

Diritto d'autore nel Software Libero e Open Source

Luca Sartoni

Agli autori di opere intellettuali, esplicitate nella realizzazione di software, viene riconosciuto il diritto morale e patrimoniale connesso all'opera stessa.

Il diritto morale è un diritto della personalità perciò imprescrittibile, irrinunciabile e non trasferibile. Il diritto patrimoniale, invece, assicura l'esclusività nelle scelte di sfruttamento economico del software.

Tra le attività che hanno in oggetto l'opera, spettanti esclusivamente all'autore, troviamo: la riproduzione, permanente o temporanea, totale o parziale, con qualsiasi mezzo o in qualsiasi forma, la traduzione, l'adattamento, la trasformazione e ogni altra modifica, qualsiasi forma di distribuzione al pubblico, la locazione e la vendita, sia dell'originale sia della copia.

Ovviamente la norma di cui stiamo parlando (Legge 633/1941 e succ. integrazioni) prevede anche una serie di diritti derivanti dall'essere licenziatario del software che si va ad utilizzare. Lo sfruttamento economico del software ha influenzato enormemente il legislatore e la prova sta nel fatto che i diritti riconosciuti ai destinatari dell'opera sono decisamente limitati se confrontati all'enorme tutela che spetta all'autore.

Un'altra importante conseguenza dello sfruttamento commerciale del software è la scissione totale dei due diritti di cui si è già parlato, il diritto morale e quello patrimoniale ingigantendo il secondo al punto da rendere quasi sconosciuto il primo.

Per poter capire meglio come si è giunti a questo risultato, è necessario raccontare una storia che ci coinvolge molto di più di quanto si possa credere.

Il software nasce agli inizi degli anni 50 dentro le università americane.

Lo sviluppo avveniva in modo collegiale tra i ricercatori; gli enormi costi derivanti dall'hardware lo rendevano esclusivo ai centri di ricerca.

Gli sviluppatori vedevano nella programmazione una disciplina matematica vecchia di secoli integrata con le novità di alcuni visionari come Turing e Godel, applicata alla moderna tecnologia disponibile.

Negli anni 70 il costo dell'hardware iniziò a non essere un problema insormontabile per molte aziende e lo sviluppo del software iniziò a distribuirsi presso enti a fine commerciale.

Le aziende considerarono la tutela del diritto patrimoniale l'unico modo per poter sfruttare il software che producevano e per ripagare gli investimenti che stavano sostenendo.

Mediante patti di non divulgazione, fatti firmare agli sviluppatori, le aziende imposero un pesante freno alla collaborazione che caratterizzava il lavoro dei programmatori.

Entro gli anni 80, quasi tutto il software prodotto era di tipo proprietario.

Con la disponibilità di hardware a basso costo, il business delle aziende si spostò sulla distribuzione del software che fino a quel momento era parte organica del prodotto che era distribuito in forma intergrata.

La complessità del software che veniva realizzato era in costante crescita. Mediante la segretezza del codice sorgente era possibile ottenere un business basato sulla distribuzione del software e sul legame che s'instaurava con il licenziatario che di fatto era vincolato al produttore e al distributore.

Nel 1983 Richard Stallman fondò il progetto GNU (GNU is Not Unix) con il proposito di liberare il software dalle pesanti limitazioni che la distribuzione commerciale imponeva.

Grazie alla collaborazione di migliaia di sviluppatori volontari, a Internet che consentì la coordinazione del lavoro e al kernel sviluppato da Linus Torvalds, nel 1991 venne pubblicato il primo sistema operativo libero: GNU/Linux.

Per poter considerare libero un software è necessario che la licenza mediante la quale sia distribuito permetta quattro fondamentali libertà:

- **Libertà 0 : eseguire il programma per qualsiasi scopo.**
- **Libertà 1 : studiare il programma e modificarlo per le proprie esigenze.**
- **Libertà 2 : distribuire copie in modo da aiutare il prossimo.**
- **Libertà 3 : migliorare il programma e distribuirne pubblicamente i miglioramenti, in modo tale che tutta la comunità ne tragga beneficio.**

La libertà 1 implica la necessità di mettere a disposizione il codice sorgente.

La licenza attraverso la quale viene distribuito Software Libero prende il nome di GPL (GNU General Public Licence).

Questa licenza, mediante l'esercizio del diritto d'autore, obbliga il licenziatario che riceve ed in taluni casi modifica il software, a ridistribuirlo mediante la medesima licenza.

Questa limitazione comporta una diffusione della licenza GPL verso tutto il software derivato. Secondo alcune interpretazioni, la licenza GPL impone, a volte, l'obbligo d'estensione di tale licenza anche alle librerie incorporate nel prodotto distribuito. Questo comporta una diffusione anche verso l'alto. Tale comportamento prende il nome di "diffusione virale della licenza GPL".

Per evitare questo comportamento, che può risultare sgradevole, indesiderato o nocivo, sono stati redatti diversi tipi di licenze che da una parte conservano i caratteri di libertà per l'utente e dall'altra non comportano obblighi troppo restrittivi nella realizzazione del software.

Una di queste licenze prende il nome di LGPL (GNU Lesser General Public Licence) ed è molto adatta nella distribuzione di librerie software.

Le idee di Richard Stallman erano viste con grande sospetto dall'ambiente commerciale statunitense e questo non rendeva facile la diffusione della licenze di software libero.

Per favorire dunque l'idea delle licenze liberali nel mondo degli affari, Bruce Perens, Eric S. Raymond, Ockman e altri cominciarono nel 1997 a pensare di creare una sorta di lobby a favore di una ridefinizione ideologica del software libero, evidenziandone cioè i vantaggi pratici per le aziende e coniarono il termine "Open Source". Ciò anche al fine di evitare l'equivoco dovuto al doppio significato di free nella lingua inglese, visto che spesso veniva interpretato come "gratuito" invece che come "libero".

Perchè un software possa essere riconosciuto dalla OSI (Open Source Initiative) e quindi essere conforme alle specifiche OSD (Open Source Definition) deve soddisfare dieci criteri:

1. **Libera redistribuzione**

La licenza non può limitare alcuno dal vendere o donare il software che ne è oggetto, come componente di una distribuzione aggregata, contenente programmi di varia origine. La licenza non può richiedere diritti o altri pagamenti a fronte di tali vendite.

Motivo: Imponendo la libera redistribuzione, si elimina la tentazione di rinunciare a importanti guadagni a lungo termine in cambio di un guadagno materiale a breve termine, ottenuto con il controllo delle vendite. Se non vi fosse questa imposizione, i collaboratori esterni sarebbero tentati di abbandonare il progetto, invece che di farlo crescere.

2. **Codice sorgente**

Il programma deve includere il codice sorgente e ne deve essere permessa la distribuzione sia come codice sorgente che in forma compilata. Laddove alcune forme di un prodotto non siano distribuite con il relativo codice sorgente, deve essere chiaramente indicato il modo per ottenerlo, ad un costo non superiore ad una

ragionevole spesa di distribuzione, preferibilmente scaricandolo gratuitamente da Internet. Per codice sorgente si intende la forma in cui un programmatore preferirebbe modificare il programma. Codice sorgente deliberatamente reso illeggibile non risponde ai requisiti. Forme intermedie come l'output di un preprocessore o compilatore non rispondono ai requisiti.

Motivo: Si richiede l'accesso al codice sorgente poiché non si può far evolvere un programma senza poterlo modificare. Il nostro obiettivo è rendere facile l'evoluzione del software, pertanto richiediamo che ne sia resa facile la modifica.

3. **Prodotti derivati**

La licenza deve permettere modifiche e prodotti derivati, e deve permetterne la distribuzione sotto le stesse condizioni della licenza del software originale.

Motivo: La sola possibilità di leggere il codice sorgente non è sufficiente a permettere la revisione indipendente del software da parte di terzi e una rapida selezione evolutiva. Per garantire una rapida evoluzione, deve essere possibile sperimentare modifiche al software e redistribuirle.

4. **Integrità del codice sorgente originale**

La licenza può impedire la distribuzione del codice sorgente in forma modificata, a patto che venga consentita la distribuzione dell'originale accompagnato da "patch", ovvero file che permettono di applicare modifiche automatiche al codice sorgente in fase di compilazione. La licenza deve esplicitamente permettere la distribuzione del software prodotto con un codice sorgente modificato. La licenza può richiedere che i prodotti derivati portino un nome o una versione diversa dal software originale.

Motivo: Incoraggiare il miglioramento è bene, ma gli utenti hanno diritto di sapere chi è responsabile del software che stanno usando. Gli autori e i tecnici hanno diritto reciproco di sapere cosa è loro chiesto di supportare e di proteggersi la reputazione.

Perciò, una licenza open source deve garantire che il codice sorgente sia facilmente disponibile, ma può eventualmente richiedere che esso sia redistribuito solo in forma originale più file patch. In questo modo le modifiche "non ufficiali" possono essere rese disponibili pur rimanendo distinte dal codice sorgente originale.

5. **Discriminazione contro persone o gruppi**

La licenza non deve discriminare alcuna persona o gruppo di persone.

Motivo: Per ottenere il massimo beneficio dal processo, il massimo numero di persone e gruppi deve avere eguale possibilità di contribuire allo sviluppo del software. Pertanto viene proibita l'esclusione arbitraria dal processo di persone o gruppi.

Alcuni paesi, inclusi gli Stati Uniti, hanno restrizioni all'esportazione di certi tipi di software. Una licenza conforme all'OSD può avvertire gli utenti di possibili restrizioni e ricordare loro che sono obbligati a rispettare la legge; in ogni caso non può incorporare tali restrizioni essa stessa.

6. **Discriminazione per campo d'applicazione**

La licenza non deve impedire di far uso del programma in un ambito specifico. Ad esempio non si può impedire l'uso del programma in ambito commerciale o nell'ambito della ricerca genetica.

Motivo: L'intenzione principale di questa clausola è di proibire trappole nelle licenze che impediscano al software open source di essere usato commercialmente. Vogliamo che le aziende si uniscano alla nostra comunità, non che se ne sentano escluse.

7. **Distribuzione della licenza**

I diritti allegati a un programma devono essere applicabili a tutti coloro a cui il

programma è redistribuito, senza che sia necessaria l'emissione di ulteriori licenze.

Motivo: Questa clausola intende proibire la chiusura del software per mezzi indiretti, come un obbligo di sottoscrizione di accordi di non diffusione.

8. **Specificità ad un prodotto**

I diritti allegati al programma non devono dipendere dall'essere il programma parte di una particolare distribuzione di software. Se il programma è estratto da quella distribuzione e usato o redistribuito secondo i termini della licenza del programma, tutti coloro che ricevano il programma dovranno avere gli stessi diritti che sono garantiti nel caso della distribuzione originale.

Motivo: Questa clausola impedisce un'ulteriore classe di licenze-trappola.

9. **Vincoli su altro software**

La licenza non deve porre restrizioni su altro software distribuito insieme al software licenziato. Per esempio, la licenza non deve richiedere che tutti gli altri programmi distribuiti sugli stessi supporti siano software open source.

Motivo: I distributori di software open source hanno il diritto di fare le loro scelte riguardo al software che intendono distribuire.

10. **Neutralità rispetto alle tecnologie**

La licenza non deve contenere clausole che dipendano o si basino su particolari tecnologie o tipi di interfacce.

Motivo: Questa clausola è diretta in particolar modo a quelle licenze che richiedano un gesto esplicito di approvazione da parte dell'utente, al fine di stabilire un contratto. Clausole che richiedano un "click" su interfacce web o di altro tipo possono essere in conflitto con importanti metodi di distribuzione del software, come i siti FTP, le raccolte su CDROM e le copie distribuite sul Web. Tali clausole possono rendere difficoltoso il riutilizzo del software. Le licenze valide devono permettere la possibilità che: 1) il software venga distribuito mediante canali diversi dal Web, sui quali non si possa richiedere un "click" esplicito prima di iniziare il download, e che 2) il programma in oggetto, o sue porzioni, possano essere utilizzare in ambienti privi di interfaccia grafica, nei quali non si possa richiedere la presenza di specifiche finestre di dialogo.

Quando si parla di Software Open Source, non sempre si sta facendo riferimento ad una licenza OSD. Molto spesso si intende esclusivamente la disponibilità alla presa visione del codice sorgente da parte del licenziatario e alcune volte ne è concessa la modifica.

La differenza sostanziale tra il Software Libero e quello Open Source è di carattere filosofico e tecnico. Il Software Libero si basa su concetti di libertà e fa della collegialità nello sviluppo un pilastro fondante. Inoltre promuove interazione tra il licenziatario e l'autore dando, tanto al primo quanto al secondo, libertà e diritti. Il Software Open Source ha eliminato alcuni concetti considerati utopici per sostituirli con altri più in linea al settore commerciale.

Entrambe le licenze di distribuzione si pongono però l'obiettivo comune di rivalutare il ruolo dell'utente troppo spesso sminuito dalla distribuzione proprietaria.