

## Sintesi dell'intervento “Perché Linux?” di Samuele Farfarini

### 1 - Apertura - Sia chiaro cosa è Linux

#### - **Linux è un software libero**

#### - **Differenze tra software libero e software gratuito**

- Il software libero non è come il software gratuito, il quale è sì gratuito, ma di fatto non lascia all'utente nessuna libertà di scelta.

“Libero” si pone ad uno stadio più elevato di “gratuito”. Un software libero è un software di cui sono garantiti alcuni diritti fondamentali, come verrà spiegato meglio più avanti:

- 1) l'utente ha la libertà di eseguire il programma per qualsiasi scopo;
- 2) l'utente ha la libertà di modificare il programma secondo i propri bisogni (perché questa libertà abbia qualche effetto in pratica, è necessario avere accesso al codice sorgente del programma, poiché apportare modifiche ad un programma senza disporre del codice sorgente è estremamente difficile);
- 3) l'utente ha la libertà di distribuire copie del programma, gratuitamente o dietro compenso;
- 4) l'utente ha la libertà di distribuire versioni modificate del programma, così che la comunità possa fruire dei miglioramenti apportati.

#### - **Software freeware sono gratuiti, ma non liberi**

- Un software freeware è sì gratuito, ma non libero. Se qualcuno ha mai provato a guardare la licenza di un software freeware, come ad esempio AntiVir o Avast, si accorgerà che questi software non sono affatto libero, bensì non possono essere redistribuiti, non possono essere usati per scopi professionali, ovvero in contesti aziendali che generano profitto, non possono essere modificati e soprattutto non è disponibile il codice sorgente. Di fatto viene permesso solo l'uso del software per fini personali ed entro i termini temporali della licenza.

#### - **Linux è anche gratuito, ma non necessariamente**

- “Free speech, not free beer”, è questo il significato in soldoni della definizione di “software libero”, secondo Richard Stallman. “Free” significa quindi libero, e non gratuito.

E' vero che molte distribuzioni di GNU/Linux sono effettivamente gratuite, o al limite costano il solo supporto su cui sono alloggiate, ma è vero che questa non è l'unica soluzione possibile. In realtà il software libero potrebbe essere pagato per finanziare a sua volta l'azione di sviluppo dello stesso.

#### - **Anche quando paghi hai molti servizi in più (es. SuSE PRO, Linspire)**

- Vi sono molte aziende che hanno deciso di creare un business su Linux, per trarre profitto dalla vendita del software. Il software che commerciano, però, è libero, soggetto alla licenza GNU GPL, e come tale non ammette lo scopo di lucro. Pertanto tali software vengono resi disponibili con l'aggiunta di manuali e con contratti di supporto telefonico, ed in più sono spesso comprensivi di alcuni pacchetti non liberi, soggetti ad altre condizioni di riproducibilità.

L'esiguo costo e l'ottimo supporto offerto fanno di queste distribuzioni un'ottima scelta a livello professionale: basti pensare che una SuSE 10 Professional costa 99€, e comprende software per 2 dvd. Ad acquistare parte dei software equivalenti per ambiente Windows, tramite licenze proprietarie, può costare facilmente anche oltre i 5000 euro.

<<http://www.suseshop.it/novell/>>

## 2 - Storia - Come è stato possibile tutto ciò?

### **- Cenni alla storia di Unix**

- Nel 1965, i laboratori della AT&T stavano lavorando con la General Electric e il Project MAC del MIT per scrivere un sistema operativo chiamato Multics. Decisero però che il progetto non sarebbe andato in porto e sciolsero il gruppo. Si decise di abbozzare un sistema operativo che doveva incontrare le necessità dei Laboratori della Bell. Vennero implementate le idee di Multics e, con un gioco di parole su quel nome, al sistema venne dato il nome Unix. Nel 1973 Unix venne riscritto in C per renderlo portabile su altre macchine, ma alla fine degli anni '70 alla AT&T venne vietato di concorrere nel mercato dell'informatica, e questa vendette a modico prezzo un sacco di licenze a vari college e università. Successivamente per Unix arrivò anche il successo commerciale.

<<http://it.wikipedia.org/wiki/Unix>>.

### **- Il nuovo concetto di software proprietario (Microsoft)**

- Il software era sempre stato proprietario fino a quel momento, in quanto sviluppato sempre internamente alle aziende e poi diffuso agli hobbisti in forma di file binari. Però la prima persona a tirare fuori il problema della proprietà intellettuale sul software è stato Bill Gates, della neonata Microsoft.

Il 3 febbraio 1976, Bill Gates, con una lettera aperta agli hobbisti, scrive: "Secondo me, la cosa più critica nel mercato hobbistico, in questo preciso momento, è la mancanza di corsi e libri di software validi e di software di qualità. (...) Ma verrà mai scritto del software di qualità per il mercato hobbistico? (...) Le reazioni che abbiamo avuto dalle centinaia di persone che dicono di usare il BASIC sono state tutte positive. Tuttavia sono evidenti due fatti sorprendenti: 1) la maggior parte di questi "utenti" non ha mai acquistato il BASIC (meno del 10% di tutti gli utenti Altair ha acquistato il BASIC); 2) se si considera l'ammontare delle royalty risultanti dalle vendite agli hobbisti, il tempo che abbiamo dedicato all'Altair BASIC vale meno di due dollari l'ora. (...) Come mai? Come la maggior parte degli hobbisti non può non sapere, la maggioranza di voi ruba il software che usa. L'hardware va pagato, ma il software è una cosa da condividere. A chi importa se la gente che ci ha lavorato viene pagata? (...) Una cosa che invece riuscite a fare è impedire che venga scritto del buon software. Chi può permettersi di svolgere gratuitamente un lavoro di qualità professionale? (...) Sarei lieto di ricevere lettere da chiunque voglia pagare il dovuto o abbia suggerimenti o commenti da fare."

Da questa lettera si nota come Bill Gates per primo estragga e svisceri il concetto di proprietà intellettuale sul software, e sostenga che il non rispetto della licenza o l'acquisto presso il produttore originario è un reato. Questa lettera un po' il punto di inizio del concetto di software proprietario, regolato la licenze di non-diffusione.

<<http://www.semlug.net/stuff/opuscolo.pdf>>

### **- le Fork, Unix proprietari, Solaris, BSD (Berkeley University) --> MacOS X**

- Negli anni Unix, essendo stato un sistema venduto in molte licenze, venne in parte rilasciato ed in parte modificato, per andare a formare moltissime versioni differenti dello stesso sistema, come illustrato in figura. Le più famose sono BSD, creato nell'università di Berkeley, SunOS, dell'università di Stanford, poi passato alla SUN Microsystems con il nome di Solaris, e alcuni Unix proprietari, come AIX, IRIX o Xenix.

Il ramo di BSD, essendo quello di cui era disponibile il codice sorgente, si è dimostrato anche il più prolifico, dando vita numerosi progetti, come Solaris o NextStep, che poi è sfociato in MacOS X.

Lo stesso MacOS X, che deriva da NextStep e ha significative iniezioni di codice da

FreeBSD, recupera anche molto codice da Linux, di cui recupera alcune strutture. In questo senso i due sistemi sono cugini. <<http://www.levenez.com/unix/history.html>>, <<http://it.wikipedia.org/wiki/Immagine:Unix.png>>

#### **- La nascita del movimento per il free software (Richard Stallman e il progetto GNU)**

- Stallman ha fondato il movimento "Free Software". Maturò quest'idea quando nel laboratorio del MIT dove lavorava sostituirono il loro sistema interno (ITS) con un sistema proprietario della Digital. Per l'uso di tale software era necessaria una dichiarazione di non-diffusione. Stallman non voleva stare a questo gioco, dato che per lui non condividere ciò che si ha con gli altri è essere persone cattive. Si accorse quindi di essere uno sviluppatore di sistemi operativi, e quindi cominciò a programmare il suo, in modo che tutti avrebbero potuto usufruirne liberamente. Scelse di renderlo compatibile con Unix in modo che gli utenti Unix potessero passare ad esso. Lo chiamò quindi GNU, acronimo dato secondo la tradizione hacker, che significa "GNU is Not Unix". Nel 1984 lasciò il MIT per lavorarvi, e da lì per i successivi 7 o 8 anni. All'inizio degli anni '90 ormai GNU era completo, mancando solo il kernel, il cui sviluppo era appena iniziato.

#### **- Linus Torvalds e l'arrivo di Linux nel progetto GNU**

- Nel 1991 Linus Torvalds, che lavorava all'università di Helsinki, programmò in modo indipendente un kernel compatibile con Unix, dato che desiderava continuare il lavoro universitario a casa, e lo chiamò Linux. Rilasciò il codice sorgente, e fu allora che alcune persone su internet trovarono Linux e GNU e videro che, senza aggiustamenti troppo grossi, sarebbero potuti stare assieme. Misero assieme i due pezzi, con il supporto della Free Software Foundation, formando il sistema GNU/Linux.

### 3 - Filosofia - Cosa comporta usare Linux?

#### **- Cenni di licenza GPL**

- GPL è l'abbreviazione GNU General Public License, che è la licenza che stabilisce se un software è "free". Questa licenza garantisce il copyright del programma all'autore, il quale, diffondendo il proprio software con questa licenza consente a qualsiasi utente di usare, copiare, redistribuire e modificare a proprio piacimento tale programma. Unici oneri per l'utente sono che le copie e le modifiche di tale software debbano essere a loro volta soggette alla stessa licenza, ed il destinatario della copia o modifica debba essere informato dei propri diritti.

Con il software deve essere incluso anche il codice sorgente, o deve essere incluso il riferimento a dove poter prelevarlo per consentire le modifiche e, nel caso l'utente non sia un programmatore, è libero di pagarne uno perché effettui le modifiche per lui. Inoltre è possibile chiedere un compenso in denaro per la diffusione o copia del prodotto, purché questo copra i soli costi di impiego per la copia o purché, se superiore, venga utilizzato per finanziare lo sviluppo del software stesso.

Non essendo questa una licenza che impone un pagamento per l'uso del software, non è valida alcuna forma di garanzia sul software stesso, né l'autore può essere citato in giudizio per qualsivoglia danni o mancati guadagni derivati dall'impiego di tale software.

#### **- Cenni di paragone con la EULA**

- EULA è l'abbreviazione di End User License Agreement, la licenza usata da Microsoft per Windows XP. Essa stabilisce che il software in questione non è venduto al cliente, ma solo concesso in licenza d'uso secondo i termini della licenza stessa. E' quindi sbagliato credere che il software acquistato sia proprio, perché è soltanto concesso, e la Microsoft, nel caso ritenga la licenza violata, può risolvere il contratto costringendo l'utente a distruggere fisicamente la copia in proprio possesso. La licenza EULA vieta inoltre la copia, la modifica

e la redistribuzione del software sotto qualsivoglia altra condizione. Nello specifico il software non può essere eseguito su più di una macchina contemporaneamente, e non su più di 2 cpu nello stesso pc. Il software non può inoltre essere trasferito, nemmeno se comprensivo di licenza, oltre il primo utente che lo ha acquistato.

E' curioso notare che, in nome di non specificate norme di sicurezza, il computer può smettere di funzionare dopo la variazione di un certo numero di componenti hardware, in quantità ignota, qualora non venga richiesto un nuovo codice di attivazione. E' invece ancora più curioso notare che Microsoft si riserva il diritto di installare software sul computer senza il consenso dell'utente, di permettere a terzi di installare software, sempre senza alcun consenso, che limiti la fruizione di contenuti multimediali in modo che la proprietà intellettuale venga preservata. Si riserva inoltre il diritto di prelevare qualunque informazione, personale dell'utente e non, dal computer senza alcun consenso ed usarlo o diffonderlo presso terzi. Cosa più simpatica di tutte è che la Microsoft non riconosce alcun tipo di responsabilità sugli effetti del software nel contesto d'uso, e dichiara di non poter essere citata in giudizio per qualsiasi danno, anche fisico o finanziario, derivante dall'uso, o da mancanze, bug o falle, di tale software. La Microsoft non risponde inoltre di nessuna garanzia sul prodotto, se non la sostituzione del supporto entro 90 giorni, né sulla correttezza delle informazioni diffuse dal proprio supporto tecnico. Il che significa che se il supporto tecnico fornisce un'informazione sbagliata, e l'applicazione di questa provoca un danno, Microsoft non può essere citata in giudizio per questo.

“Dulcis in fundo” Microsoft ritiene di poter negare il “pacifico godimento” dell'utente. La clausola di pacifico godimento afferma che l'utente può essere lasciato da solo per usare il prodotto per i propri scopi. Negandolo, di fatto Microsoft sostiene che l'utente non può fare quello che preferisce, anche senza infrangere alcun punto della licenza, ma può unicamente fare quello che a Microsoft vada bene che faccia.

[http://www.cybersource.com.au/about/comparing\\_the\\_gpl\\_to\\_eula\\_it.pdf](http://www.cybersource.com.au/about/comparing_the_gpl_to_eula_it.pdf)

#### **- Cosa si può fare con un software opensource**

- Per riprendere i concetti in modo più sintetico, con un software Opensource o, per dirla più correttamente, soggetto alla licenza GPL o una analoga, è possibile usarlo come meglio si crede, per diffusione, numero di computer o obiettivi di utilizzo. E' possibile copiarlo e ridistribuirlo, o è possibile modificarlo, o pagare chi lo fa per noi, e redistribuire il prodotto modificato, purché questo sia a sua volta soggetto alla stessa licenza.

Il principio è particolarmente vantaggioso anche per le aziende, perché può provare o usare tutto il software che desidera in modo libero e completo, e può modificare a proprio piacimento qualsiasi software per adattarlo al meglio alle proprie esigenze aziendali. Molto probabilmente l'ammontare di ore-uomo spese per la modifica del software avranno un costo molto inferiore al costo di una nuova licenza per un diverso prodotto, che sarà comunque non altrettanto adatto.

Inoltre il software può a sua volta non venir diffuso, in modo da preservare il segreto aziendale.

#### **4 - Conclusione - Perché usare Linux in soldoni**

##### **- Perché ti garantisce i diritti della GPL**

- Il software può essere acquistato al solo costo del supporto, usato interamente o in parte, redistribuito, modificato, e possono essere a loro volta redistribute le modificazioni.

##### **- Perché gratis**

- Permette di avere tutto il software necessario, dai programmi di scrittura, a quelli di calcolo, a quelli per il multimedia, in modo assolutamente gratuito. Questo permette in casa di non vivere nell'illegalità, mentre in situazioni lavorative permette di abbattere

pesantemente i costi delle licenze del software, che sono solitamente singole e, anche se acquistate a gruppi con forti sconti, rappresentano pur sempre uscite per svariate migliaia di euro.

**- Perché è legale**

- In base a quanto precedentemente detto, l'uso di Linux in qualsiasi ambito, sia professionale che casalingo è assolutamente legale.

**- Perché garantisce la libertà dell'utente**

E' contro il potere, e non è politica questa. Ma è contro il potere perché garantisce che chiunque possa sviluppare come vuole per quel sistema. E l'utente sarà sempre libero di scegliere quello che preferisce, quello che si addice di più alle sue esigenze. Non esiste una multinazionale che arriva e dice come devono essere le cose.

Professionalmente è il massimo desiderabile, perché una qualsiasi azienda può adattare come meglio preferisce tutto il software che vuole alle proprie specifiche esigenze, anche per uso interno. Può inoltre provarlo, senza ricorrere ad acquisti di licenze di pacchetti probabilmente non adatti.

Condizioni d'uso del presente documento:

Questo documento è rilasciato secondo la licenza Creative Commons 2.5.

E' possibile copiare interamente o parzialmente tale documento per qualsivoglia scopo citando la fonte originaria, è possibile modificarlo in qualsiasi modo evidenziando in modo riconoscibile le modifiche effettuate.