubt that a FOSS license is enforceable in Italy, except in marginal cases. The same reasoning as in *Welte v. Skype*⁷ would apply. Regardless of whether the license is a contract or a bare copyright license,⁸ copyright can be considered a general prohibition barring unrestricted use of the software. Therefore other than the license, nothing permits the use of copyright, and the absence of a license would defeat any arguments by an infringer directed at voiding the license.

In other words, in a FOSS license enforcement case the alleged infringer cannot invoke the nullity of the license and at the same time argue that the use was legitimate under the same license, unless there is another legal theory that permits said infringer to use the software. For instance, if the infringer claims that the obligation to release the modified source code is unenforceable because, e.g., the GNU GPL is null, and this is a condition to use the software according to the same GNU GPL, this argument will fall flat on its face, because it would result in claiming that the use by the alleged infringer was not permitted at all.

But what about actually enforcing an obligation? Such as, in the above example, actually enforcing the disclosure of the modified source code if the alleged infringer refused to perform such obligation (or to meet such condition). In theory, a licensee could be forced to do something which is provided for by a contractual obligation. Under Italian law, this is in general possible under Art. 2931 of the Italian Civil Code. To trigger such provision, a contractual obligation must exist, a “positive” one (of doing or giving something), whereas a “condition” is not sufficient. Most FOSS licenses just set out the conditions that must be met in order to permit the four liberties, and also the copyleft clauses are laid down more as true conditions than as obligations. Therefore it is hard, and not necessary, to construe a contractual obligation. There are exceptions, though, to this finding. Two examples come to mind: explicit downstream patent licenses (as in the Mozilla Public License) and liability disclaimers.

Downstream patent licenses state that the contributor (sometimes the distributor, as with the GNU General Public License v.3⁹) of software will explicitly provide all downstream recipient of the software it distributes with a worldwide royalty-free patent license for the patents it owns or controls. This is a positive effect of the license. However, this is strictly speaking not a result of the in-bound license from the upstream, but a result of distribution (or “conveyance”) of software under the same public license, which is a separate act of licensing. In fact, without distribution, the grant is not applicable, and the act of granting itself is conceived not as an obligation to grant, but as a direct grant embedded

⁷Case *Welte vs. Skype Technologies SA* (Germany) reported inter alia by Groklaw (<http://www.groklaw.net/article.php?story=20080508212535665>)

⁸On this we have a precise opinion, see C. PIANA, *Licenze pubbliche di software e contratto*, I Contratti, IPSOA, Milan, 2006, p. 720-727, also available for download at http://www.piana.eu/repository/720_727.pdf

⁹Cfr. GPL v.3, Section 11

in the license. Again, there is no need to construe a valid contractual obligation to make the patent grant work.

Similarly, all licenses provide for a liability disclaimer, which again is not an obligation, but an effect of the license, or a waiver. This is a focal point and needs to be dealt with more in depth.

Waiver and liability

Typically, FOSS licenses contain very strong disclaimer clauses, which discharge the author from all liability.¹⁰ The reason for this is that FOSS is often made available without monetary compensation of any sort, as a result of which the author generates insufficient income to pay for liability insurances and legal costs.¹¹

Under Italian law there is a first issue. According to Art. 1229 of the Italian Civil Code, no disclaimer of liability can be made to the effect of excluding liability for gross negligence or willful acts. Any provisions to the contrary are null and void (although the overall agreement may remain valid). The nullity can be declared *ex officio* without a specific demand from the parties (Art. 1421 ICC), but it must be instrumental to a demand made by the parties. Therefore, the provisions of the licenses are null and void insofar as they unconditionally exclude all liability without making such distinction. However, the invalidity does not extend to the contractual clauses that are not affected by the nullity (art. 1429 ICC) and in any case the clauses that are null and void can be converted into different clauses with similar effect, such as the parties would have wanted if they had been aware of the nullity (Art. 1424 ICC). All those rules should be read in the light of the license quite likely being a unilateral act (Art. 1424 ICC).

Should the disclaimer be ineffective, could a software developer be liable for damages caused by his or her software, under Italian law, in the light of the fact that his or her software is released for free (under the FOSS license)? Apart from the cases of gross liability and wilful acts, or a liability in tort, the answer seems negative. On contractual grounds it is impossible to determine a liability. A license is just that, a permission, therefore it does not impose any obligation to

¹⁰See e.g., the BSD license (<http://www.opensource.org/licenses/bsd-license>): THIS SOFTWARE IS PROVIDED BY <copyright holder> 'AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <copyright holder> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

¹¹B. PERENS, "The Open Source Definition", <http://perens.com/OSD.html>

deliver anything upon the developer. Suppose that somebody wants to integrate the software in a larger product for a particular scope, and the software is unfit to it. The integrator is then permitted and invited to do all the modifications, including the adaptations and quality assurance activities, to make sure that the combination works. There is a considerable difference between this case and a proprietary software license. In proprietary software licensing consideration is exchanged against delivery of the software or even permission to use software, is a *sale* (Art. 1471 ICC). Being a sale, it bears certain statutory warranties, including the product being free from defects that reduce its intended use (Art. 1490 ICC). But the same cannot apply to FOSS, which is not “sold” — unless there is a separate agreement on that particular piece of code — but just offered for perusal. If there is a separate agreement, such as a software development agreement, the relationship between the client and the developer — in particular the liability for defective software — is governed by this specific contract and not by the license.

A liability cannot be construed on the basis of product liability rules as well, because product liability would invariably arise from a defect in a physical item, and without a contractual link other than the license, the developer cannot reasonably be considered the “provider”. At any rate, any such limitation would be void outright if the product liability law were applicable (Art. 124 of the Italian Consumers Code).

A tort liability would also be hard to find and the onus would be entirely on the plaintiff. Art. 1227 of the Italian Civil Code provides that damages are not due insofar as they could have been avoided on duly exercising diligence. But, more than that, it must be established that the damage has been caused by an unlawful act (i.e. an act against the law and conflicting with the conduct expected by an average subject). Now, the “exchange” between the developer and the user is “I give you the Freedoms, but the code is all you get, not a product, mind that I don’t guarantee anything”. It is generally understood that because the exchange is gratuitous, the avoidance of liability flows naturally from said act (again, except the case of gross negligence) unless a warranty is expressly given.¹² All this makes it particularly difficult to construe a solid case for tort liability of large proportions, and in any case there is no pre-made contractual language that can prevent it, especially because Art. 1225 of the Italian Civil Code limits maximum liability to what could have been reasonably expected when the obligation arose.

Liability for lack of title is also a possibility. Releasing software as FOSS by an upstream provider is an act upon which third parties might rely for downstream re-licensing. If there is a gap in the chain of title, that could mean that the lower end of the chain suffers losses, e.g. because of litigation it suffers from upstream, even if no knowledge of any infringement existed. Can this distributor of soft-

¹²As provided for by Art. 798 of the Italian Civil Code for defects of the thing which was donated, which stipulates that without an express pact the liability only arises from a willful act.

ware demand to be indemnified by its software provider who has “obfuscated” the real status of the copyright title of that particular piece of code? Such indemnification is hard to construe, because there is no contractual link between the party requesting indemnification and its upstream provider. What remains, in the absence of express warranties and representation, is a non contractual liability. Certainly the licenses have no warranties and representation — indeed the contrary is the case.

Any downstream developer or integrator must do its own due diligence or require other kinds of warranties and representations from its upstream — or better, do both. Indeed, that of providing certain levels of indemnification for FOSS is a business case for some companies.

The copyleft principle

Principle

A characteristic found in many (but not all¹³) FOSS licenses is the so-called “copyleft” principle. FOSS licenses which incorporate the copyleft principle,¹⁴ lay down by contract that everyone in the chain of consecutive users, in return for the right of use that is assigned, has to distribute to other users the improvements the software and its derivative works, if he chooses to distribute such improvements or derivative works. In other words, software which incorporates copyleft FOSS, must be distributed as copyleft FOSS. As a general statement, it is not possible to incorporate copyright protected parts of copyleft software in a proprietary licensed work.

Often Copyleft is referred to as “virality” of the license. However, this wording has a pejorative implication and is misleading as to how the copyleft principle operates. The copyleft clauses imposes a condition as in: “if you want to do X, then you must do Y, else you can’t do X”, but this concept has been mistaken for “if you do X, then you are obligated to do Y, else I can force you to do Y”. In other words, the result of distributing derivatives of copyleft software under a different license is an infringement of the upstream license, it is not that non properly licensed software is by magic transformed into copyleft software. It does not cause per se any re-licensing of the infringing work in copyleft software, unless the infringer wants to cure the violation by relicensing. In other words,

¹³Neither the principles (freedoms) of the Free Software movement, nor the Open Source Definition mandate the copyleft clause. Several FOSS licenses do not contain a copyleft clause. Examples hereof are the Berkeley Software Distribution (BSD) license and the Apache license

¹⁴E.g., GPL version 3 Art. 5 stipulates: “You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it” GPL version 2 Art. 2 b stipulates: “You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License”.

copyleft licenses are no more “infective” than proprietary ones (in which, if the price is not paid, the right to use the software is not obtained).

We have discussed above about the difference between the effects of an obligation and of a condition in the section on enforcement and we make reference to said discussion for further details.

Validity

The question of the validity of the copyleft clause coincides with the question whether an author can effectively lay down how derivative works have to be distributed. The answer to this question is affirmative. The author of the original work has no rights in the derivative work as a whole, but based on his rights in the original work he is able to permit or prohibit the distribution of the derivative work. A derivative work can therefore only be operated subject to the consent of the copyright owner of the original work.

An issue could in theory be raised against creating exclusive rights which are not provided by law, as copyleft is sometimes accused of doing. But copyleft does not create any exclusive right that is not already granted by the law: instead it carves its permissions out of the already existing exclusive right to authorize derivative works. Since the downstream recipient of the software must clear its derivatives vis à vis all upstream copyrights, this either is done generally and conditionally by using the copyleft license, or is done otherwise. If and insofar as the conditions are approved — and met — the software is clear. Otherwise the software is not clear and clearance must be sought on an individual basis, as in any other kind of software distribution. That, by the way, is how dual (proprietary + copyleft) licensing works, such that Richard Stallman has dubbed the scheme as “sale of exceptions [to a copyright license]”.¹⁵

Damages

According to Art. 158 of the Italian Copyright Law, damages caused by copyright violations are compensated under Italian law in accordance with the general legal principles applicable to unlawful acts (Art. 2056 and 2059 of the Italian Civil Code) and with the principles of breach of contractual obligations (Art. 1223, 1224 and 1225 of the Italian Civil Code). Those provisions establish that damages shall be awarded in a measure sufficient to restore the economic (Art. 2056 ICC) and moral (Art. 2059 ICC) losses of the aggrieved party. The economic loss is calculated in terms of actual damage and lost profit, limited to the damage that was foreseeable at the time of the breach unless the act was done intentionally or due to gross negligence.

Dual damages, triple damages or other forms of punitive damages are not awarded under Italian law. Traditionally, these damages were considered radically incompatible with fundamental principles of Italian Law (so called “divieto

¹⁵See R. STALLMAN, *On Selling Exceptions to the GNU GPL*, FSF website, available at <http://www.fsf.org/blogs/rms/selling-exceptions>

di locupletazione”). However, with the introduction of TRIPS, a limited version of punitive damages (i.e. damages unrelated to the actual loss suffered) has been introduced for patent and trademark violations, under the name of “civil punishment” — such as awarding to the rightholder the infringing products that have been confiscated. Similarly, in copyright violations an award of damages not directly related to the lost profits and the actual losses can easily be achieved by applying moral damages (Art. 2059 of the Italian Civil Code, expressly mentioned in Art. 158.3 of the Italian Copyright Law) and through an award equal the profits unlawfully obtained as a result of the violation by the violator (and profit here could include the advantage coming from the avoidance of production costs). Infringements of software copyrights follow the same regime as infringements of every other copyright. The aforementioned principle is therefore applicable in cases involving the infringement of software copyright. The same applies to FOSS.

Infringements of software copyrights follow the same regime as infringements of every other copyright. The aforementioned principle is therefore applicable in case of copyright infringements of software. Same applies to FOSS.

It may be assumed that the damage to the copyright owner will be in any case very limited if existing, as the author has made his work freely available. This argument is not necessarily bound to succeed, because the factual premise falls flat in its face in many cases where FOSS indeed is utilized for business purposes, as in the following cases.

Besides establishing a reputation and recognition — which can indeed be a valuable asset — an author can have other reasons to make his or her work “freely” available.¹⁶ The author may also gain a direct monetary advantage from the free distribution of his work. The simplest and traditional way is to add advertisements to the software (“ad-ware”). Another way is to offer specific services relating to the work, such as assistance, maintenance, customization, indemnification, etc., or other related products. In this latter example, the free circulation of the work ensures the work attracts many users. The author can generate his income from the provision of support and consulting services, or by licensing “proprietary add-ons”¹⁷ (roughly this is the “open core” strategy). Another business model is the so-called dual licensing model.¹⁸ This model uses — as the name suggests — two different licenses. The first license is often a strong copyleft license. This first — free — license ensures the work is circulated quickly and reaches a wide range of users. A second license without the copyleft conditions can then be obtained for a licensing fee by those who want to use the work in a wider application and at the same want to avoid that their own additions or the

¹⁶See e.g. C., DIBONA, D., COOPER and M., STONE, “Introduction”, in *Open Sources 2.0: The Continuing Evolution* (Ed. C., DiBona, D., Cooper and M., Stone), O’Reilly, 2006, p. XXV-XL.

¹⁷Add-ons are additions to the free work to which the author reserves all rights, and which can only be used against payment

¹⁸See e.g. M., OLSON, “Dual Licensing”, in *Open Sources 2.0: The Continuing Evolution* (Ed. C., DiBona, D., Cooper and M., Stone), O’Reilly, 2006, p. 35

overall product are affected by the copyleft conditions, which is e.g. the case of proprietary applications using FOSS libraries and components

However, there is no straightforward theory of damages for a FOSS licensing violation. Arguably, if the program is dual-licensed, it would be easy to establish the damage as the lost profits that the copyright holder has suffered, which corresponds to what the infringer ought to pay to obtain a proprietary license. If the infringing party has obtained licensing fees as a result of the infringement, again the damages could be determined relatively easily by calculating the share of the profit that has unfairly been generated as a result of the violation, using the fiction established in Art. 158. If this does not apply, the judge can refer to the costs unfairly avoided as a result of the violation (as if the FOSS developer had worked for the infringer), or the cost of the next most feasible proprietary alternative to the FOSS program — that the infringer has avoided licensing thanks to the violation — can be taken into consideration. This alternative can in fact be considered an indication of the price that the infringer ought to have paid to obtain a similar licensing from the FOSS developer at the same licensing conditions (in other words, as if the product was dual licensed). The problem with this latter approach is that in many cases the FOSS developer is unwilling or unable to license the software on proprietary terms. If he or she might issue a license, but refuses to do it on moral or other grounds, one could argue that this *increases* the damage instead of eliminating it, because the price to obtain this waiver to copyleft could be immensely high if negotiated *ex ante*. If the license is not possible (e.g., because of upstream constraints such as copyleft) again this is no reason to negate the compensation, because obtaining to an act of “violence” — the operative equivalent of a proprietary license from somebody who arguably would not be willing to license the software program on proprietary terms because moral motives — is a morally challenging act that must be compensated with moral damages. Moral damages can be awarded on an *ex bono et aequo* basis (Art. 2059, 2056 and 1226 ICC), which very frequently also takes into consideration the profit gained by the violator.

FOSS in the Public Administration

While the private sector is more or less free to decide what kind of software to adopt for their own use — there may be some restrictions coming from certain law, like the need to fill tax returns using a certain application, but this is seldom compelling — the public sector is forced to follow strict rules in the procurement of software. This goes beyond the rules imposed on public procurement procedures, aiming at ensuring equal treatment to all competitors (“par condicio”), it involves rules on *what kind* of software is to be procured.

The Code for the Digital Public Administration¹⁹ (“CAD”) has provisions that dictate how the software for the public sector must be. It says that it has to have certain characteristics, such as enabling interoperability and services exchange.

¹⁹Digs 85/2005 “Codice dell’Amministrazione Digitale”

It also dictate which kind of *licensing conditions* are to be explored and chosen after a comparison through a formalized procedure (“Comparative analysis”). Art. 68.1 contains said list:

- Software made *ad hoc* for or by the concerned administration
- Reuse of software made *ad hoc* for other administration [which is compelled to license it for free, and to hand over also the source code]²⁰
- Free and open source software [FOSS]
- Software obtained as a service (Cloud Computing)
- Software licensed under proprietary conditions
- Any combination of the above

Furthermore, Art. 68.1-ter provides that only when there is a motivated impossibility to procure software under the conditions of reuse and/or FOSS, then the public administration can redress to the acquisition of software under proprietary conditions.²¹ In other words, reuse and FOSS *must be preferred* ²² over proprietary software, even if on pure technical and economic bases proprietary solutions score better.

FOSS cases in Italy

No cases have been reported yet (November 2013).

Legal procedures

In Italy the nature of legal procedures in the field of copyright does not differ in nature from that in general civil cases, and a procedure for temporary relief is also available. A copyright case is heard by one of the 11 specialized sections for Industrial and intellectual property Law. If a case falls within the jurisdiction of a court that has no specialized sections, jurisdiction in the whole case (even if it has other components which are not subject to this specialized competence, e.g., the interpretation of an agreement) is allocated to the court with the specialized section on a regional basis. The court decides with a panel of three judges.

²⁰Art. 69 CAD

²¹See Aliprandi, Simone and Piana, Carlo (2013) *FOSS in the Italian public administration: fundamental law principles*, *International Free and Open Source Software Law Review*, 5(1), pp 43 – 50 DOI: 10.5033/ifosslr.v5i1.84, available at <http://www.ifosslr.org/ifosslr/article/view/84>

²²However, the effectiveness of this measure can be fully ascertained with an in-field survey after a period of time, as many of similar laws have been largely disregarded, absent clear compliant enforcement measures

Some special investigative powers have been given to the courts, such as ordering any involved third parties to provide information on the origin of the goods and their chain of distribution.²³ Also the possibility of obtaining a description of the infringing goods in addition to their seizure is noteworthy. A temporary or permanent enjoinder to distribute the infringing works can be obtained. The decision may include an order for publication in the press of its operative part at the expense of the the infringing party.

The author of the work can always join the proceedings to protect his or her own interests even if he or she has already disposed of the economic rights in the work.

Recommended literature

Articles

- FLOSS: gli indirizzi europei, la normativa italiana e le leggi regionali, Chapter 3 of the book “Finalmente libero! Software libero e standard aperti per le pubbliche amministrazioni” (2007, Mac Graw-Hill) edited by Michele Marchesi, Giulio Concas, Giulio De Petra, Flavia Marzano, Pietro Zanarini
- Marco Bertani: Profili giuridici delle licenze di software libero / open source nell’ordinamento italiano, in “I quaderni di dirittodautore.it”, Anno III, n. 24, <http://www.dirittodautore.it/quaderni.asp?mode=3&IDQ=82>
- Nerina Boschiero: Le licenze F/OSS nel diritto internazionale privato: il problema delle qualificazioni, in AIDA 2004 (2005, Giuffrè)
- Marco Ciurcina, Carlo Piana: Le licenze FLOSS: stato dell’arte ed evoluzioni, Chapter 11 of the book “Il software libero in Italia” (Shake, 2009) edited by Andrea Glorioso
- Carlo Piana: Licenze pubbliche di software e contratto, in I contratti, n. 7/2006, IPSOA; http://www.piana.eu/repository/720_727.pdf
- Marco Ricolfi: Software e limitazioni delle utilizzazioni del licenziatario, in AIDA 2004 (2005, Giuffrè)
- Lele Rozza: Le principali iniziative legislative sul FLOSS, Chapter 10 of the book “Il software libero in Italia” (Shake, 2009) edited by Andrea Glorioso
- Marco Saverio Spolidoro: Open source e violazione delle sue regole, in AIDA 2004 (2005, Giuffrè)
- Vincenzo Zeno-Zencovich, Pieremilio Sammarco: Sistema e archetipi delle licenze open source, in AIDA 2004 (2005, Giuffrè)

²³Art. 156 ter of the Italian Copyright Law

Books

- Simone Aliprandi: Copyleft and Opencontent. L'altra faccia del copyright (2005, PrimaOra)
- Simone Aliprandi: Apriti standard! Interoperabilità e formati aperti per l'innovazione tecnologica (2010, Ledizioni), <http://www.aliprandi.org/apriti-standard>
- Aa.Vv.: Open Source. Atti del Convegno (Foggia, 2-3 luglio 2004) (2005, Giuffrè)
- Giuseppe Sanseverino: Le licenze free e open source (2007, Edizioni Scientifiche Italiane)