

# Un po' di chiarezza sul mondo FLOSS: Open source, Free, pubblico dominio e licenze Creative Commons.

di Stefano Penge

Questo testo è rilasciato sotto licenza  
Creative Commons 4.0 BY SA



## ***Che cosa significa "codice sorgente"***

Tutti i programmi che usiamo (da quelli sugli smartphone a quelli sul web) e che ci forniscono servizi (dalla gestione del traffico telefonico al controllo dei satelliti) sono simili: sono insiemi di istruzioni che i computer eseguono. Come ricette o sceneggiature.

La versione del programma che viene eseguita dai computer è, di solito, ottimizzata per essere veloce e senza errori. Questo processo (che si chiama "compilazione") genera una versione del programma che può essere letto dai computer, ma non dagli esseri umani.

Ma ogni programma è stato in origine scritto da qualcuno, in un certo linguaggio di programmazione (ce ne sono migliaia, come lingue umane). La prima versione del programma, quella originale, non ottimizzata, si chiama "codice sorgente". E' un testo, e come tutti i testi in teoria si può leggere – avendo le competenze per farlo - , modificare, copiare. Ma non necessariamente queste operazioni sono permesse dall'autore, o meglio da chi detiene il copyright sul programma. Normalmente, insieme al programma si riceve una licenza che specifica cosa *non* si può fare con esso: non si può copiare, non si può analizzare, non si può modificare. E' il caso, ad esempio, di Microsoft Windows o di MS Office, ma anche della maggior parte delle app sui nostri smartphone.

Alcuni programmi invece sono distribuiti con una licenza che permette esplicitamente di fare queste operazioni. Sono i programmi con licenza "aperta". Un esempio è Linux, che è un sistema operativo come Windows, o Libre Office, che è l'omologo di Microsoft Office. O dello stesso Android.

## **Licenze**

Quando si parla di software libero si fa spesso confusione tra concetti diversi, come per esempio la gratuità e la libertà.

Forse avete letto il termine Creative Commons (CC).

**Creative Commons** è una licenza che si applica a documenti (testi, audio, immagini, video). E'

un'idea nata all'inizio di questo millennio da Lawrence Lessing, un professore di diritto dell'Università di Stanford, e si ispira ai "commons", cioè ai terreni che nella tradizione normativa anglosassone sono "comuni", dove tutti possono pascolare o tagliare legna. Questo concetto è trasferito dalla terra all'informazione.

L'altra caratteristica delle licenze CC è che sono modulari, cioè l'autore può decidere quali diritti concede all'utilizzatore.

**GPL** (General Public License), BSD, MIT, Apache, Mozilla, EUPL etc sono invece licenze che si applicano principalmente al codice sorgente. All'interno delle licenze aperte per il codice sorgente ci sono due grandi classi: le licenze ufficialmente approvate dalla Open Source Initiative e quelle che soddisfano i requisiti della Free Software Foundation (vedi sotto).

**Public Domain** si riferisce a software (o a contenuti) che non hanno un autore conosciuto, o per cui l'autore ha rinunciato ai propri diritti.

**Freeware** invece si riferisce esclusivamente a software gratuito, ma non dice nulla sulla licenza o sulla disponibilità del codice sorgente. Molti programmi didattici sono freeware, ma non hanno una licenza "open". Significa che sono gratuiti, ma non che sono aperti. Se fossero stati rilasciati con licenza GPL qualcuno avrebbe potuto migliorarli o tradurli in altri linguaggi, per altri sistemi operativi.

## **Motivazioni**

La questione dell'apertura del codice sorgente ha due motivazioni distinte: quella etica e quella tecnica. La motivazione etica è quella che nasce agli albori della storia dell'informatica. Originariamente, l'hardware veniva venduto, il software no. Ad un certo punto, le aziende cominciarono a capire che rendere pubblico un sorgente significava avvantaggiare la concorrenza. Il codice venne "chiuso", nel senso che venne utilizzata la legge sul diritto d'autore e ai clienti venne impedito sia di analizzare il sorgente, sia di modificarlo, sia di trasferirlo ad altri. Contro questo processo si sono alzate alcune voci, prima fra tutte quella di Richard Stallman, programmatore al laboratorio di Intelligenza Artificiale del MIT, che a partire dal 1983 ha cominciato a scrivere un sistema operativo compatibile con UNIX (che era il sistema operativo più diffuso) ma libero. Di questo sistema operativo, una parte importante (il kernel, cioè il nocciolo) è stato scritto inizialmente da Linus Torvalds, un programmatore finlandese. La somma di queste parti ha dato origine a LINUX. La motivazione tecnica è quella che si traduce nell'aspettativa che un codice sorgente aperto sia più facile da migliorare di uno chiuso. Inoltre, se il software è chiuso, è meno sicuro, perché non si può sapere se contiene *bachi* (errori) o *backdoor*, cioè parti del codice che permettono un'intrusione da parte di chi ha scritto il codice, oppure una raccolta di informazioni sull'utente. Le due motivazioni sono indipendenti, anche se associate.

La visione del Software Libero (Free Software) è legata agli aspetti etici. Free significa libero (*non necessariamente gratis*).

Un software è libero se garantisce quattro libertà fondamentali:

- *Libertà 0: Libertà di eseguire il programma per qualsiasi scopo*
- *Libertà 1: Libertà di studiare il programma e modificarlo*

- *Libertà 2: Libertà di ridistribuire copie del programma*
- *Libertà 3: Libertà di migliorare il programma e di distribuirne pubblicamente i miglioramenti.*

Il software libero è legato ad una comunità di sviluppatori che migliorano condividendo il proprio lavoro. Si basa sull'idea che non ha senso nascondere quello che si fa, ma che usando quello che hanno fatto gli altri si può fare di più e meglio. Naturalmente bisogna essere bravi, e l'etica "hacker" è anche un'etica in cui conta di più l'abilità che il denaro.

"Opensource" è invece un termine che nasce nel 1998 ad opera di Bruce Perens, Eric Raymond e altri, e ha portato alla creazione di un ente (Open Source Initiative, <http://opensource.org/>) che valuta e gestisce le licenze. L'obiettivo era quella di dare una credibilità anche commerciale al software aperto e spingere le aziende a investirci. Il vantaggio è duplice: da un lato, se un'azienda non deve acquistare licenze per i sistemi con cui sviluppa software, può sopravvivere con investimenti molto più piccoli; dall'altro, se rilascia il proprio software con una licenza open, permette ad altri di scaricarlo, migliorarlo, utilizzarlo, e quindi in definitiva di contribuire alla sua diffusione.

Per intendere complessivamente tutto l'ambito dei software a licenza aperta senza distinguere tra software libero e opensource si usa a volte il termine FLOSS (Free-libre Open Source Software).

## **Aspetti economici**

Spesso il software FLOSS è *anche* gratuito, ma non lo deve essere necessariamente. Ma è sulla gratuità che si accentra il dibattito.

Per prima cosa, occorre domandarsi: come è possibile che un prodotto sia gratuito?

E' chiaro che il lavoro di produzione (e manutenzione) del software, come qualsiasi altro lavoro, deve essere pagato. Ci sono varie possibilità:

- lavoro pagato da un'impresa che decide di rinunciare a possibili ricavi dalla vendita di quel software (per immagine o perché in questo modo altre società potranno contribuire al suo miglioramento)
- lavoro pagato da un ente pubblico. Qui nascono problemi relativi alla distribuzione: l'AGID, (l'Agenzia per l'Italia Digitale) sostiene che se un dipendente pubblico "regala" un software che ha prodotto nell'orario di lavoro è passibile di denuncia
- lavoro svolto da volontari che non hanno bisogno di essere pagati.
- lavoro pagato da un'impresa che offre servizi basati su quel software

Il primo caso è quello di Android, che utilizza il kernel di Linux ed è rilasciato sotto licenza Apache. L'ultimo è quello delle imprese che si basano sul FLOSS per sopravvivere.

Oggi il FLOSS è una realtà economica importante, anche in Italia. La quasi totalità dei siti web si poggia su software opensource, sia per ragioni di costi che di performance e sicurezza: dal sistema operativo dei server ai vari servizi (HTTP, posta, database) fino ai linguaggi e alle applicazioni che gestiscono i siti. Adottare il modello di business opensource per un'azienda significa vendere servizi basati su programmi opensource: formazione, personalizzazione, consulenza.

Per le aziende informatiche tradizionali, la presenza di un'offerta FLOSS è una minaccia. I clienti – privati o pubblici - potrebbero decidere di utilizzare i servizi di un'azienda che propone software FLOSS al posto del software proprietario. Il costo delle licenze di un computer per ufficio basato su Windows più MS Office è di qualche centinaio di euro all'anno, il che non è molto. Ma se questo costo viene moltiplicato per tutti i computer di un'Ente pubblico, si capisce che il passaggio ad un sistema basato su Linux e Libre Office – che ha un costo di 0 euro in licenze - è molto vantaggioso economicamente.

Per evitare questo rischio, di solito le aziende sostengono che il software FLOSS funziona peggio, è meno garantito, o che il costo totale dell'operazione di "migrazione" da software proprietario a software libero viene a costare di più che continuare a utilizzare il software proprietario.

Questa difesa è legittima. Quello che va notato è che per l'economia globale del Paese c'è una differenza importante.

Il valore di una merce tradizionale è dato dal mercato, ma anche dal lavoro che è stato necessario per produrla. Nel caso del software, questo è meno vero, perché di un singolo software si possono fare infinite copie praticamente a costo zero. In pratica, genera velocemente dei margini che vanno a chi possiede il copyright del software.

Quando si acquista un servizio basato su software FLOSS, quello che si paga non è la merce, ma il lavoro che rende possibile quel servizio. Significa che il denaro va a pagare stipendi, contribuisce a creare occupazione. Da questo punto di vista, che lo Stato e gli Enti Pubblici favoriscano, o quanto meno equiparino, software proprietario e software FLOSS, significa spingere lo sviluppo di competenze e di imprese che adottano quel modello di business.

Non c'è niente di male nel fatto che un'azienda venda licenze di software prodotto all'estero. Ma è sicuramente più utile all'economia nazionale che si punti sullo sviluppo di software piuttosto che sulla sola vendita.

Contemporaneamente, quello che sta succedendo negli ultimi anni, purtroppo, è che le aziende informatiche si limitano a vendere le licenze del software opensource, esattamente come se fosse software proprietario. Molto spesso esistono diverse versioni dello stesso software:

- la versione "community", che è liberamente scaricabile, non garantita e non supportata
- una o più versioni "enterprise", che vengono vendute con un supporto. Queste versioni portano incorporati dei livelli di servizio, delle garanzie, degli aggiornamenti.

Le versioni enterprise hanno spesso delle funzionalità aggiuntive rispetto a quelle community. Tuttavia dovrebbero essere accompagnate dallo stesso tipo di licenza, ovvero dovrebbero essere modificabili, ridistribuibili etc. Questo a volte è invece limitato da contratti aggiuntivi.

In alcuni casi, inoltre, si verifica che la versione community è malfunzionante e poco aggiornata. Insomma, va verificato caso per caso, non bastano le etichette.

## ***Diritto al sapere e FLOSS***

Di fatto le scuole - e le Università - usano oggi molto software FLOSS, non perché sia aperto e modificabile, ma perché si scarica gratuitamente. Non si pongono il problema di rendere possibile un'economia locale del software aperto, non sentono questa responsabilità. Usano - qualche volta - Libre Office, ma non si curano di far crescere una generazione di informatici che sanno sviluppare

moduli per Libre Office. Il caso recentissimo della Provincia di Bolzano è eclatante: dopo essere passati anni fa a Libre Office, sono ora ritornati alle licenze di Microsoft Office 365 motivando la scelta con la superiorità tecnica della suite Microsoft, con il risparmio (!) e con la mancanza di informatici competenti sulle tecnologie open nell'area provinciale.

Facciamo un altro esempio: il sito della scuola - fatto con Wordpress o Joomla – molto spesso lo cura il professore di informatica la sera, da casa. Costa di meno che affidarlo ad una società esterna. Che le competenze o la disponibilità di tempo che può avere il (rispettabilissimo) docente non siano sufficienti, o che in questo modo si tagli una fetta di mercato importante per le cooperative, le piccolissime imprese locali che offrono servizi legati al web, non sembra essere percepito come un problema. C'è un taglio alla spesa, è vero, ma non c'è un investimento locale. Uno stato che spende di meno, ma non usa il risparmio per investire sulle competenze, non va lontano. Dal punto di vista dell'economia nazionale, avrebbe più senso che la personalizzazione, la gestione, l'hosting, fossero affidate a imprese locali, e che i risultati di queste personalizzazioni fossero messe a disposizione degli altri (come impone anche la legge sul riuso del software). Purtroppo non succede quasi mai.

Infine: non va confuso il diritto al sapere con il diritto ad avere software gratis. Ad esempio, la pretesa che tutto il software didattico (o in generale quello utile agli enti educativi) sia gratuito "perché serve all'educazione" è quanto meno strana. Equivale a pretendere che le lavagne o libri siano gratis. Ora il contenuto del libro è una cosa; i servizi collegati (la stampa, la diffusione, l'aggiornamento) un'altra. Software gratis significherebbe la chiusura di tutte le attività economiche collegate: ce la sentiamo di mandare tutti a casa in nome del diritto all'informazione? E perché non anche quelli che producono banchi, o computer?

## ***Economia globale e gratuità***

Abbiamo visto che la gratuità è una caratteristica spesso associata al software FLOSS, ma che non esaurisce i suoi vantaggi. Contemporaneamente, bisogna domandarsi: ma è sempre bene che un software o un servizio sia gratuito?

Se si usa un servizio ci si dovrebbe interrogare sulla sostenibilità di quel servizio. Perché domani potrebbe finire (ci sono casi di social network che hanno chiuso, con tutti i dati degli utenti dentro: cosa fareste se domani chiudesse Facebook o se diventasse a pagamento?) o perché potrebbe continuare a vivere solo facendo chiudere gli altri. Si potrebbe definire il "cannibalismo del gratuito". Se in una passeggiata in montagna trovo una sorgente, bevo, perché so che tanto l'acqua continuerà a sgorgare. Se trovo un rifugio con qualcosa da mangiare, prima di finire tutto penso al fatto che fino al prossimo rifornimento i prossimi visitatori non troveranno niente. E' questa forma di attenzione ecologica (non nel senso di "verde", ma nel senso di pensiero che guarda al ciclo delle risorse) che manca completamente nei consumatori digitali. Uso un servizio gratuito offerto da X, ma non mi domando quali saranno gli effetti del mio uso - non solo su di me e su X, ma sul resto dell'ecosistema. Se Y pensava di sopravvivere vendendo quel servizio, quando io e tutti gli altri usiamo il servizio gratuito di X, Y dovrà chiudere.

Occorre avere consapevolezza che anche usare un servizio gratuito non è un atto senza conseguenze economiche.

C'è un equilibrio necessario tra *responsabilità* e *convenienza*. Equilibrio che nel caso del digitale è talmente invisibile che viene pesantemente rotto a vantaggio della sola convenienza. Insomma, c'è un diritto a gestire le proprie risorse, ma anche un dovere a gestire quelle comuni. Siccome il digitale non si tocca con mano, sembra sempre che sia eterno come la sorgente in montagna - e invece ad un certo punto si secca.