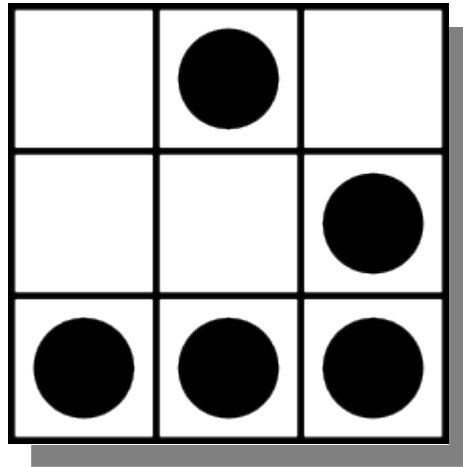


Diff. tra Windows e Linux



Premessa

Lungo questa presentazione, si prende come esempio di software proprietario il sistema operativo

Microsoft Windows®

ma i concetti espressi, valgono indipendentemente dal nome del software. Vogliamo confrontare due filosofie di sviluppo, due diversi modelli di business, prima ancora che due specifici sistemi operativi.

E' stato scelto Windows in quanto maggiormente conosciuto.

Che cos'è Linux?

Un kernel, il nucleo del sistema operativo.

- Il sistema operativo, in tutta la sua interezza (interfaccia grafica, riga di comando, compilatori, editor di testo...) si chiama GNU.
- Linux = accoppiata GNU + Linux
- GNU/Hurd, GNU/BSD, ...

Che cos'è GNU/Linux?

Un sistema operativo a se stante, esattamente come lo è Microsoft Windows®.

- Non si avvia “da windows” come gli altri cdrom.
- Non è un semplice programma, ma IL programma che “gestisce” gli altri programmi (e rende loro disponibili le risorse del sistema, in modo controllato).
- Se non ci fosse un sistema operativo, il computer non sarebbe molto più che una scatola di metallo
- GNU/Linux è un'alternativa a Microsoft Windows®. E' un suo concorrente.

Breve sommario

Ecco un breve sommario dei punti che andremo a toccare:

- Licenze utilizzate
- Differenze economiche
- Installare programmi
- Continuità del supporto e dei formati
- Completezza del sistema
- Robustezza del sistema
- Sicurezza del sistema
- Performance del sistema

Licenze utilizzate

L'utilizzo di un software proprietario, in genere, comporta il pagamento di una licenza d'utilizzo (più o meno costosa).

Questo essenzialmente si spiega con il modello di business che il software proprietario utilizza:

Il software è un **prodotto**, e come tale, va **venduto**.

Per poter vendere il software (e garantirsi dei ricavi adeguati all'investimento), le licenze proprietarie sono principalmente volte a tutelare l'autore del codice.

Licenze utilizzate

Ecco allora che ci viene vietato di copiare il software **da noi acquistato**, ci viene vietato di regalarlo, di modificarlo, di farci qualsiasi cosa che non sia il semplice utilizzo.

E' un po' come se fossimo in affitto.

Oltretutto, alcuni produttori di software, includono all'interno delle loro licenze voci piuttosto curiose.

Facendo riferimento alla licenza EULA (licenza utilizzata da Microsoft per il rilascio di innumerevoli suoi prodotti), andremo a leggere alcuni pezzi della licenza (dovreste averlo fatto tutti.....)

Licenze utilizzate

“Accetti che, al fine di proteggere l'integrità di contenuti e software protetti da DRM (“Contenuti Sicuri”), Microsoft possa preparare aggiornamenti di sicurezza per componenti del sistema operativo che saranno automaticamente scaricati sul tuo computer. Questi aggiornamenti di sicurezza potrebbero impedirti di copiare e/o riprodurre Contenuti Sicuri, e l'utilizzo di altro software sul tuo computer. Se Microsoft rilascerà questo genere di aggiornamenti, farà ragionevoli sforzi per mostrare, su un sito web, la ragione di questi aggiornamenti.”

<http://www.microsoft.com/windowsxp/home/eula.msp>

Licenze utilizzate

Con questa voce, la Microsoft si è arrogata il diritto di installare software sul vostro computer e fare “ragionevoli sforzi” per avvertirvi.

Concedereste a qualcuno un simile diritto? Non vorreste almeno avere la possibilità di scegliere cosa installare e cosa no?

Infondo il software l'avete pagato!

Non è più “come se fossimo in affitto”. E' molto peggio!
Non scelgo nemmeno cosa portare nel luogo dove vivo!

Licenze utilizzate

A differenza del software proprietario, GNU/Linux utilizza licenze libere.

La licenza più utilizzata, per il software libero (e quindi, per GNU), è la “GPL”, sigla che sta per:

GNU Public License

Questa, garantisce all'autore ed **al cliente finale** una serie di libertà.

Licenze utilizzate

All'autore: la libertà che il proprio software non venga “chiuso” (rubato) dal primo che passa.

Al cliente finale: la libertà di modificare, copiare, vendere, regalare, ridistribuire il software ricevuto, a patto che tutto questo venga fatto con una licenza libera.

La tutela dei diritti del cliente finale (oltre che i diritti dell'autore) porta a stravolgere il modello di business

Il software non si può più vedere come un prodotto, ma diventa uno **strumento**, tramite il quale, viene venduto del **tempo-uomo**, della **conoscenza**, un **servizio**.

Differenze Economiche

L'utilizzo di licenze libere, porta (anche se non necessariamente in modo diretto) tutta una serie di differenze sul costo del software.

Se io cliente, pur pagando per un software, sono poi autorizzato dalla licenza a ridistribuirlo gratuitamente, è difficile far pagare il software.

Quindi non obbligatorio che GNU/Linux sia gratuito, come è consuetudine dire

Ma nella pratica, la diffusione del software libero, porta come conseguenza il fatto che si possano trovare tutta una serie di distribuzioni GNU/Linux gratuitamente scaricabili da Internet e ridistribuibili.

Differenze Economiche

Windows XP Home Edition	270,00	EUR
Office 2003	536,48	EUR

806,48 EUR

Arch Linux	0,00	EUR
OpenOffice.org 2.0	0,00	EUR

0,00 EUR

Installare programmi

L'installazione dei programmi, è a lungo stata una delle barriere all'adozione del software libero da parte della massa (paura del terminale, confusione sui passi da eseguire, poca padronanza della tecnica)

L'adozione di licenze libere per rilasciare il software, porta inevitabilmente al proliferare di applicativi e formati diversi che, pur avendo lo stesso obiettivo, sono implementati in modo diverso.

Ecco perché, mentre su Windows, l'installazione di software è fatta più o meno sempre allo stesso modo (doppio click sull'installer, avanti, avanti, avanti, fine), su GNU/Linux (e più in generale su tutti i sistemi operativi con licenza libera), esistono modi diversi per ottenere lo stesso risultato.

Installare programmi

Possiamo dividere in due grandi categorie, i metodi di installazione di software sotto GNU/Linux.

- Installazione da sorgente
- Installazione tramite pacchetti precompilati
- Compilazione automatizzata

Installare programmi

L'installazione da codice sorgente (resa possibile solo dalla disponibilità del medesimo, e ciò esclude tutti i “pacchetti binari” rilasciati ad esempio da ATI o da nVIDIA) implica:

- Indipendenza dalla distribuzione
- Ricerca del sorgente (sf.net?)
- Minimale controllo delle dipendenze
- Ottimizzazione per il sistema
- Disinstallazione difficoltosa
- Difficoltà nel tenere traccia dei files installati

Installare programmi

In compenso, l'installazione da sorgente è in genere estremamente semplice:

- Download del sorgente (`wget http://...`)
- Decompressione (`tar -xzf filename`)
- Configurazione (`./configure`)
- Compilazione (`make`)
- Installazione nel sistema (`make install`)

Per tutto questo, sarà necessario utilizzare la riga di comando.

Installare programmi

L'installazione da pacchetti precompilati (appositamente per la distribuzione utilizzata) implica:

- Dipendenza dalla distribuzione
- Più semplice ricerca del pacchetto
- Poca ottimizzazione per il sistema
- Disinstallazione banale
- Controllo dei files installati
- Certezza dell'integrità del sistema

Installare programmi

A seconda del gestore di pacchetti (praticamente uno per distribuzione), possono essere aggiunte funzionalità diverse, quali il controllo delle dipendenze, o il loro reperimento automatico tramite la rete.

Ogni distribuzione (o quasi) ha il suo packet manager. Questo porta ad una quantità di formati diversi (rpm, deb, tgz...), il che moltiplica gli sforzi di chi li rende disponibili online (pacchettizzatori della distribuzione, sviluppatori del software, ecc ecc)

Installare programmi

D'altro canto, la personalizzazione del pacchetto, consente ad ogni distribuzione di “garantire” l'integrità del sistema una volta installato il pacchetto.

Infatti se chi prepara il pacchetto è anche chi prepara il sistema completo, saprà esattamente a cosa fare attenzione, e come non corrompere il sistema stesso.

Installare programmi

A seconda del sistema di pacchettizzazione, i comandi per effettuare l'installazione saranno quindi diversi, ma sempre più o meno banali:

RPM: `rpm -i nomepacchetto.rpm`

(Red Hat, Fedora, Suse, Mandrake, ...)

DEB: `apt-get install nomepacchetto`

(Debian e derivati, via rete)

TGZ: `installpkg nomepacchetto.tgz`

(Slackware e derivati)

Installare programmi

Alcune distribuzioni mettono a disposizione degli utenti particolari software che consentono di gestire in modo più intuitivo l'installazione e l'aggiornamento dei vari pacchetti che compongono il sistema.

Si tratta in genere delle distribuzioni più “user friendly”, quelle più volte agli utenti alle prime armi:

- Mandriva (nel Mandriva Control Center)
- SuSe (con Yast)
- Fedora
- ...

Filter: Selections

Selection

- Graphical Base System
- KDE Desktop Environment
- All of KDE
- Gnome system
- Help & Support Documentation
- Office Applications
- Games
- Multimedia**
- Simple Webserver
- LDAP Server and Tools
- Network/Server
- C/C++ Compiler and Tools
- Advanced Development
- Tcl/Tk Development System


Package	Avail. Ver.	Inst. Ver.	Summary
<input checked="" type="checkbox"/> ImageMagick	5.4.7-71	5.4.2-96	Image / film viewer and co
<input checked="" type="checkbox"/> SDL	1.2.4-167	1.2.3-167	Simple DirectMedia Layer
<input checked="" type="checkbox"/> aalib	1.2-673	1.2-534	AA-lib - an ascii art libr
<input type="checkbox"/> alevt	1.6.0-48		Teletext/Videotext decode
<input checked="" type="checkbox"/> alsa	0.9.0.cvs20020903-9	0.9.0_cvs20020320-12	Advanced Linux Sound Ar
<input checked="" type="checkbox"/> alsa-tools	0.9.0rc1-44		Various ALSA Tools
<input checked="" type="checkbox"/> arts	1.0.3-96	1.0.0-2	Modular software synthes
<input type="checkbox"/> atk	1.0.3-42		atk - accessibility toolkit
<input checked="" type="checkbox"/> audiofile	0.2.3-154	0.2.3-52	Audio File Library
<input checked="" type="checkbox"/> broadcast2000	1.0c-367		Broadcast 2000 is a video
<input checked="" type="checkbox"/> cdcons	2.4-145		Console CD Player
<input checked="" type="checkbox"/> cdda2wav	1.11.a28-22		CD-grabbing tool
<input checked="" type="checkbox"/> cdparanoia	IIIalpha9.8-271	IIIalpha9.8-180	CD-audio grabber for prot
<input checked="" type="checkbox"/> cdrdao	1.1.5-339		Write CD-R's in disk-at-o
<input checked="" type="checkbox"/> cdrecord	1.11.a28-22	1.11.a13-48	Tool for writing CDR's
<input checked="" type="checkbox"/> cups-libs	1.1.15-44	1.1.12-69	libraries for CUPS
<input checked="" type="checkbox"/> curl	7.9.8-38		Tool for Transferring Data
<input checked="" type="checkbox"/> dap	2.1.2-373		Digital Audio Processor
<input checked="" type="checkbox"/> dialog	0.62-601	0.62-509	Menus and input boxes fo
<input checked="" type="checkbox"/> esound	0.2.28-45	0.2.23-62	A sounddaemon for Enlight
<input checked="" type="checkbox"/> fam	2.6.9-34	2.6.7-53	file alternation monitoring
<input checked="" type="checkbox"/> fftw	2.1.3-619		Discrete Fourier Transfor

Description | **Technical Data** | Dependencies | Versions

ImageMagick - Image / film viewer and convertor

ImageMagick is a robust collection of tools and libraries to read, write, and manipulate an image in many image formats including popular formats like TIFF, JPEG, PNG, PDF, PhotoCD, and GIF. With ImageMagick you can create images dynamically, making it suitable for Web applications. You can also resize, rotate, sharpen, color reduce, or add special effects to an image and save your completed working the same or differing image format. Image processing operations are available from the command line, as well as through C, C++, and PERL-based programming interfaces.

Disk space

 42%

Check Dependencies Auto check Cancel Accept

Installare programmi

Esiste poi la possibilità di avere una compilazione automatizzata, che ai vantaggi della compilazione aggiunge quelli della pacchettizzazione.

Alcune distribuzioni mettono a disposizione questo metodo di installazione

- Gentoo (completamente basata su compilazione automatica)
- ArchLinux (tramite 'abs')
- Debian

...

Oppure: strumenti software (autopackage)

Continuità del supporto

Abbiamo detto, parlando delle licenze, che il modello di business adottato dai produttori di software proprietario, si basa sulla visione del software come prodotto.

La prima e maggiore conseguenza di questo modello, è la necessità di mantenere vivo il mercato, spingendo gli utenti ad acquistare, in modo da rientrare negli investimenti sostenuti.

Una volta saturato il mercato, si pone il problema di rinnovare la necessità di acquisto della clientela.

Continuità del supporto

Questo porta ad una pratica conosciuta come “obsolescenza programmata”.

Questa pratica consiste nel rendere necessaria (in maniera più o meno giustificata) la sostituzione di parti hardware o software del sistema in modo periodico, in modo da spingere gli utenti a più frequenti acquisti, che portano a maggiori guadagni da parte delle aziende produttrici.

Vengono così programmati i rilasci di nuove versioni del software, o di hardware più performanti, al fine di anticipare l'obsolescenza del prodotto venduto

Continuità del supporto

Nel mondo del software libero, che prende molto più a cuore le necessità degli utenti finali, in quanto protagonisti dello sviluppo stesso, l'obsolescenza programmata è una realtà inapplicabile.

Fintanto che qualcuno avrà bisogno del supporto per un hardware teoricamente obsoleto, questo supporto verrà portato avanti (al massimo, direttamente da chi ne ha bisogno).

Ecco allora che il più recente kernel Linux continua a girare tranquillamente su macchine 386.

Continuità del supporto

Questo non significa necessariamente che la release più recente della vostra distribuzione sia adatta a girare su hardware molto vecchio.

Ma compilando opportunamente un sistema, si può avere un sistema molto performante anche su macchine dichiarate obsolete.

Il problema dell'accrescimento dei requisiti hardware da parte dei software più recenti, affligge anche Linux. Una distribuzione GNU/Linux recente richiede circa 3GB per un'installazione completa.

Continuità dei formati

Un'altro aspetto dell'obsolescenza programmata, colpisce i formati.

Per formato, intendiamo quell'insieme di specifiche che determinano la struttura dei dati all'interno di un documento.

Per ragioni di mercato, i produttori di software proprietario tendono a mantenere segrete le specifiche dei formati dei loro software.

Quando l'obsolescenza programmata colpisce un determinato formato, questo non viene più mantenuto, portando gli utenti alla necessità di aggiornare il proprio software.

Continuità dei formati

Non dovrebbe suonarvi nuovo il fatto che i documenti salvati in formato nativo (.doc) con Word '95, non possono essere aperti da Office XP, in quanto il formato non è riconosciuto.

L'utilizzo di formati proprietari può portare (e capita sempre più spesso) alla perdita dei dati salvati nei nostri documenti.

Continuità dei formati

Con l'utilizzo di software libero, l'obsolescenza dei formati è un altro rischio che viene scongiurato.

Il fatto di possedere il sorgente del nostro programma, garantisce a tutti l'accesso alle specifiche del formato stesso.

Anche nel caso in cui il produttore del software da noi utilizzato decidesse improvvisamente di non mantenere il formato da noi utilizzato, possiamo sempre chiedere ad un programmatore professionista di scrivere un software in grado di leggere i nostri documenti.

Completezza

Il computer che avete a casa, quasi certamente, vi sarà arrivato con un'installazione di default di Windows.

Conoscerete quindi il software installato di default da Microsoft, per quel che riguarda il proprio sistema operativo:

- 1 client di posta elettronica (Outlook)
- 1 browser web (Internet Explorer)
- 1 player multimediale (Windows Media Player)
- 2 editor di testo (Notepad e WinPad)
- 1 suite per l'ufficio (quando c'è...)

Completezza

Sempre per conseguenza del tipo di licenze utilizzate, il software per GNU/Linux è disponibile in grande quantità.

Potete trovare un interessante elenco comparativo tra i vari software per Windows ed il loro corrispettivo sotto GNU/Linux, all'indirizzo

<http://linuxshop.ru/linuxbegin/win-lin-soft-en/table.shtml>

Il documento è aggiornato al 16.07.2003, ma già così, il rapporto tra software per Windows ed equivalente software su GNU/Linux, è di circa 1 a 3.

Tutto questo, perchè VOI possiate scegliere ciò che preferite.

Completezza

E questo è quel che riguarda il software disponibile.

Ma il software che viene effettivamente installato?

Abbiamo preso ad esempio, l'installazione completa di Slackware GNU/Linux 10.1.

Abbiamo abbandonato il tentativo di fare un elenco completo e dettagliato del software installato, quando abbiamo riempito la seconda slide di programmi.

Vi basti sapere che, oltre ai 10 editor di testo, 6 player multimediali, 8 client di posta e 5 browser web che avevamo elencato fino a quel momento, apparivano anche software per la grafica avanzata, 3 interfacce grafiche differenti e 25 diversi giochi.

Robustezza

Quante volte sento parenti amici e conoscenti che mi vengono a dire “Ah, oggi che disastro... stavo lavorando su un documento, quando si è bloccato tutto, e ho perso il lavoro”.

Quante volte, sento gente in vena di consigli che mi ricorda di “salvare spesso, che Windows si pianta”

Quante volte, VOI avete visto il vostro sistema bloccarsi improvvisamente, e rendere necessario il suo riavvio?

Sono le parole che contano, o i fatti? La Microsoft considera i propri sistemi stabili e sicuri. Eppure si bloccano, di tanto in tanto.

Robustezza

Non voglio tediarvi con spiegazioni del COME questo capita. Vi do solo un dato, l'unico “numero” che ho trovato cercando (<http://counter.li.org>)

3833.2 giorni = 10 anni (ever observed)

1861.3 giorni = 5 anni (max current)

[dati aggiornati al 31 ottobre 2006]

Sicurezza

Una delle cause della poca robustezza delle macchine Windows, è molto probabilmente il proliferare di virus.

Le cause di questo proliferare, non possono essere attribuite solo (come qualcuno vuol farci credere) al fatto che Windows sia più diffuso degli altri sistemi operativi, e quindi “al centro dell'attenzione”.

Sicuramente LA causa della diffusione dei virus su Windows, è che il sistema viene costantemente utilizzato con privilegi di amministratore.

Sicurezza

Cosa significa “privilegi di amministratore”?

Questa è la classica domanda che mi viene rivolta a questo punto, ed è proprio la prova della insufficiente gestione utenti dei sistemi Microsoft.

Su tutti i sistemi che si rispettino, esistono vari utenti, con livelli di privilegi diversi.

Non è necessario che TUTTI gli utenti possano accedere a TUTTE le funzioni del computer.

A cosa mi serve avere la possibilità di installare software sulla macchina, quando devo solo fare lavoro d'ufficio, navigare il web o inviare e ricevere posta?

Sicurezza

Ecco allora che si rende possibile (e necessario, parlando di sicurezza) suddividere i privilegi tra i vari utenti, differenziando l'utente amministratore (che avrà la possibilità di installare il software) dagli utenti con cui la macchina viene utilizzata.

Nel momento in cui un utente normale vuole installare software, dovrà fare IN PIU il cambio di utente (un comando ed una password), ma dall'altra parte, chi potesse compromettere il sistema con i privilegi di utente generico (un virus ad esempio) non potrà auto-installarsi sul sistema.

Sicurezza

Questo spiega come, mentre su Windows (che ha una gestione degli utenti piuttosto inefficiente, e che utilizza l'utente con privilegi di amministratore come utente di default) si trova sommerso dai virus, altri sistemi operativi (come MacOSX, o GNU/Linux) non soffrano minimamente di questo problema.

PARE ESISTANO 5 VIRUS SU GNU/LINUX

(e comunque non funzionano)

CONTRO I CIRCA 73.000 PER WINDOWS

(fonte aggiornamenti antivirus Symantec)

Performance

I moderni sistemi operativi, ci hanno abituato a poter eseguire più programmi contemporaneamente.

Questa ci sembra oggi un'operazione assolutamente normale, tanto che ci stupisce il fatto che fino a meno di 20 anni fa, i sistemi operativi non erano in grado di farlo.

Ancora più stupefacente, è il fatto che a tutt'oggi, il processore (il vero cuore elaborativo dei nostri computer) continua a non poter eseguire che un'operazione alla volta.

Si rende quindi necessario creare l'illusione che più programmi possano essere eseguiti in contemporanea.

Performance

Per creare questa illusione, esistono due diversi e distinti modi:

- Il multithread
- Il multitask

Il multithread si applica alla parallelizzazione all'interno di uno stesso programma.

Il multitask, invece è il parallelismo tra programmi diversi. Per ottenere l'effetto di multitasking, si deve implementare un'operazione chiamata “contest switching”. Si tratta di “memorizzare” temporaneamente tutti i dati relativi ad una applicazione, per poter passare ad eseguire un altro processo.

Performance

A causa di un'implementazione del “contest switching” non ottimale, il multitasking è estremamente penalizzante sui sistemi operativi Microsoft.

Ecco quindi che quando abbiamo diverse applicazioni che lavorano in parallelo (Office e un lettore mp3 e un browser web e un client di posta), il sistema comincia a rallentare vistosamente, rendendone quasi impossibile la reattività.

Questo non succede assolutamente su GNU/Linux, dove il “contest switching” è estremamente veloce.

Performance

La dimostrazione di ciò, purtroppo, è piuttosto laboriosa.

Occorrerebbero due macchine uguali, l'utilizzo di software di benchmark, e non abbiamo che pochi minuti.

Vi invito a provare con una qualsiasi distribuzione Live (che non necessita di installazione) a testare la reattività di un sistema GNU/Linux sotto carico.

Sfatiamo alcuni miti

“Mi hanno detto che Linux è difficile”

29 milioni di utenti usano GNU/Linux, e non sono certo tutti guru dell'informatica.

L'installazione del sistema operativo, ormai, è semplice tanto quanto (se non più) quella di Windows.

Il supporto della community è forse LA risorsa più importante di questo sistema operativo.

Perché dovrebbe essere difficile? Perché cambiano le icone? Vi lasciate scoraggiare da così poco?

Posso solo invitarvi a provarlo, per dimostrarvi che di difficile non ha proprio nulla!

Sfatiamo alcuni miti

“Per installare Linux devo togliere Windows”

L'idea che su un computer ci possa essere un solo sistema operativo alla volta, è un retaggio del monopolio Microsoft.

Sul computer su cui sta girando la presentazione che vedete, sono installati Windows XP e Slackware GNU/Linux 10.1.

Il disco fisso è semplicemente stato suddiviso (partizionato) in due blocchi fin dal giorno della prima accensione. Su uno gira Windows, sull'altro GNU/Linux, ed entrambi funzionano a meraviglia.

Sfatiamo alcuni miti

“Per provare Linux devo installarlo per forza”

Negli ultimi anni, abbiamo assistito ad un fiorire delle distribuzioni così dette “live”.

Si tratta di distribuzioni che stanno su un singolo CD, e che possono essere “lanciate” senza bisogno di alcun processo di installazione.

Il sistema precedentemente installato non viene in alcun modo modificato, e tutto quello che verrà fatto dal sistema “live” sarà completamente temporaneo (in quanto risiede in ram, non su disco).

Sfatiamo alcuni miti

“Non so programmare, a cosa mi serve il sorgente?”

La disponibilità del sorgente non è direttamente legata all'uso del sistema operativo.

Si tratta più che altro di una “garanzia”.

Si può utilizzare il sorgente, come abbiamo visto, per l'installazione del software, ma questo non è necessariamente il procedimento che vorrete seguire, poiché potrete disporre di tutta una serie di pacchetti precompilati.

Sfatiamo alcuni miti

“Ma Windows e Linux sono compatibili?”

Suddividiamo in due il concetto di “compatibilità”:

- Compatibilità hardware
- Compatibilità software

Per quanto riguarda la compatibilità hardware, problemi non ce ne sono. GNU/Linux è disponibile per un'infinità di piattaforme diverse, tra le quali spicca sicuramente la piattaforma x86, che è quella utilizzata anche da Windows.

L'unico problema può essere rappresentato da alcune classi di periferiche, per le quali il supporto è ancora in via di sviluppo.

Sfatiamo alcuni miti

“Ma Windows e Linux sono compatibili?”

A livello di software invece, il software che gira su Windows NON gira su GNU/Linux.

Esistono modi per ottenere questa compatibilità, grazie ad implementazioni su GNU/Linux delle interfacce di programmazione (API) utilizzate dal software su Windows. Molti programmi (e giochi) possono girare così su entrambi i sistemi.

Ma perchè usare software in “emulazione”, quando possiamo usufruire di una quantità inimmaginabile di software di altissima qualità, disponibile nativamente per GNU/Linux?

Sfatiamo alcuni miti

“Ma Windows e Linux sono compatibili?”

Esistono su GNU/Linux software per fare qualunque cosa voi facciate su Windows: dalla grafica, alla gestione della posta, dal software per aprire le presentazioni “Power Point”, a quelli per lo sviluppo di pagine web simili a DreamWeaver.

Si tratta solo di trovare il software che fa per voi, magari con l'appoggio di qualcuna delle associazioni culturali che si occupano di promuovere l'OpenSource (come il POuL?).

Se hai dubbi, chiedi!

Questa veloce panoramica su GNU/Linux finisce qui.

Le associazioni culturali che promuovono GNU/Linux e l'OpenSource in generale, sono moltissime e a disposizione (spesso e volentieri in forma completamente gratuita) per rispondere alle vostre domande.

<http://www.openlabs.it>

<http://www.poul.org>

(or <mailto:alt-os@poul.org>)