

[N.d.T. : quella che segue è la traduzione in italiano di un vecchio articolo di Dominic Humphries <http://linux.oneandoneis2.org/LNW.htm> (adesso a questo link c'è una nuova versione di LNW)

All'epoca sul sito erano presenti, oltre all'originale in inglese, una serie di rimandi a traduzioni in lingue diverse, tra le quali mancava l'italiano. Pur non essendo un fenomeno con l'inglese, decisi di tentarne la traduzione. A fine lavoro, constatando che in effetti non era perfettamente aderente all'originale, abbandonai l'idea di segnalare all'autore l'esistenza della mia traduzione, che rimase in un cassetto dimenticato del mio hard disk. Rispuntato per caso di recente, considerato che l'articolo originale continua a piacermi e che lo reputo ancora piuttosto attuale, avrei deciso di pubblicarlo sul sito. Ovviamente il tipo di licenza è il medesimo dell'articolo originale "Creative Commons - Attribuzione - Non commerciale 2.5 - <http://creativecommons.org/licenses/by-nc/2.5/deed.it>]

Alessandro Crotti

#####

Nel seguente articolo mi riferisco al SO [\(1\)](#) GNU/Linux e a vari progetti di FOSS [\(2\)](#) con il termine generico di "Linux". E' esclusivamente per questioni di leggibilità....

LINUX NON E' WINDOWS !

Se, come me, passi una parte del tuo tempo su un Linux forum, sarai probabilmente disturbato dal numero di interventi che esordiscono con cose simili:

"Ciao! Uso Linux da alcuni giorni, ed è veramente grande! Ma, è un peccato che [questo o quest'altro] non funzioni come in Windows. Perchè gli sviluppatori non riscrivono completamente il software in modo che si comporti di più come in Windows? Sono sicuro che Linux conquisterebbe molti più utilizzatori di quanto faccia oggi!"

Potresti anche aver risposto a questi interventi, trovandoti coinvolto in un'animata discussione tra un "Linux newbie [\(3\)](#)" convinto che la sua idea, basata su anni di esperienza con un SO differente, più alcune ore con Linux, sia brillante e rivoluzionaria, e che a te non piace solo perchè sei un utente Linux "vecchia-scuola", convinto che la GUI [\(4\)](#) sia un parto del Demonio e che tutti dovrebbero essere obbligati ad utilizzare la CLI [\(5\)](#).

Questo articolo si propone di spiegare a quei newbies per quale motivo le loro idee tendono ad essere respinte piuttosto che condivise.

Partiamo con l'argomento più amato: "Se Linux facesse così, convertirebbe molte più persone sottraendole a Windows!"

Allora, lasciatemi spiegare una cosa fondamentale per capire Linux: la comunità Linux **NON** sta cercando di fornire all'utente medio di Windows un SO sostitutivo. L'obbiettivo di Linux non è "Linux in ogni computer".

Non è assolutamente così. Certo, sono entrambi SO. Possono entrambi essere usati per fare le stesse cose. Ma questo fa di Linux "**un'alternativa**", non "**un sostituto**". Sembrerebbe una distinzione insignificante, ma è di vitale importanza.

Linux <=> Windows come Motocicletta <=> Automobile:

Sono entrambi veicoli che ti portano da A a B su una strada, ma hanno forme differenti, dimensioni differenti, meccanismi di controllo differenti, e funzionano in modo totalmente diverso.

Non sono liberamente intercambiabili. Hanno diversi utilizzi e differenti punti di forza e di debolezza, e dovresti scegliere quello più appropriato, non sceglierne uno e aspettarti che possa fare tutto quello fa l'altro.

Qualcuno alla guida di un'autovettura potrebbe trovarsi imbottigliato nel traffico, e vedere un motociclista superarlo sfilandogli su un lato. Potrebbe invidiare la possibilità che hanno i motociclisti di ignorare questo problema così penalizzante per le macchine. Se quell'automobilista dicesse "*Io so tutto delle macchine, quindi so tutto anche delle motociclette!*" sarebbe in errore.

* Se quell'automobilista acquistasse una moto e trovasse di essere confuso dall'acceleratore controllato da una manopola rotante anziché da un pedale, potrebbe lamentare il fatto che le motociclette dovrebbero essere adattate con un pedale per il gas.

* Se quell'automobilista avesse una moglie e due figli, potrebbe considerare un difetto la capacità delle moto di trasportare solo un passeggero. Potrebbe suggerire di ricostruire le moto in modo che possano caricare quattro persone, affiancate due a due.

* Se provasse a fare un giro in moto, e cadesse perchè non è abituato ad utilizzare il proprio peso per stare in equilibrio, potrebbe suggerire che le moto vengano ridisegnate con quattro ruote.

* Se si trovasse a sbilanciarsi facendo una curva, potrebbe suggerire di fornire le moto di barre stabilizzatrici per stare agevolmente in posizione verticale durante le curve.

* Se volesse proteggere la sua motocicletta dai furti, lamenterebbe il fatto che non ci sono portiere per chiudere all'esterno dei potenziali ladri, il che rende il suo nuovo mezzo di trasporto troppo semplice da rubare.

* Se trovasse ingombrante il casco, potrebbe suggerire la dotazione di un airbag nel manubrio, in alternativa a quel fastidioso accessorio

E in tutti questi casi sarebbe in errore. Perchè penserebbe che una motocicletta possa sostituire un'automobile, penserebbe che possa e debba fare tutto quello che una macchina è in grado di fare. Penserebbe che funzioni come una macchina e che quelle caratteristiche mancanti, possano semplicemente essere aggiunte.

Allo stesso modo, dei ben intenzionati nuovi utenti di Linux, danno suggerimenti su come rendere Linux più simile a quello che loro sono abituati a usare. Perdonano tempo, per le stesse ragioni.

Linux e Windows possono essere utilizzati per gli stessi scopi, ma esattamente come le automobili e le motociclette non significa che si possa passare direttamente da uno all'altro, e non significa che le caratteristiche di uno possano essere passate sull'altro

Molte persone pensano che migrare da Windows a Linux sia come passare da una BMW a una Mercedes. Pensano che i controlli dovrebbero essere gli stessi, che la loro esperienza si possa trasferire direttamente, e che le differenze siano in massima parte una

questione di estetica.

Loro pensano: "Io ho bisogno di una macchina per utilizzare la strada e ho bisogno di un SO per utilizzare un computer. Le macchine funzionano tutte allo stesso modo, quindi i SO devono funzionare tutti allo stesso modo." Ma questo è errato. "Io ho bisogno di un veicolo per utilizzare la strada e ho bisogno di un SO per usare un computer. Io so come guidare un'automobile, ma sono ignorante riguardo alle motociclette. Io so come usare Windows, ma sono ignorante riguardo a Linux" - **Questo è corretto.**

Un utente Windows deve capire che è solo un utilizzatore esperto di Windows, non è un utilizzatore esperto di computer; così come un automobilista ha esperienza solo come guidatore di automobili, non come guidatore "di-tutti-i-veicoli-da-strada". Un utente Windows deve realizzare che con Linux è tornato ad essere un dilettante, come un automobilista con la motocicletta. Un utente Windows deve imparare che ci sono diversi sistemi per portare a conclusione lo stesso compito, così come l'automobilista deve abituarsi a un manubrio in sostituzione di un volante, e ad usare un casco anche se non lo aveva mai fatto prima. E devono entrambi prepararsi ad accettare che "differente" non significa "inferiore".

Questo semplice fatto crea grandi difficoltà ai più affermati utenti Windows. Loro arrivano a Linux con diverse abitudini radicate e un atteggiamento del tipo: "Grazie, ma io so esattamente come usare un computer!"

Il problema è che non lo sanno. Loro sanno solo come usare Windows. Quando passano a un diverso SO, questi "power users" possono essere quelli con i maggiori problemi, in quanto hanno più cose da "disimparare".

Mentre un newbie dice semplicemente "Non so" e inizia a documentarsi e chiedere sui forum, il "Windows Power User" dice "Io so come si fa questa cosa, si fa così, così, così, e . . . Non funziona! Stupido SO!" E quindi dicono "Se io, con tutta la mia esperienza, non sono in grado di farlo funzionare, per un newbie non c'è speranza! Linux non è ancora pronto per l'uso desktop!".

Non riescono a realizzare che tutta la loro esperienza **sta lavorando contro di loro**, causandogli più problemi che a un utente con meno esperienza. Commettono l'errore di pensare a Linux come a **un software diverso per fare le stesse cose** di Windows, mentre è **un software diverso che fa cose diverse**. Non sta facendo un cattivo lavoro per svolgere lo stesso compito, sta facendo un buon lavoro per svolgere un compito differente.

Linux è un'alternativa a Windows, non un sostituto. E non lo sarà mai, perchè ha obiettivi incompatibili. L'obiettivo della Microsoft è di mettere il proprio software sul maggior numero possibile di PC, in quanto ha come priorità il profitto. Linux ha obiettivi diversi, perchè è libero. Ha differenti priorità.

Capire questo è capire l'essenza del FOSS. E' perfettamente comprensibile che un Linux newbie non lo capisca ancora, è una novità per lui. E' abituato a pensare in termini di software proprietario. Lasciate quindi che lo spieghi:

Il tipico FOSS è creato da qualcuno che, non trovando un programma di suo gradimento, decide di farselo per conto suo. Poi, siccome è un bravo ragazzo, ne divulga il codice sorgente dicendo al mondo: "Prego, servitevi!". Egli può fare questo, perchè non costa nulla duplicare il software, e quindi non gli costa nulla di più donarlo al mondo intero piuttosto che tenerlo solo per sè.

Peraltro, è importante ricordare che non avrà neppure alcun beneficio donando il suo

programma. Che venga utilizzato da una persona o da un miliardo non fa differenza per lo sviluppatore. Certo, ha la soddisfazione di sapere che ha creato un prodotto popolare; il numero di utilizzatori può essere una bella spinta per il proprio ego. Un modo, se volete, di acquisire punti. Ma non gli porterà alcun guadagno in denaro: E' FOSS.

Se il software è un successo, altre persone si interesseranno al progetto, e daranno il loro contributo per migliorarlo. Questo è il più grande vantaggio del FOSS : ogni utilizzatore è un potenziale sviluppatore. Chiunque può partecipare e dare il suo contributo perchè il programma funzioni meglio, faccia più cose, e contenga meno bachi. E' un bene che un programma attragga una comunità di programmatori. Ma è un bene per il programma, perchè lo rende migliore. Non rende più ricchi i programmatori.

Il FOSS è l'esatto contrario del software proprietario come Windows: nel FOSS l'obbiettivo è il software, non il numero di utenti finali. Un programma che lavora bene ma ha solo pochi utilizzatori, è considerato un fallimento per gli standard commerciali, mentre è un successo per quelli del FOSS.

Fare FOSS significa fare software di qualità, software utile. Se vuoi usarlo, ci si aspetta che tu investa del tempo per imparare "come usarlo".

E' stato creato e ti è stato dato gratuitamente, da persone che ci hanno investito molto del loro tempo, senza fini di lucro personale. Il minimo che tu possa fare per ripagare il loro contributo, è investire un po' del tuo tempo prima di lamentarti del fatto che non funziona come i programmi di Windows.

"Eh no carino, aspetta un momento..." dice il newbie sogghignando. "Esistono dei progetti Linux che hanno come obbiettivo di rimpiazzare Windows, e non solo di fornire un'alternativa!"

E' facile capire da dove venga questa idea. KDE e Gnome, per esempio, forniscono un ambiente desktop che è molto più "Windows-like" dei tipici "window managers" di Linux e della CLI.

Linspire è una distribuzione basata quasi interamente attorno all'idea di rendere Linux più "Windows-like".

Ciononostante, paradossalmente, questo prova la mia tesi più di quella del newbie. Perchè? Perchè questi sono dei normali progetti di FOSS, interamente indirizzati a creare del software di qualità. L'unica differenza, è che una delle definizioni di "qualità" è : "quanto potrebbero essere usati agevolmente da un utilizzatore di Windows?" .

Si tratta quindi di tipici progetti Linux al 100%, con il solo intento di fare del buon software. Gli sviluppatori di questi programmi sono animati da uno spirito ancora più altruistico, in quanto non stanno facendo un programma a loro uso e consumo. Stanno invece creando programmi interamente a beneficio di altre persone: programmi che rendano più semplice la transizione da Windows a Linux.

Questi programmatori, consci del fatto che ci sono molti utenti Windows che vorrebbero passare a Linux, investono buona parte dei loro sforzi per creare un ambiente in Linux avente un aspetto confortabilmente familiare per un utilizzatore di Windows. Ma **non** lo fanno per creare un sostituto di Windows, anche se il risultato potrebbe dare questa impressione. E' l'obbiettivo a cui mirano che fa la differenza. L'obbiettivo non è fare un sostituto di Windows; l'obbiettivo è rendere più agevole la transizione a Linux per il tipico utente Windows.

Non è raro incontrare dell'ostilità da parte della comunità Linux nei confronti di questi progetti. Talvolta per comprensibili motivi razionali ("KDE è un divoratore di risorse, quindi usate Fluxbox"), altre volte sembra più un atteggiamento di irrazionale ostilità del genere "Il software che assomiglia a Windows non è buono". In realtà, non si tratta di un atteggiamento anti-Microsoft o anti-Windows. E' invece un comprensibile scarso apprezzamento per quello che non si ritiene necessario.

L'utilizzatore 'tipico' di Linux è un hobbysta: usa il computer perchè è divertente. Programmare è divertente, fare hacking [\(6\)](#) è divertente. E Linux di gran lunga il miglior SO per un hobbysta che voglia fare hacking. Puoi smontarlo fino ai suoi componenti fondamentali, e riassemblarlo per adattarlo alle tue esigenze specifiche.

Ciononostante, l'influsso corrente di utenti Linux, ha una larga percentuale di non-hobbysti non-hackers. Questi cercano un computer che **semplicemente funzioni**, un computer che funzioni come Windows. Non sono interessati a spendere del tempo per modificare Linux secondo le proprie esigenze. Vogliono che funzioni così com'è.

E questo è comprensibile, ma per la percezione di un tipico Linux-user, è come se qualcuno comprasse una macchinina Lego pre-assemblata, con mattoncini incollati tra loro in modo irreversibile. Lo trovano incomprensibile. L'unico tipo di reazione che possono avere è esclamare perplessi: "Ma per quale motivo si può volere una cosa del genere?"

Se vuoi una macchinina già costruita, compra un modellino. Se vuoi una macchinina da costruire, compra una Lego. Ma per quale motivo potrebbe qualcuno volere una macchinina Lego, che può essere usata solo come un automodello? Il bello dei Lego è che puoi smontarli e ricostruirteli da solo!

Questa è la reazione di un tipico Linux-user alla brigata dei "non può semplicemente funzionare?": "se vuoi che semplicemente funzioni, usa Windows. Se ti interessa l'hacking, usa Linux. Per quale motivo vuoi passare a Linux se non ti interessa trarre vantaggio dalla sua natura "open source"? [\(7\)](#)"

La risposta, generalmente, è che non vogliono **realmente** passare a Linux. Loro vogliono semplicemente **abbandonare** Windows: stanno scappando dai virus, dai malware [\(8\)](#), stanno cercando di superare le restrizioni a cui sono soggetti da parte del loro software "acquistato"; stanno cercando di sottrarsi all'E.U.L.A. [\(9\)](#). Non stanno cercando di **entrare in Linux**, stanno cercando di **uscire da Windows**. E Linux è semplicemente l'alternativa più conosciuta.

Di questo parleremo ancora più avanti. . .

Potresti pensare: "OK, questo spiega per quale motivo gli sviluppatori non si impegnano per fare in modo che il loro software funzioni come Windows. Ma sicuramente il software Linux potrebbe essere fornito di una GUI in stile "Windows-user-friendly" senza che questo interferisca con i principi del FOSS "

Ci sono alcune ragioni per cui questo non si verifica.

Primo: Pensi veramente che qualcuno abbia creato un programma con un'interfaccia utente deliberatamente brutta?

Quando qualcuno regala una parte consistente del proprio tempo alla creazione di un programma, cercherà di dargli l'interfaccia utente (UI) migliore possibile. L'UI è una parte molto importante del software: non c'è modo di sfruttarne le funzionalità se non accedervi tramite l'UI. Puoi non sapere quale sia, ma c'è sempre un motivo per cui l'UI è fatta così. Qual'è? Perchè è la migliore UI che il creatore potesse creare.

Prima che tu insista a dire che un'UI più Windows-like migliorerebbe il software, considera questo: il creatore del software, che per definizione conosce meglio di chiunque altro il programma, non è d'accordo con te. Potrebbe anche sbagliare, ma le probabilità sono a suo favore.

Secondo: Linux è già fornito di GUI user-friendly. Non riesco a pensare ad alcuna funzione, non importa quanto ad alto livello, che non possa essere controllata per mezzo di una GUI. Puoi compilare il kernel (make xconfig), configurare il firewall (fwbuilder), partizionare l'hard disk (qtparted). . . c'è tutto, grazioso, interattivo, intuitivo e user-friendly.

Ma il 'ciclo di realizzazione' in Linux non è come in Windows. Non avrai la realizzazione di una "definitiva e pulita" GUI dall'inizio. La GUI aggiunge solo complessità e nessuna funzionalità al software. Uno sviluppatore non si siede a progettare una graziosa GUI che non fa niente, si siede e progetta un programma che faccia ciò di cui lui ha bisogno.

La prima cosa che un programma deve fare, è essere usabile dall'interfaccia a riga di comando (CLI). Sarà probabilmente eseguibile con ogni sorta di opzione e utilizzerà un prolisso file di configurazione. E' così che inizia, perchè **è la funzionalità** ad essere richiesta. Tutto il resto viene dopo. E se anche un giorno avrà una GUI, è importante ricordare che potrà sempre essere completamente controllato dalla CLI e dal file di configurazione.

Questo perchè la CLI ha diversi vantaggi:

la CLI è universale. Ogni sistema Linux ha la CLI. Ogni file eseguibile può essere lanciato dalla CLI. E' facile lavorare con un software da remoto tramite la CLI.

Nessuna di queste cose è vera per la GUI:

alcune macchine Linux non hanno il sistema X11 di gestione della sessione grafica installato; Alcuni programmi non hanno una GUI; Alcuni software non sono raggiungibili dai menu delle GUI; Spesso non è semplice o pratico utilizzare una GUI da remoto.

Per ultimo, ci possono essere differenti GUI per svolgere lo stesso lavoro, e non c'è modo di sapere quale di queste sia installata sul tuo computer.

Quindi, ricorda, se chiedi "Come posso fare. . . ?", ti verrà probabilmente risposto come farlo tramite la CLI. Questo non significa che puoi farlo solo tramite la CLI. Semplicemente riflette l'importanza che hanno la CLI e la GUI relativamente allo sviluppo di un software.

* Windows è completamente GUI-centrico. E' un SO basato sulla GUI, con una orribile (ma prossima al miglioramento) CLI. Non c'è quasi nessun software per Windows senza una GUI. Questo tende a far pensare alle persone che la GUI sia una parte integrante e vitale del software.

Ma in Linux, un programma viene rilasciato non appena è funzionante. Solo dopo che è diventato stabile, ragionevolmente privo di bachi, e ricco di funzioni, come valore aggiunto viene addizionato di una GUI.

Potete pensare a un programma senza una GUI come a "un'anteprima" piuttosto che ad una versione definitiva del prodotto. Un FOSS raramente si può definire 'finito', viene sempre migliorato. Potrà esserci un giorno in cui sarà reso user-friendly. Ma per la maggior

parte del tempo, è più importante che **"lavori bene"**, piuttosto che **"sembri bello"**. Il FOSS è più un viaggio che una meta.

L'ultima cosa da tenere a mente è che la GUI di un programma sarà con ogni probabilità un programma a se stante. Potrà anche essere sviluppato in modo completamente indipendente dal software originale, da sviluppatori completamente diversi. Se vuoi una GUI per un programma, non è improbabile che tu la ottenga con una installazione separata, piuttosto che con una unica assieme al programma sottostante.

Nonostante tutto questo, come ho detto, già adesso in Linux puoi fare pressochè tutto per mezzo di una graziosa GUI, come in Windows. Devi solo ricordare che la GUI è l'ultimo passo, non il primo.

Linux NON mette la forma prima della sostanza.

Terzo: Linux è deliberatamente disegnato per l'utente bene informato, piuttosto che per l'ignorante newbie. E questo per due motivi:

* L'ignoranza fa comodo, ma ha vita breve. La conoscenza è eterna. Potranno volerci giorni, settimane, o mesi perchè il tuo livello di conoscenza passi da "Linux newbie" a "utente Linux medio", ma una volta giunto a quel punto, hai anni di utilizzo di Linux davanti a te.

Aggiungere migliaia di righe di codice, per rendere il software più semplice da usare per i nuovi arrivati, sarebbe come dotare permanentemente di rotelle laterali tutte le biciclette. Questo potrebbe facilitare all'inizio, ma poi. . . ?

Non ti compreresti una bicicletta con le rotelle adesso, ne sono sicuro. E non perchè sei un tipo originale a cui non piacciono le cose semplici, ma perchè le troveresti inutili, come sono inutili per chiunque, tranne per chi sta imparando.

* Non importa quanto sia valido un software, è valido esattamente quanto il suo utilizzatore. La porta più sicura del mondo non rappresenta una barriera per i ladri, se lasci la chiave nella serratura o la finestra aperta. Il motore più perfezionato del mondo non andrà molto lontano se metti gasolio nel serbatoio al posto della benzina.

Linux mette tutto il potere nelle mani dell'utente. Questo include il potere di danneggiare irreparabilmente lo stesso SO. L'unico modo per far lavorare Linux come si deve, è imparare quanto basta per sapere quello che stai facendo. Rendere più semplice per un utente l'uso di funzionalità che non comprende, vorrebbe dire aumentare le probabilità che rompa qualcosa accidentalmente.

Quarto: Hai letto in un punto qualunque di tutto il testo precedente, che il FOSS avrebbe dei benefici ad attrarre un alto numero di classici utenti Windows?

Fai con calma. Ri-leggi tutto, se vuoi. Posso aspettare.

Il principio guida di Linux e del FOSS è "fare buon software". Non è "Fare un software che sostituisca Windows". L'unico contributo che potrebbe dare a Linux un'orda di classici utilizzatori di Windows, sono le lamentele. Di cosa si lamenterebbero? "*...ma non funziona così in Windows...*"

No, non funziona così. Se funzionasse come Windows, Linux farebbe cagare. Sarebbe una copia mal fatta che nessuno vorrebbe usare. Il motivo per cui le persone adorano Linux, è che non funziona come Windows!

Non fa tutto al posto tuo, non ritiene che tu rimanga per sempre un newbie ignorante,

non ti nasconde quello che sta facendo al suo interno.

Windows è come un autista che ti scarrozza in giro; Linux ti mette in mano le chiavi facendoti sedere al posto di guida. **Se non sai guidare è un problema tuo.** E anche la colpa è tua. Ci sono un sacco di persone che ti aiuteranno a imparare se lo chiedi. E se suggerissi di fornire il sistema di una specie di pilota automatico, avresti una risposta del tipo: "*mmh...questo potrebbe agevolare l'utilizzatore, ma ti perdi il meglio della guida*". Ma sarai sicuramente maltrattato se cerchi di convincere chiunque, che ciò di cui Linux ha *veramente* bisogno è uno chauffeur.

"Ma questo renderebbe Linux un fenomeno di massa!", lamenta il newbie.

Può darsi. Ma quanti sviluppatori trarrebbero un vantaggio da ciò? Linux è gratuito [N.d.T.(10) -as in beer- nella versione originale). Nessuna delle persone che hanno creato Linux trarrebbe un profitto da un più ampio bacino di utenza. Nessun partecipante ai Linux forum trarrebbe un profitto da un più ampio bacino di utenza. Non è intenzione di Linux raggiungere un più ampio bacino d'utenza - questo è un obiettivo del software commerciale.

L'obiettivo di Linux è fare un SO veramente valido. Gli sviluppatori sono impegnati ad aggiungere funzioni, rimuovere difetti, e migliorare le implementazioni esistenti. Non sono impegnati ad affiggere manifesti che pubblicizzano "quanto siano buoni i loro prodotti". Questo dovrebbe dirti quali sono le loro **priorità.**

E guarda, grazie a questo atteggiamento, cos'è successo al numero di utenti. Linux è nato piccolissimo ed è diventato enorme. Il motivo per cui è diventato così popolare? Perché ha sempre puntato alla qualità. Gli utenti attratti da Linux sono quelli che cercano la libertà e la qualità che solo un FOSS può dare loro. Linux è diventato grande perché non si è mai curato di quanto grande avrebbe potuto diventare. L'obiettivo degli sviluppatori è solo farlo funzionare, e funzionare bene, così hanno attratto gli utenti che cercano un SO che funziona, e che funziona bene.

Gettare improvvisamente via tutto, per concentrarsi invece sulla realizzazione di un sostituto di Windows, significherebbe uccidere la cosa che ha fatto di Linux quello che è. Ci sono delle società là fuori che vedono Linux crescere, e vorrebbero guadagnarci sopra. Sono frustrate dalla GPL(11) che gli rende estremamente difficile vendere Linux ai prezzi della Microsoft. "*Linux morirà se rimane aperto...*" dicono, "*perché nessuno ci può guadagnare, così com'è.*"

Non realizzano che trasformare Linux in un software proprietario, sarebbe come uccidere la gallina dalle uova d'oro. Linux è diventato grande perché è un FOSS, e nessuno a cercato di farne un sostituto di Windows. Linux prospera perché combatte Windows su fronti nei quali non potrà mai essere sconfitto: **l'apertura e la qualità.**

Per molti utenti Windows, Linux è una copia inferiore di Windows. Ha funzionalità meno evidenti, minore integrazione, ed è molto più complesso. Per quel tipo di utenti, Linux è un brutto SO. Ed è effettivamente così, perché non incontra le loro necessità. Loro hanno bisogno di un SO molto semplice da usare, che faccia tutto senza che tu debba imparare nulla.

Windows è creato per utenti non-tech. La percezione diffusa presso questi utenti è che Linux sia difficile da usare. In effetti non è così, ma è un fraintendimento comprensibile.

Linux è facilissimo da usare. Ve lo assicuro, è veramente facile da usare. Qual'è il motivo per cui non è percepito così? Perché il concetto "facile da usare" è stato brutalmente distorto. Nell'uso comune "facile da usare" significa "qualcosa di facile da fare senza avere a priori la minima idea di come si faccia". Ma questo non è "facile da usare". questo è "facile da indovinare".

E' come la differenza fra :

* Una cassaforte sulla quale c'è un avviso con scritto: "*puoi aprire questa cassaforte ruotando la manopola fino al 32, poi a 64, poi a 18, poi a 9, quindi gira la chiave e alza la leva*".

oppure:

* una macchina che può essere aperta premendo un bottone su un comando a distanza.

E' più semplice aprire la macchina, vero? Premere un bottone da qualunque posto in prossimità del veicolo, invece di numerose specifiche rotazioni di una manopola numerata. Ma cos'è più semplice per chi non ha assolutamente idea di come aprire ne' una macchina ne' una cassaforte? La cassaforte ha delle chiare istruzioni su di essa, mentre per la macchina ci sono solo dei bottoni, che tra l'altro non sono neppure attaccati alla macchina.

Per Linux è la stessa cosa. **E' facile da usare, ma non è sempre facile da imparare**. Solo se hai intenzione di investire il tuo tempo per imparare lo troverai facile.

A titolo di esempio prendete questo esercizio arbitrario: Volete spostare cinque linee dal centro di un documento di testo, alla fine dello stesso.

Con MS Word, MS WordPad, o MS Notepad, tutti editor "user-friendly" di Windows, il sistema più veloce è il seguente:

- Shift-Down
- Shift-Down
- Shift-Down
- Shift-Down
- Shift-Down
- Ctrl-X
- Ctrl-End
- Ctrl-V

(Questo assumendo che utilizzate la tastiera. Altrimenti con il mouse serviranno una serie di Click-e-Trascinamenti più un autoscroll affidabile.)

Con **vi**, invece, è il seguente:

- d5d
- Shift-g
- p

(O se conoscete veramente **vi**, basterà ":1,5m\$" per fare la stessa cosa!)

Vi batte i programmi Microsoft alla stragrande! Com'è possibile? Perché **vi** è stato progettato **per essere funzionale**, mentre la Microsoft progetta **per essere user-friendly**. La Microsoft divide ogni lavoro in tanti semplici passi, così saranno necessari molti più passi per concludere lo stesso compito.

Questo fa di **vi**, il text-editor più veloce e semplice da usare per qualunque tipo di lavoro. Ovviamente solo se sai come usarlo. Nel caso non lo sapessi "d5d" vuol dire "Metti cinque righe del testo nella memoria temporanea, quindi cancellale dal documento", e questo è un esempio di come devi lottare per fare in modo che **vi** funzioni. Ma dopo che lo hai appreso, puoi volare lontano col minimo sforzo.

Così, quando un newbie vede quanto possa fare le cose facilmente e velocemente un utilizzatore esperto di **vi**, non può non essere d'accordo sul fatto che **vi** è superiore a Word nell'elaborazione di testi. Quindi prova a usarlo anche lui. Lo fa partire e si trova di fronte a uno schermo pieno di "~" . Allora prova a digitare qualcosa, ma sullo schermo non compare nulla.

Finalmente scopre che esiste una **modalità comandi** e una **modalità per l'inserimento di testo**, così inizia ad usare **vi** con una conoscenza approssimativa delle sue funzioni. E fatica a fare tutto ciò, perché ci sono così tante cose da imparare per riuscire a far funzionare **vi** come si deve. Ed ecco che comincia a lamentarsi "vi sarebbe migliore se fosse facile da usare come Word!"

Ma in realtà dovrebbe dire: "Io non so come usare vi e non posso perdere tempo ad impararlo." Ma questo significa che **è un suo problema**, che non può essere imputato al programma. Secondo lui non importa se migliaia di persone stanno felicemente usando **vi** senza problemi: ...è troppo difficile da usare e deve essere modificato!

E credetemi, se lui fosse in grado di creare un text-editor "user-friendly" come Word e funzionale come **vi**, sarebbe sommerso dagli applausi. Probabilmente riceverebbe una nomination per il Premio Nobel per la Genialità, in quanto nessun altro è mai stato in grado di farlo in precedenza. Ma lamentarsi del fatto che **vi** è difficile da usare, lo fa invece sommergere dalle risate, **perché il problema non è vi, è lui**.

E' come acquistare un pennello di Leonardo da Vinci e lamentarsi del fatto che ciononostante non si è in grado di dipingere. Non è stato il pennello a dipingere la Gioconda, ma la capacità dell'artista. Il pennello è uno strumento che si affida alle capacità dell'utilizzatore. E non c'è altro modo per acquisire quelle capacità se non con la pratica.

Con **vi** è uguale. Ed è la stessa cosa con molti software per Linux che i nuovi utenti lamentano essere "troppo difficili" o "non sufficientemente intuitivi", siano questi editor di testo, gestori di pacchetti, o la stessa linea di comando.

Prima di insistere a dire che qualcosa in Linux deve essere modificato, c'è una domanda importante da porsi: "Gli utenti esperti hanno dei problemi?"

La risposta è "**No**", quindi è un "**tu**" problema. Se le altre persone lo usano con profitto, perché tu non riesci? Ti sei preso il tempo necessario per imparare a usarlo? O ti aspetti che funzioni semplicemente dicendo 'abracadabra'?

"User-friendly" e "funzionalità" sono caratteristiche contrapposte. Tutti quei bottoncini e quei menu a cascata, vitali per rendere un software "facile da usare", sono una spina nel fianco per l'utente esperto. E' la differenza tra navigare da A a B usando una mappa e una bussola, e andare da A a B seguendo una serie di segni direzionali sulla superficie stradale: Chiunque può farlo seguendo i segni, ma impiegherà il doppio del tempo di chi sa come arrivarci "direttamente" .

Se voglio incollare il valore di una formula in Excel, lo devo fare passando attraverso i menu Modifica->Incolla Speciale->Valori . Io non voglio impiegare il mio tempo attraversando tutti questi 'amichevoli' menu, sotto-menu e finestre di dialogo del cavolo. Io voglio semplicemente farlo.

E, per essere onesti, se riprogrammo i tasti funzione e registro qualche macro, posso ottenere che Excel e Word facciano più cose premendo un pulsante.

Ma questo non è essere user-friendly. Questo richiede all'utente di investire molto tempo nel software. Linux ti richiede di investire il tuo tempo **nell'imparare ad utilizzare le funzionalità esistenti**. Il software "User-friendly" ti spinge a impegnare il tempo **per creare nuove funzionalità**.

Se questa è la via che preferisci, OK, vai avanti così. Ma almeno sii onesto e confessa che il problema sta nella tua ignoranza e non nel software. Tutto il software Linux è estremamente semplice da usare, quando hai imparato come funziona. Se non impari, non sarà mai semplice, ma non per colpa del software.

Ora, potresti cominciare a pensare che Linux abbia un problema caratteriale. Non ama gli utenti, non vuole rendere la vita facile agli utenti . . . è solo per degli snob l33t h4xx0r5! [\(12\)](#)

Nulla è più lontano dalla realtà. Certo che Linux vuole gli utenti! E sicuramente non vuole rendergli la vita più difficile. Ma devi realizzare che le sue definizioni di "**utente**" e "**facile o difficile**" possono essere differenti dalle tue, e differenti dalla cultura del software proprietario.

Linux vuole gli utenti che vogliono Linux. E intendo con tutto quello che comporta. Il libero, open-source software; la capacità di effettuare modifiche e personalizzazioni del tuo software; essere al posto di guida avendo il controllo totale.

Ecco esattamente cos'è Linux. Le persone migrano a Linux perchè sono stufi dei virus, stufi dei BSOD [\(13\)](#), stufi degli spyware [\(14\)](#). Questo è comprensibile. Ma queste persone non vogliono Linux. Loro in realtà vogliono semplicemente Windows senza i suoi difetti. Loro non vogliono "**realmente**" Linux. Quindi perchè Linux dovrebbe volere loro?

Ma se queste persone provano Linux come reazione ai virus e agli spyware, e successivamente capiscono di amare quest'idea di un SO che possono controllare totalmente. . . allora iniziano a volere Linux . Ed ecco che Linux inizia a volere loro..

Prima di decidere che vuoi passare a Linux, fatti questa domanda: "*Perchè voglio cambiare?*"

Se la risposta è "*Voglio un SO che metta il potere in mano al proprio utilizzatore e si aspetta che questi sappia come usarlo*": Passa a Linux. Dovrai investire una discreta parte del tuo tempo e dei tuoi sforzi per fargli fare quello che vuoi, ma sarai ripagato da un computer che fa esattamente quello che vuoi tu.

MA . .

Se la risposta è "Voglio Windows senza i suoi problemi", rifai una installazione pulita di Windows XP SP2, configura un buon firewall, installa un buon anti-virus, non usare mai IE(15) per navigare in internet, esegui regolarmente gli aggiornamenti, fai ripartire il computer dopo ogni installazione di software e tieniti informato riguardo ai problemi di sicurezza.

Io personalmente ho usato i SO Windows dal 3.1 passando per 95, 98, NT, e XP, e non sono mai stato infettato da virus, da spyware, ne' stato crackato. Windows può essere un SO sicuro e stabile, ma dipende da te renderlo così.

Se la risposta è "Voglio un sostituto di Windows senza i suoi problemi", compra un Apple Mac. Ho sentito cose magnifiche riguardo alla distribuzione Tiger di OS X, e in più usa dell'hardware esteticamente molto gradevole. Dovrai pagare per acquistare un nuovo computer, ma avrai quello che cerchi.

In entrambi i casi, non passare a Linux. Saresti deluso sia dal software che dalla comunità.

Linux non è Windows!

Note:

(1) SO : Sistema Operativo

(2) FOSS : Free and Open Source Software - Software Libero e Open Source

(3) newbie : neofita, novellino. Indica il nuovo arrivato. A seconda del contesto può essere utilizzato in senso affettuoso o dispregiativo.

(4) GUI : Graphical User Interface - Interfaccia Grafica. È lo strato di un'applicazione software che si occupa del dialogo con l'utente del sistema, utilizzando un ambiente grafico.

(5) CLI : Command Line Interface - Interfaccia a linea di comando. E' un termine generico per descrivere un programma che permette all'utente di interagire scrivendo righe di comando su un terminale testuale.

(6) fare hacking : Nonostante il termine sia spesso travisato dai mezzi di informazione, assegnandogli un'accezione negativa, la definizione prevalentemente utilizzata in ambito tecnico è "lo studio dei sistemi informatici al fine di potenziarne capacità e funzioni".

(7) open source : a sorgente aperto. Indica un software rilasciato con un tipo di licenza per la quale il codice sorgente è lasciato alla disponibilità di eventuali sviluppatori, in modo che con la collaborazione (in genere libera e spontanea) il prodotto finale possa raggiungere una complessità maggiore di quanto potrebbe ottenere un singolo gruppo di programmazione.

(8) malware : un qualsiasi software creato con il solo scopo di creare danni più o meno estesi al computer su cui viene eseguito.

(9) EULA : End User License Agreement - Contratto con l'utente finale. E' la licenza contenuta in tutti i software commerciali che l'utente finale deve necessariamente accettare per poter installare il programma.

(10) free as in beer : gratuito come la birra. L'autore si riferisce alla (famosa) definizione data dalla Free Software Foundation del "Free Software" : <<... is a matter of liberty, not price. To understand the concept, you should think of 'free' as in 'free speech', not as in 'free beer'... >> - << ...è una questione di libertà non di prezzo. Per capire il concetto dovete pensare a 'free' come alla 'libertà di parola' , non alla 'birra gratis'... >> . In italiano il problema non si pone, in quanto la parola 'free' nelle due accezioni viene tradotta in modi diversi (libero e gratuito). Nell'articolo l'autore si contrappone scherzosamente alla frase citata, per sottolineare che in questo caso si tratta di un software "gratuito" oltre che "libero".

(11) GPL : General Public License. Tipo di licenza che copre il software libero. Scritta da Richard M. Stallman e Eben Moglen nel 1989 nella sua prima versione per distribuire i programmi creati dal Progetto GNU. Contrapponendosi alle licenze per software proprietario, GPL permette all'utente libertà di utilizzo, copia, modifica e distribuzione del software.

(12) l33t h4xx0r5 : hackers elitari

(13) BSOD : Blue Screen Of Death. Identifica l'impietoso messaggio di un errore non meglio identificato (se non segnalando gli indirizzi di memoria e la libreria che lo ha causato) all'interno delle procedure di elaborazione di alcuni sistemi operativi.

(14) spyware : Uno spyware è un tipo di software che raccoglie informazioni riguardanti l'attività online di un utente (siti visitati, acquisti eseguiti in rete etc) senza il suo consenso, trasmettendole tramite Internet ad un'organizzazione che le utilizzerà per trarne profitto, tipicamente attraverso l'invio di pubblicità mirata.

(15) IE : Microsoft Internet Explorer. E' il browser Internet prodotto da Microsoft.