

# Corso Di Linux

Liceo Fermi, Bologna - Marcello Galli, novembre 2009

## Introduzione

La storia di Linux inizia da Unix, perche' in realta' Linux non e' che un sistema Unix, anche se e' un Unix rifatto da capo.

Il sistema Unix e' stato sviluppato negli anni 70, nei laboratori della AT&T, la compagnia telefonica americana, quando Internet non esisteva, i PC non esistevano, ed i computers erano giganteschi, costosissimi ed avevano prestazioni peggiori delle CPU di certi telefonini di oggi.

I sistemi operativi erano pero' stati gia' molto studiati e praticamente tutte le cose principali dei sistemi operativi erano state capite e sperimentate. Unix era una cosa nuova, che raccoglieva tutta l'esperienza sui sistemi operativi di allora. Pero' era un po' un sistema da laboratorio, con una struttura semplice e versatile, potente, ma privo di fronzoli e fatto per essere usato da gente esperta di computers.

Nella figura si vedono due dei creatori di Unix, Ken Thompson e Dennis Ritchie, che lavorano al computer che avevano recuperato per svilupparci Unix. Come si vede non c'era il video, neanche il mouse e per l'input e l'output si dovevano battere istruzioni su una telescrivente: una specie di macchina da scrivere lentissima e rumorosa. Dicono anche che le prime telescriventi avessero tasti molto duri da spingere; pero' erano un grosso passo avanti rispetto alle schede perforate che si erano usate fino ad allora.



Figure 1: Gli autori di Unix al lavoro

La AT&T in quel periodo aveva problemi con la giustizia, infatti aveva il monopolio dei telefoni ed era sotto accusa come monopolista. Per questo non poteva vendere Unix, e lo dava quasi gratis alle universita' americane. Unix e' quasi tutto scritto in linguaggio C, e veniva dato assieme al programma sorgente, per cui gli studenti potevano modificarlo e farci esperimenti, aggiunte e modifiche.

Il risultato fu che Unix si sviluppò molto, specie all'università di Berkeley, ma in modo un po' caotico, infatti ogni studente che ci metteva le mani o scriveva programmi per Unix seguiva criteri suoi, poi i programmi venivano usati da tanta gente, che si abituava a quella sintassi che non veniva più cambiata. Per questo motivo i comandi per una macchina Unix sono tutti abbreviati (pensate doverli scrivere su una telescrivente) ed hanno sintassi diverse uno dall'altro. Ad esempio il comando "list" (che mostra un elenco dei files) è diventato "ls"; "print working directory" è diventato "pwd"; il comando "change directory" è diventato "chd" e così via. Tutte sigle inpronunciabili, ma ormai la gente si è abituata e nessuno si mette a cambiare i comandi, anche perché così abbreviati si fa presto a scriverli.

Quando iniziarono a svilupparsi le prime reti di computer, il software di rete fu subito inserito in Unix, che divenne un sistema molto adatto ad essere utilizzato in rete, anzi si può dire che le reti di computer si sono sviluppate assieme ad Unix.

A metà degli anni 80 la AT&T si liberò dei suoi problemi legali, e si mise a vendere Unix, senza dare più i sorgenti in linguaggio C. In quel periodo nascevano le prime workstation grafiche: computer con mouse e video, ed i fabbricanti di workstation trovarono che Unix era un sistema operativo già pronto da adattare ai loro computers con poca spesa, lo comprarono dalla AT&T, e svilupparono la loro versione di Unix, che vendevano assieme alle loro macchine. Per cui in quel periodo c'erano 6 o 7 diversi tipi di Unix, tutti simili ma tutti un po' diversi, tutti molto cari da comperare e tutte le ditte si tenevano nascosto il codice sorgente di Unix, per non farlo copiare alla concorrenza.

Le workstation grafiche ebbero grande diffusione ed Unix si diffuse con loro, diventando una specie di standard per i sistemi operativi, con tutti i suoi comandi abbreviati e tutte le cose che ci avevano aggiunto gli studenti delle università, che nessuno pensava più di mettere in ordine.

Però non c'era più in giro un sistema operativo da modificare e studiare, per imparare e fare esperimenti, ed in molte università si finiva per fare soprattutto teoria di sistemi operativi e poca pratica. Andrew Tannenbaum, che adesso è un vecchio un professore di un'università di Amsterdam, cercò di rimediare a questo problema, scrivendo un piccolo sistema Unix per gli studenti, chiamato "Minix".

Però Tannenbaum non voleva che troppa gente mettesse le mani dentro Minix, altrimenti diventava troppo grosso e complicato per gli studenti, e poi, se uno voleva Minix da modificare, doveva anche comperare il suo libro. Quindi minix non era proprio quello che gli studenti e gli appassionati di sistemi operativi volevano: un sistema gratis su cui pasticciare liberamente.

Chi fornì questo sistema fu Linus Torwalds, che allora era studente di informatica ad Helsinki (siamo ormai negli anni 90). Linus si era comperato uno dei primi PC, ma non era contento del DOS, ci mise Minix, ma non era contento neanche di quello, e allora provò a scrivere un sistema operativo nuovo per il suo PC, che fu chiamato Linux, ed era ovviamente una specie di clone di Unix, visto che Unix era un riferimento per tutti.

Iniziava allora la rete Internet, e, tramite internet, subito gli appassionati di informatica iniziarono a lavorare ed a sviluppare Linux: un oggetto gratuito su cui pasticciare. A Linux fu abbinato il software della GNU, una associazione fondata da Richard Stallman (un personaggio molto particolare), che voleva il software circolasse sempre assieme al source, per poter essere modificato dagli utenti.

Stallman voleva fare un sistema completo; aveva i compilatori e molti programmi ausiliari, ma gli mancava il kernel. Linux era il kernel che ci voleva; il kernel, più i compilatori e tutto il software della GNU costituiscono un "sistema Linux".

Linux si sviluppò velocemente, e gli sviluppatori ci misero dentro di tutto, software per tanti tipi di reti, drivers, file systems, etc. etc.. Siccome è scritto quasi tutto in C è relativamente facile adattarlo per farlo andare su computer di tipo diverso, e quindi è stato portato su decine di computers diversi.

Va sui PC, e siccome ormai i PC sono diventati così potenti da soppiantare anche i sistemi server delle grosse aziende, ad un certo punto le ditte che fanno i server si sono messe a vendere server con CPU tipo quelle dei PC, e ci mettono sopra Linux invece dei loro vecchi Unix proprietari. Infatti gli costa molto meno pagare qualcuno che lavori su Linux per svilupparlo e fare qualche aggiunta apposta per loro che fare un sistema operativo nuovo. Moltissime ditte ora utilizzano Linux per le loro apparecchiature, specie piccoli apparecchi come modem, router etc.. È un sistema che funziona, già pronto, per adattarlo basta pagare un piccolo gruppo di programmatori. Un sistema operativo nuovo invece ha costi altissimi. Questo è uno dei motivi di successo di Linux: costa poco e funziona bene.

Costa poco perché c'è tanta gente che ci lavora, ognuno fa un po' del lavoro e nessuno deve pagarlo tutto. In questo senso si può dire che Linux segue un modello commerciale anomalo, ove i costi di sviluppo sono distribuiti, il prodotto non appartiene a nessuno ed i guadagni vengono dal mercato indotto: servizi ed hardware. In definitiva Linux è gratis perché dietro ci sono un sacco di soldi.

## A chi appartiene Linux ?

Non c'è nessuno che ufficialmente è "proprietario" di Linux.

Linux è un programma di cui ognuno può fare quello che vuole; gli unici vincoli sono dati dalla licenza sotto cui è distribuito, che è la licenza "GNU", ideata da Richard Stallman, con lo scopo di favorire la diffusione del software e dare libertà ai programmatori. Questa licenza dice essenzialmente che quando si vende o si distribuisce del software chi lo riceve deve poter accedere al source e poterne fare quello che vuole e modificarlo come gli pare, ma, se lo vende o lo distribuisce a sua volta, anche modificato, deve dare, a chi lo riceve, sempre facoltà di fare quello che vuole del software.

Per questo il software con licenza GNU non piace a chi scrive software "proprietario" e non vuole che i suoi clienti mettano il naso nei programmi che comperano; se infatti usa nel suo programma software sotto licenza GNU, poi deve dare ai suoi clienti anche il source e quelli possono farne quello che vogliono, come modificarlo e rivenderlo.

C'è un'eccezione a questa regola: il software della GNU può essere abbinato a software "proprietario" se è utilizzato in forma di librerie, cioè come parte staccata dal programma proprietario, di cui si vuole tenere segreto il source.

Per questo i costruttori di schede grafiche 3D per i giochi (come la nVidia), danno driver per Linux che sono "moduli" del kernel, cioè parti separate dal kernel. In questo modo possono tener segreti i loro driver e non farsi spiare dalla concorrenza. Altrimenti dovrebbero dare a chi compra una scheda grafica anche il source del driver e permettergli di modificarlo, cosa che non vogliono assolutamente fare. I moduli "proprietary" abbinati al kernel danno sempre un sacco di problemi: il kernel si evolve velocemente ed i produttori di hardware non si curano di prodotti un po' vecchi, per cui alla fine i driver proprietari ed il kernel non corripodono mai bene e si fa fatica a far funzionare un certo hardware con Linux. Le schede grafiche per i giochi sono un classico esempio.

C'è sempre molta discussione sul fatto di ammettere o no driver proprietari come moduli del kernel, ma alla fine nessuno molla: i produttori di hardware non vogliono far vedere i loro programmi, quelli della GNU non vogliono rinunciare alla loro licenza e non distribuiscono i driver proprietari, e la gente vuole usare l'hardware con Linux, per cui la situazione resterà confusa, come è adesso. Se volete usare al meglio l'ultima scheda grafica Nvidia con Linux vi tocca cercare il driver sul sito web della nVidia e compilare voi il programma di collegamento fra il driver ed il kernel; c'è una procedura della nVidia che vi aiuta e fa tutto lei, per cui alla fine la scheda va (il più delle volte), ma c'è sempre del lavoro da fare e qualche problemino.

## Come è organizzato lo sviluppo di Linux

Di Linux ognuno può fare quello che vuole, ma siccome mantenere un grosso sistema operativo è un lavoro enorme gli sviluppatori si sono organizzati.

Tutto il lavoro è pubblico, anche le liti e gli insulti fra gli sviluppatori (sono abbastanza litigiosi). C'è una mailing list, anzi diverse mailing list, divise per argomento, se volete sbirciare trovate gli archivi in internet: <http://www.ussg.iu.edu/hypermil/linux/kernel/> È una lista piena di argomenti tecnici, vi risulterà incomprensibile; gli sviluppatori si mandano lunghi pezzi di codice via mail, a volte c'è anche dell'assembler. Se non siete degli specialisti del kernel non ci capirete niente (a parte quando litigano, quello si capisce bene). In <http://kernel.org> si capisce di più; lì ci sono tutte le versioni del kernel. L'archivio con tutte le proposte di modifiche al kernel invece è in: <http://git.kernel.org>, ma anche quello è molto tecnico. C'è anche il sito per i principianti del kernel: <http://kernelnewbies.org/> ma è per principianti già esperti di informatica. Un kernel non è una cosa semplice, se volete studiare un

sistema operativo iniziate con qualche libro un po' introduttivo, sono molto belli i libri di Tannenbaum su Minix, anche se a livello universitario ed abbastanza impegnativi.

Ognuno puo' farsi il kernel con le modifiche che vuole, ma come si fa a far aggiungere le proprie modifiche al kernel ufficiale? Potrebbe sembrare semplice: si organizza il software secondo le istruzioni che sono sul sito internet del kernel e lo si manda alla lista. A questo punto sembra che la parte semplice finisca; molti lamentano di venir ignorati dagli altri lettori della lista, oppure maltrattati e criticati selvaggiamente per il pessimo codice che scrivono, chi sopravvive alle critiche ed insiste (e forse ha fatto un buon lavoro) alla fine ottiene l'attenzione di qualche sviluppatore importante che si occupa di una parte specifica del kernel, questo riguarda e correggere il codice degli altri e, se lo soddisfa, lo mette nel suo archivio. Anche lui fa correzioni e critiche. Quando il software lo soddisfa lo manda a Linus, od a Andrew Morton, che e' un po' il suo vice, solo le modifiche che Linus approva vanno a finire nel kernel ufficiale.



Figure 2: Gli sviluppatori del kernel (Linux Kernel Summit 2009, Tokyo)

E' sorprendente come lo sviluppo del kernel, fatto in questo modo, non abbia alcuna organizzazione. In realta' l'organizzazione c'e' ed e' molto piu' rigida di quanto sembri, Linus ed il suo gruppo di collaboratori di fiducia decidono tutto loro, mettendosi d'accordo via mail o alle riunioni in cui si incontrano una paio di volte l'anno. Non ci sono incarichi ufficiali, ma in realta' ognuno si occupa di una parte specifica del kernel, che conosce bene, e fa da revisore per il software di quella parte del kernel. Le modifiche, devono superare 2 livelli di revisione, una revisione pubblica, da parte della lista, ed una da parte dell'incaricato della sezione del kernel interessato, alla fine solo Linus decide cosa viene messo nel kernel. Linus si fida in realta' dei suoi collaboratori, perche' le modifiche proposte sono migliaia e non puo' guardarle tutte.

Per entrare nel gruppo degli sviluppatori principali bisogna convincerli di essere bravi e saper lavorare bene, altrimenti non ti prendono e non accettano neanche le tue aggiunte al kernel. In questo modo solo la roba buona arriva al kernel, il resto non sopravvive alla marea di critiche. Insomma e' una giungla organizzata per far sopravvivere solo il software migliore. Tenete presente che migliaia di persone collaborano al kernel e le proposte di modifica sono tantissime, molte hanno poi dietro grossi interessi commerciali.

Gli sviluppatori stessi sono pagati da ditte diverse, spesso in concorrenza fra loro. Linus lavora all' "Open Source Development Laboratory" della "Linux Foundation" ( <http://www.linuxfoundation.org/> ) una istituzione finanziata da un consorzio di produttori di hardware e software, altri sono impiegati di grosse ditte come HP, Novell, Red Hat, Intel. Ci sono anche appassionati, ma la maggior parte degli

sviluppatori del kernel sono professionisti pagati da ditte interessate al kernel per i loro prodotti.

Le discussioni e le liti fra gli sviluppatori sono all'ordine del giorno. Il sistema funziona perche' nessuno puo' barare, c'e' troppa competizione e troppi interessi in gioco, il lavoro di tutti e' sotto gli occhi di tutti ed alla fine il tuo software deve funzionare bene, oppure subito si viene criticati e ci si fa una brutta fama. Gli sviluppatori finiscono per guadagnare di piu' se sono famosi e stimati.

## Un sistema Linux

Linux, per essere precisi, e' solo il kernel, cioe' il sistema operativo. Ma in genere quando si parla di Linux si intende un sistema completo, con tanti programmi per fare le cose piu' disparate.

Infatti oltre al kernel Linux esistono tanti altri progetti che producono software distribuito in modo gratuito, sotto licenze molto libere, tipo la GPL. Abbiamo gia' parlato del progetto GNU (<http://www.gnu.org>), della "free software foundation" di Richard Stallman, che vuole produrre un sistema operativo "libero", nel senso che il software e' distribuito con il source e si da' facolta' di modificarlo e rivenderlo. Questo progetto produce tanti programmi utili, quelli classici per Unix, in piu' compilatori gratuiti per i linguaggi piu' importanti, editors per i testi e tante altre cose. Queste cose sono sempre fornite assieme al kernel tanto e' vero che e si parla di sistemi GNU/Linux, perche' il kernel per essere compilato ed usato utilizza il software del progetto GNU.

C'e' poi Openoffice, un insieme di programmi tipo Office della Microsoft, distribuito gratis dalla Sun (ditta che fa grossi computers), per fare concorrenza ai sistemi con software Microsoft.

Poi c'e' il sistema grafico sviluppato al MIT per le macchine Unix degli anni 80: X11, che e' il sistema che gestisce il video sia nelle macchine Unix che in Linux. Anche questo, sviluppato da un consorzio, e' gratis.

Ci sono 3 o 4 sistemi alternativi per il gestire le finestre del desktop, come KDE, GNOME, XFCE. Anche questi sono sviluppati da grosse collaborazioni, a volte con dietro da grosse ditte.

C'e' Apache, il programma per i server web, sviluppato da un consorzio, che e' utilizzato dall'80% dei server del mondo.

C'e' tantissimo software, piu' di quanto se ne possa elencare, in parte sviluppato da appassionati e studenti, in parte da ditte, per un motivo o per un altro disponibile in forma gratuita.

Per avere un sistema completo dovremmo fare una raccolta di un po' di questo software, e metterlo insieme in modo organico. E' un grosso lavoro, ma per fortuna c'e' chi lo fa per noi, e sono quelli che fanno le "distribuzioni Linux".

Queste "distribuzioni" sono raccolte di software, e contengono il kernel, i programmi della GNU e tanto altro software, tutto ordinato in modo coerente, con aggiunta di qualche tool per la gestione di tutto il sistema. Alcune di queste si scaricano gratis dalla rete, altre sono a pagamento, ma costano poco, infatti se costassero molto la gente non le comprerebbe, si farebbe la distribuzione da se, visto che e' tutto software gratuito che si trova in rete. Chi fa la distribuzione cura anche gli aggiornamenti, specie quelli per correggere bachi del software legati alla sicurezza. Per questo e' utile utilizzare una distribuzione, altrimenti si dovrebbero seguire tutti i bollettini sulla sicurezza ed aggiornarsi il software da soli. Un lavoro che richiede molto tempo.

Le distribuzioni hanno poi dei programmi che facilitano l'installazione e la rimozione di software, organizzato in "pacchetti": files compressi che contengono le diverse parti di un programma, tipo i files .exe autoestraenti usati per installare programmi sotto Windows. I programmi di gestione dei pacchetti gestiscono anche le dipendenze fra i diversi programmi e le librerie. Ogni distribuzione ha il suo programma di gestione del software, certe hanno anche del software grafico che facilita la gestione del sistema.

Scegliere una distribuzione o l'altra e' una questione di gusti, non ha senso parlare di Debian, Red Hat, Ubuntu, come se fossero cose diverse, sono solo diverse "compilation"; come quando si fanno le raccolte di brani musicali da ascoltare con l'iPod, come li metti in ordine conta, ma non e' la collezione che fa i brani.

Quindi quando si installa Linux su una macchina non si installa solo il sistema operativo, ma una enorme raccolta di software che contiene di tutto.

Ci sono centinaia di distribuzioni, c'è anche un libro (Linux from Scratch) che spiega come farsi una distribuzione da se, ma le distribuzioni più diffuse sono poche, vediamo alcune e le loro caratteristiche.

- Slackware: è stata una delle prime, agli inizi era molto spartana, adesso è più user-friendly, ma è una distribuzione dove l'utente deve fare un certo lavoro per la gestione del sistema. Per esempio il sistema di gestione dei pacchetti non gestisce le dipendenze, in modo automatico, ma per le dipendenze ci sono programmi a parte.
- Suse: originariamente una distribuzione tedesca, utilizzata soprattutto per i server, è stata poi comperata dalla Novell, americana, per 200 milioni di dollari. Ha ora una versione per server: "SUSE Linux Enterprise", che si paga, ed una versione gratuita: "openSUSE", ove viene fatta la sperimentazione delle componenti che poi andranno nella versione commerciale. La SuSE ha un programma unificato per la gestione del sistema, che si chiama Yast.
- Red-Hat: è americana, è forse la distribuzione commerciale più diffusa, ha un giro d'affari di parecchie centinaia di milioni di dollari annui. Anche questa ha una versione per server, a pagamento, ed una liberamente disponibile, su cui viene fatta la sperimentazione: Fedora.
- Mandriva: è la fusione della Connective (brasiliiana) e della francese Mandrake, a sua volta derivata dalla Red-Hat. Diffusa in Francia.
- Debian: è una distribuzione abbastanza peculiare, progetto iniziato da Ian Murdock, è una distribuzione mantenuta da un migliaio di volontari, organizzati secondo regole democratiche, con una costituzione, comitati, votazioni etc. Contiene tantissimo software, ma rilascia nuove versioni raramente, non ogni 6 mesi come molte altre distribuzioni, anche perché la preparano non solo per PC, ma anche per macchine di altri tipi. È organizzata molto bene, ma richiede un certo lavoro per configurare il sistema, poiché la sua filosofia è che è l'utente che deve scegliere come configurare il suo sistema e non la distribuzione.
- Knoppix: è una derivata dalla Debian. Klaus Knopper, un ingegnere tedesco, la ha concepita come un sistema operativo che sta su un CD, corre utilizzando il CD e non scrive nulla sui dischi del computer. Intesa per provare Linux senza doverlo installare sul computer è poi diventata una distribuzione. Oggi molte distribuzioni hanno versioni cosiddette "live" cioè che stanno su CD e non scrivono sui dischi del computer.
- Ubuntu: anche questa è una derivata della Debian. È stata creata da un miliardario sudafricano: Mark Shuttleworth, con l'idea di una distribuzione che potesse diffondersi non solo in ambiente di macchine server, ma anche essere utilizzata al posto di Windows, da non esperti di informatica. Shuttleworth ha assunto un po' di sviluppatori Debian ed ha messo su una distribuzione che fa quello che manca alla Debian: rilascia nuove versioni ogni 6 mesi e produce un sistema quasi tutto già configurato, pronto da usare. Produce anche una versione per server, per cui garantisce aggiornamenti per la sicurezza per 5 anni. Ubuntu ha avuto molto successo, ma non tanto da soppiantare Windows.

## Altri sistemi operativi gratuiti

Linux non è il solo sistema operativo gratuito in circolazione. In particolare bisogna ricordare i diversi tipi di Unix che sono stati sviluppati all'università del Berkeley, quando la At&T ha smesso di dare il source di Unix. Hanno avuto meno successo di Linux perché sono arrivati tardi, a causa di dissidi fra gli sviluppatori, ma hanno il loro mercato ed i loro fan. Sono 3: OpenBDS, NetBSD e FreeBSD.

FreeBSD è forse il più usato, ha buona reputazione come sistema per i server ed ha una licenza meno restrittiva della licenza GNU, può quindi essere incluso in prodotti commerciali; per questo parti di FreeBSD sono nel sistema operativo degli Apple, ed in tanti apparati elettronici.

© Marcello Galli, Novembre 2009. Sito di riferimento: <http://www.helldragon.eu>